Exercise 3: Understanding the Impact of Partitioning and Index Policy

Overview

In this exercise, you will evaluate the impact of different partitioning and indexing strategies on the performance of some common operations in Cosmos DB.

Time Estimate

60 minutes

**Scenario:** You have a dataset stored as a JSON file that contains ball-by-ball summaries of international cricket matches. Each document in the dataset has the following format.

The **info** subdocument contains the overall information about the match, such as the teams, location, match type (T20, ODI, Test), the umpires, the toss (the winner chooses whether to bat or bowl first), and the match result.

The **innings** subdocument is a ball-by-ball summary of an innings, describing who the bowler was, the batsman facing the delivery, and the outcome of the delivery (were any runs scored, was it a no-ball or wide, was the batsman out and if so, how, and so on).

For a T20 and an ODI match, there will be one innings per team. A T20 match has 20 overs per side (an over is 6 deliveries) in an innings (120 deliveries plus any extras), and an ODI has 50 overs per side (300 deliveries plus extras). A Test match has up to two innings per team, and each innings can vary in length (a team bats until all its players are out or they declare).

You want to store this information in a document database, but want to assess the most optimal way to index the data, and the affects that different partitioning strategies will have on the performance of some common queries, such as:

* Find the details of matches of a particular type (T20, ODI, Test) where team XYZ was playing at home.
* Find the details of matches of a particular type where team XYZ was playing away.
* Find the details of matches of a particular type where team XYZ was playing team ABC.
* Find the details of matches of a particular type played in a specified city.
* Find the details of all matches of a particular type.

Task 1: Create a New Cosmos DB Database and Verify Consistency

1. From the Azure portal, navigate to your Cosmos DB instance. Under **SETTINGS**, click **Default consistency**, and verify that the default consistency for the account is **Session**.
2. Click **Data Explorer**, and then click **New Database**.
3. On the **New Database** blade, in the **Database id** box, type **Cricket**, and then click **OK**.
4. Under **SETTINGS**, click **Keys**.
5. Make a note of the **PRIMARY CONNECTION STRING** value for use later in the lab.

Task 2: Test the Effects of Index Policy on Insert Operations

1. In File Explorer, navigate to **C:\OpsgilityTraining\dt-1.8.1**, and double-click **dtui.exe**.
2. In the **DocumentDB Data Migration Tool** window, on the **Welcome** page, click **Next**.
3. On the **Source Information** page, in the **Import from** drop-down list, choose **JSON files(s)**, and then click **Add Files**.
4. In the **Open File** dialog box, navigate to the **C:\OpsgilityTraining\Data** folder, click **MatchesData.json**, and then click **Open**.
5. On the **Source Information** page, click **Next**.
6. On the **Target Information** page, in the **Export to** drop-down list, choose **DocumentDB - Sequential record import (partitioned collection)**.

**Note:** The collection created by this load operation is not actually going to be partitioned, but selecting the **Sequential record import** option will give you a fair comparison of the impact of indexing with subsequent load operations that import data into a partitioned collection.

1. In the **Connection String** box, enter the **PRIMARY CONNECTION STRING** for your Cosmos DB account.
2. At the end of the string, append the text **database=Cricket**. At this point you should be able to verify your connection successfully.
3. In the **Collection** box, type **Matches**.
4. Leave the **Partition Key** box empty.
5. Leave the **Collection Throughput** box set to **1000** (the default).
6. In the **Id Field** box, type **id**.
7. Expand **Advanced Options**, select the **Disable Automatic Id Generation** check box, and then click **Next**.
8. On the **Advanced** page, click **Next**.
9. On the **Summary** page, click **Import**.

**Note:** While the import is running, observe that the load rate is very slow. This is because of the index applied to the **innings** subdocument. This subdocument contains many fields, and constructing the index for this subdocument can take a considerable time.

1. Leave the import running.
2. In the Azure portal, navigate to your Cosmos DB instance and click **Data Explorer**.
3. In the **SQL API** pane, expand **Cricket**, expand **Matches**, and then click **Scale & Settings**.
4. On the **Scale & Settings** tab, change the **Throughput** to **10000**, and then click **Save**.
5. In DocumentDB Data Migration Tool, notice that the rate of import improves significantly. However, you are now incurring ten-times the financial cost.
6. Wait for the import operation to complete (there are 3939 documents).
7. In Internet Explorer, on the **Scale & Settings** tab, change the **Throughput** to **1000** (to save costs), and then click **Save**.
8. In the DocumentDB Data Migration Tool, click **New Import**.
9. In the **New Import** dialog box, click **No** to reuse the settings from the previous run.
10. On the **Source Information** page, ensure that the JSON file **C:\OpsgilityTraining\Data\MatchesData.json** is specified, and then click **Next**.
11. On the **Target Information** page, ensure the connection string is the same as before.
12. In the **Collection** box, type **MatchesInningsNotIndexed**.
13. Leave the **Partition Key** empty.
14. Leave the **Collection Throughput** set to **1000** (the default).
15. Leave the **Id Field** set to **id**.
16. In the **Advanced Options** section, leave the **Disable Automatic Id Generation** check box selected.
17. In the **Enter Indexing Policy** box, type the following code to specify the following policy. This policy prevents the fields in the **innings** subdocument from being indexed (it is highly unlikely that you would want to search this subdocument for information about a specific delivery, rather you are more likely to process this data sequentially, so an index is superfluous):
18. {
19. "indexingMode": "consistent",
20. "automatic": true,
21. "includedPaths": [
22. {
23. "path": "/\*",
24. "indexes": [
25. {
26. "kind": "Range",
27. "dataType": "Number",
28. "precision": -1
29. },
30. {
31. "kind": "Hash",
32. "dataType": "String",
33. "precision": 3
34. }
35. ]
36. }
37. ],
38. "excludedPaths": [
39. {
40. "path": "/innings/\*"
41. }
42. ]

}

1. On the **Target Information** page, click **Next**.
2. On the **Advanced** page, click **Next**.
3. On the **Summary** page, click **Import**. Notice that the import proceeds far more quickly than before without the need to increase the **Throughput** for the database.
4. Wait for the operation to finish, and verify that it imports 3939 documents.

**Note:** If you decide later that you do require an index over the **innings** data, you can modify the index policy for the collection using the Azure portal.

Task 3: Create Collections with Different Partitioning Strategies

1. In DocumentDB Data Migration Tool, click **New Import**.
2. In the **New Import** dialog box, click **No** to reuse the settings from the previous run.
3. On the **Source Information** page, specify the JSON file **C:\OpsgilityTraining\Data\MatchesData.json**, and then click **Next**.
4. On the **Target Information** page, ensure the connection string is the same as before.
5. In the **Collection** box, type **MatchesPartitionedByHomeTeam**.
6. In the **Partition Key** box, type **/info/hometeam**.
7. Leave the remaining fields at their current settings, and then click **Next**.
8. On the **Advanced** page, click **Next**.
9. On the **Summary** page, click **Import**. Notice that the import proceeds far more quickly than before without the need to increase the **Throughput** for the database.
10. Wait for the operation to finish.
11. Repeat steps 1 to 10 to import the data into a **Collection** named **MatchesPartitionedByAwayTeam**, and set the **Partition Key** to **/info/awayteam**.
12. Repeat steps 1 to 10 to import the data into a **Collection** named **MatchesPartitionedByMatchType**, and set the **Partition Key** to \*\*/info/match/\_type\*\*.
13. Repeat steps 1 to 10 to import the data into a **Collection** named **MatchesPartitionedByCity**, and set the **Partition Key** to **/info/city**.
14. Close the DocumentDB Data Migration Tool.

Task 4: Compare the Performance of Different Partitioning Strategies

1. On the Start menu, click **Visual Studio 2017**.
2. On the **File** menu, point to **Open**, and then click **Project/Solution**.
3. In the **Open Project** dialog box, navigate to the **C:\OpsgilityTraining\CricketQuery**, click **CricketQuery.sln**, and then click **Open**.
4. In Solution Explorer, click **Queries.cs**.
5. On the **Queries.cs** tab, examine the contents of the file.
6. The **Queries** class contains a number of methods that implement the queries specified at the start of this demonstration. Notice that the **queryOptions** field defined near the start of the class specifies the **FeedOptions** that are used by all of the queries; these options allow cross partition queries, scanning for non-indexed data, parallelization of partitioned queries, and cause queries to return query metrics:

private static FeedOptions queryOptions = new FeedOptions { MaxItemCount = -1, EnableCrossPartitionQuery = true, MaxDegreeOfParallelism = -1, PopulateQueryMetrics = true, EnableScanInQuery = true };

1. In Solution Explorer, click **Program.cs**.
2. On the **Program.cs** tab, scroll down to the **Worker** class. The **DoWork** method runs each of the queries using each of the collections in turn, and calls the **Process** method to handle the results. The queries are run twice over each collection, to reduce the effects of caching by Cosmos DB (the first execution might have to retrieve data from disk storage, whereas the second run could use cached data). The **Process** method iterates through the data returned by a query and displays the performance information. This performance information is either the elapsed time taken to run the query (if the **mode** parameter is ‘2’), or the detailed query metrics (if the **mode** parameter is ‘1’). The boolean **print** parameter specifies whether the contents of the documents should be output, for debugging purposes.
3. In Solution Explorer, click **App.config**.
4. On the **App.config** tab, enter the values for the following settings:
   * **EndpointUrl**. Replace **~URI~** with the **URI** you noted earlier from your Cosmos DB account.
   * **PrimaryKey**. Replace **~ PRIMARY KEY~** with the **PRIMARY KEY** you noted earlier from your Cosmos DB account.
   * **Database**. Leave this set to **Cricket**.
   * **Collections**. This is the set of collections you created in the database. Leave this set to the default list.
   * **NumWorkers**. Leave this set to **1**.
   * **TraceFolder**. The output of the **Process** method is written to files in this folder. Set this to **C:\OpsgilityTraining\**.
   * **PrintMode**. This setting specified whether the **Process** method should output each document retrieved. We are only interested in the metrics, so leave this set to **false**.
5. On the **Build** menu, click **Build Solution**, and wait until the build has succeeded.
6. On the **Debug** menu, click **Start Debugging**.
7. At the **Which mode?** prompt, type **2**, and wait for the app to finish.
8. In File Explorer, navigate to **C:\OpsgilityTraining**, and then double-click **Matches.txt**. This file contains a copy of the elapsed timings for the queries using the **Matches** non-partitioned and fully indexed collection. It should look similar to the following. Ignore the first elapsed time for each query. The second time is a base benchmark you can use for comparison with the queries run using the other collections:
9. Performance stats using collection Matches
10. =====================================================================
11. Query: Find T20 games for England at home, Elapsed Time: 710 ms
12. Query: Find T20 games for England at home, Elapsed Time: 56 ms
13. Query: Find T20 games for England away, Elapsed Time: 49 ms
14. Query: Find T20 games for England away, Elapsed Time: 50 ms
15. Query: Find ODIs played in England between England and Australia, Elapsed Time: 66 ms
16. Query: Find ODIs played in England between England and Australia, Elapsed Time: 52 ms
17. Query: Find all games played in Nottingham, Elapsed Time: 78 ms
18. Query: Find all games played in Nottingham, Elapsed Time: 54 ms
19. Query: Find all T20 games, Elapsed Time: 400 ms

Query: Find all T20 games, Elapsed Time: 360 ms

1. In File Explorer, double-click **MatchesInningsNotIndexed.txt**. This file contains the statistics for the queries run using the non-partitioned collection without the **innings** data indexed. The timings should be similar to those of the fully indexed partition; none of the queries actually retrieve **innings** data:
2. Performance stats using collection MatchesInningsNotIndexed
3. =====================================================================
4. Query: Find T20 games for England at home, Elapsed Time: 83 ms
5. Query: Find T20 games for England at home, Elapsed Time: 57 ms
6. Query: Find T20 games for England away, Elapsed Time: 49 ms
7. Query: Find T20 games for England away, Elapsed Time: 48 ms
8. Query: Find ODIs played in England between England and Australia, Elapsed Time: 52 ms
9. Query: Find ODIs played in England between England and Australia, Elapsed Time: 105 ms
10. Query: Find all games played in Nottingham, Elapsed Time: 52 ms
11. Query: Find all games played in Nottingham, Elapsed Time: 75 ms
12. Query: Find all T20 games, Elapsed Time: 425 ms

Query: Find all T20 games, Elapsed Time: 347 ms

1. In File Explorer, double-click **MatchesPartitionedByHomeTeam.txt**. This file contains the statistics for the queries run using the collection partitioned by the home team. Note that the query that searches for data for the away team takes significantly longer than searching by the home team. Also, the query that searches for games played between two teams can also take advantage of the partitioning structure; one of the teams is the home team:
2. Query: Find T20 games for England at home, Elapsed Time: 72 ms
3. Query: Find T20 games for England at home, Elapsed Time: 59 ms
4. Query: Find T20 games for England away, Elapsed Time: 321 ms
5. Query: Find T20 games for England away, Elapsed Time: 113 ms
6. Query: Find ODIs played in England between England and Australia, Elapsed Time: 51 ms
7. Query: Find ODIs played in England between England and Australia, Elapsed Time: 54 ms
8. Query: Find all games played in Nottingham, Elapsed Time: 92 ms
9. Query: Find all games played in Nottingham, Elapsed Time: 126 ms
10. Query: Find all T20 games, Elapsed Time: 446 ms

Query: Find all T20 games, Elapsed Time: 490 ms

1. In File Explorer, double-click **MatchesPartitionedByAwayTeam.txt**. This file contains the statistics for the queries run using the collection partitioned by the away team. This time, the query that fetches data using the away team runs faster than the query that specifies the home team. Again, the query that searches for games played between two teams can take advantage of the partitioning structure as one of the teams is the away team:
2. Performance stats using collection MatchesPartitionedByAwayTeam
3. =====================================================================
4. Query: Find T20 games for England at home, Elapsed Time: 123 ms
5. Query: Find T20 games for England at home, Elapsed Time: 125 ms
6. Query: Find T20 games for England away, Elapsed Time: 86 ms
7. Query: Find T20 games for England away, Elapsed Time: 87 ms
8. Query: Find ODIs played in England between England and Australia, Elapsed Time: 50 ms
9. Query: Find ODIs played in England between England and Australia, Elapsed Time: 50 ms
10. Query: Find all games played in Nottingham, Elapsed Time: 84 ms
11. Query: Find all games played in Nottingham, Elapsed Time: 88 ms
12. Query: Find all T20 games, Elapsed Time: 527 ms

Query: Find all T20 games, Elapsed Time: 594 ms

1. In File Explorer, double-click **MatchesPartitionedByMatchType.txt**. This file contains the statistics for the queries run using the collection partitioned by the match type. The performance of the query that finds matches by city does not appear to have improved much. The reason for this is that there are only three match types (T20, ODI, and Test). Partitioning across a field that has a small number of distinct values is not as beneficial as partitioning across a field that has a large number of values:
2. Performance stats using collection MatchesPartitionedByMatchType
3. =====================================================================
4. Query: Find T20 games for England at home, Elapsed Time: 73 ms
5. Query: Find T20 games for England at home, Elapsed Time: 58 ms
6. Query: Find T20 games for England away, Elapsed Time: 99 ms
7. Query: Find T20 games for England away, Elapsed Time: 48 ms
8. Query: Find ODIs played in England between England and Australia, Elapsed Time: 52 ms
9. Query: Find ODIs played in England between England and Australia, Elapsed Time: 51 ms
10. Query: Find all games played in Nottingham, Elapsed Time: 163 ms
11. Query: Find all games played in Nottingham, Elapsed Time: 148 ms
12. Query: Find all T20 games, Elapsed Time: 457 ms

Query: Find all T20 games, Elapsed Time: 475 ms

1. Close all open instances of Notepad.
2. In File Explorer, delete all the text files in the **C:\OpsgilityTraining** folder.
3. In Visual Studio 2017, on the **Debug** menu, click **Start Debugging**.
4. At the **Which mode?** prompt, type **1**, and wait for the app to finish. This mode generates comprehensive query metrics rather than the elapsed timings gathered previously. The elapsed times were a rather crude measure that includes many variables, such as the time taken to pass data across the network, and the performance of the client. They are useful as a guide, but the detailed query metrics provide a better view of how a query is run by Cosmos DB.
5. In File Explorer, double-click **Matches.txt**. This file should now contain the detailed metrics for the queries using the **Matches** non-partitioned and fully indexed collection. It should look similar to the following. As before, you can ignore the first set of stats for each query and use the second set as the base benchmark for comparison purposes. The key figures to look at are the **Request Charge** and the **Run Time (ms)** value in the **Scheduling Metrics** table:
6. Performance stats using collection Matches
7. =====================================================================
8. Query: Find T20 games for England at home
9. ---------------------------------------------------------------
10. [0, Retrieved Document Count : 87
11. Retrieved Document Size : 2,763,363 bytes
12. Output Document Count : 87
13. Output Document Size : 0 bytes
14. Index Utilization : 100.00 %
15. Total Query Execution Time : 12.31 milliseconds
16. Query Preparation Times
17. Query Compilation Time : 0.12 milliseconds
18. Logical Plan Build Time : 0.04 milliseconds
19. Physical Plan Build Time : 0.06 milliseconds
20. Query Optimization Time : 0.01 milliseconds
21. Index Lookup Time : 0.15 milliseconds
22. Document Load Time : 10.67 milliseconds
23. Runtime Execution Times
24. Query Engine Execution Time : 0.85 milliseconds
25. System Function Execution Time : 0.00 milliseconds
26. User-defined Function Execution Time : 0.00 milliseconds
27. Document Write Time : 0.16 milliseconds
28. Client Side Metrics
29. Retry Count : 0
30. Request Charge : 44.12 RUs
31. Partition Execution Timeline
32. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
33. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
34. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
35. â 0â 01:27:57.470579â01:27:57.618499â 87â 0â
36. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
37. Scheduling Metrics
38. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
39. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
40. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
41. â 0â 15.10â 147.51â 17.03â 164.54â 1â
42. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
43. ]
44. Query: Find T20 games for England at home
45. ---------------------------------------------------------------
46. [0, Retrieved Document Count : 87
47. Retrieved Document Size : 2,763,363 bytes
48. Output Document Count : 87
49. Output Document Size : 0 bytes
50. Index Utilization : 100.00 %
51. Total Query Execution Time : 12.07 milliseconds
52. Query Preparation Times
53. Query Compilation Time : 0.12 milliseconds
54. Logical Plan Build Time : 0.04 milliseconds
55. Physical Plan Build Time : 0.06 milliseconds
56. Query Optimization Time : 0.01 milliseconds
57. Index Lookup Time : 0.19 milliseconds
58. Document Load Time : 10.31 milliseconds
59. Runtime Execution Times
60. Query Engine Execution Time : 0.98 milliseconds
61. System Function Execution Time : 0.00 milliseconds
62. User-defined Function Execution Time : 0.00 milliseconds
63. Document Write Time : 0.14 milliseconds
64. Client Side Metrics
65. Retry Count : 0
66. Request Charge : 44.12 RUs
67. Partition Execution Timeline
68. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
69. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
70. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
71. â 0â 01:27:57.688399â01:27:57.773990â 87â 0â
72. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
73. Scheduling Metrics
74. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
75. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
76. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
77. â 0â 0.02â 85.59â 0.04â 85.63â 1â
78. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
79. ]
80. Query: Find T20 games for England away
81. ---------------------------------------------------------------
82. [0, Retrieved Document Count : 40
83. Retrieved Document Size : 1,278,504 bytes
84. Output Document Count : 40
85. Output Document Size : 0 bytes
86. Index Utilization : 100.00 %
87. Total Query Execution Time : 5.84 milliseconds
88. Query Preparation Times
89. Query Compilation Time : 0.16 milliseconds
90. Logical Plan Build Time : 0.04 milliseconds
91. Physical Plan Build Time : 0.06 milliseconds
92. Query Optimization Time : 0.01 milliseconds
93. Index Lookup Time : 0.29 milliseconds
94. Document Load Time : 4.53 milliseconds
95. Runtime Execution Times
96. Query Engine Execution Time : 0.36 milliseconds
97. System Function Execution Time : 0.00 milliseconds
98. User-defined Function Execution Time : 0.00 milliseconds
99. Document Write Time : 0.05 milliseconds
100. Client Side Metrics
101. Retry Count : 0
102. Request Charge : 22.07 RUs
103. Partition Execution Timeline
104. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
105. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
106. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
107. â 0â 01:27:57.798422â01:27:57.887827â 40â 0â
108. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
109. Scheduling Metrics
110. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
111. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
112. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
113. â 0â 0.02â 89.40â 0.05â 89.45â 1â
114. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
115. ]
116. Query: Find T20 games for England away
117. ---------------------------------------------------------------
118. [0, Retrieved Document Count : 40
119. Retrieved Document Size : 1,278,504 bytes
120. Output Document Count : 40
121. Output Document Size : 0 bytes
122. Index Utilization : 100.00 %
123. Total Query Execution Time : 5.67 milliseconds
124. Query Preparation Times
125. Query Compilation Time : 0.12 milliseconds
126. Logical Plan Build Time : 0.04 milliseconds
127. Physical Plan Build Time : 0.12 milliseconds
128. Query Optimization Time : 0.01 milliseconds
129. Index Lookup Time : 0.15 milliseconds
130. Document Load Time : 4.61 milliseconds
131. Runtime Execution Times
132. Query Engine Execution Time : 0.34 milliseconds
133. System Function Execution Time : 0.00 milliseconds
134. User-defined Function Execution Time : 0.00 milliseconds
135. Document Write Time : 0.05 milliseconds
136. Client Side Metrics
137. Retry Count : 0
138. Request Charge : 22.07 RUs
139. Partition Execution Timeline
140. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
141. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
142. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
143. â 0â 01:27:57.977398â01:27:58.031464â 40â 0â
144. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
145. Scheduling Metrics
146. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
147. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
148. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
149. â 0â 0.02â 54.06â 0.04â 54.10â 1â
150. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
151. ]
152. Query: Find ODIs played in England between England and Australia
153. ---------------------------------------------------------------
154. [0, Retrieved Document Count : 35
155. Retrieved Document Size : 2,558,870 bytes
156. Output Document Count : 35
157. Output Document Size : 0 bytes
158. Index Utilization : 100.00 %
159. Total Query Execution Time : 8.41 milliseconds
160. Query Preparation Times
161. Query Compilation Time : 0.16 milliseconds
162. Logical Plan Build Time : 0.04 milliseconds
163. Physical Plan Build Time : 0.12 milliseconds
164. Query Optimization Time : 0.01 milliseconds
165. Index Lookup Time : 0.20 milliseconds
166. Document Load Time : 7.15 milliseconds
167. Runtime Execution Times
168. Query Engine Execution Time : 0.36 milliseconds
169. System Function Execution Time : 0.00 milliseconds
170. User-defined Function Execution Time : 0.00 milliseconds
171. Document Write Time : 0.06 milliseconds
172. Client Side Metrics
173. Retry Count : 0
174. Request Charge : 30.86 RUs
175. Partition Execution Timeline
176. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
177. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
178. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
179. â 0â 01:27:58.088415â01:27:58.138720â 35â 0â
180. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
181. Scheduling Metrics
182. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
183. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
184. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
185. â 0â 0.03â 50.30â 0.06â 50.36â 1â
186. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
187. ]
188. Query: Find ODIs played in England between England and Australia
189. ---------------------------------------------------------------
190. [0, Retrieved Document Count : 35
191. Retrieved Document Size : 2,558,870 bytes
192. Output Document Count : 35
193. Output Document Size : 0 bytes
194. Index Utilization : 100.00 %
195. Total Query Execution Time : 8.62 milliseconds
196. Query Preparation Times
197. Query Compilation Time : 0.11 milliseconds
198. Logical Plan Build Time : 0.09 milliseconds
199. Physical Plan Build Time : 0.09 milliseconds
200. Query Optimization Time : 0.01 milliseconds
201. Index Lookup Time : 0.17 milliseconds
202. Document Load Time : 7.40 milliseconds
203. Runtime Execution Times
204. Query Engine Execution Time : 0.42 milliseconds
205. System Function Execution Time : 0.00 milliseconds
206. User-defined Function Execution Time : 0.00 milliseconds
207. Document Write Time : 0.05 milliseconds
208. Client Side Metrics
209. Retry Count : 0
210. Request Charge : 30.86 RUs
211. Partition Execution Timeline
212. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
213. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
214. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
215. â 0â 01:27:58.198436â01:27:58.250340â 35â 0â
216. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
217. Scheduling Metrics
218. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
219. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
220. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
221. â 0â 0.04â 51.90â 0.07â 51.97â 1â
222. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
223. ]
224. Query: Find all games played in Nottingham
225. ---------------------------------------------------------------
226. [0, Retrieved Document Count : 38
227. Retrieved Document Size : 3,230,737 bytes
228. Output Document Count : 38
229. Output Document Size : 0 bytes
230. Index Utilization : 100.00 %
231. Total Query Execution Time : 13.76 milliseconds
232. Query Preparation Times
233. Query Compilation Time : 0.08 milliseconds
234. Logical Plan Build Time : 0.02 milliseconds
235. Physical Plan Build Time : 0.04 milliseconds
236. Query Optimization Time : 0.00 milliseconds
237. Index Lookup Time : 0.06 milliseconds
238. Document Load Time : 12.67 milliseconds
239. Runtime Execution Times
240. Query Engine Execution Time : 0.63 milliseconds
241. System Function Execution Time : 0.00 milliseconds
242. User-defined Function Execution Time : 0.00 milliseconds
243. Document Write Time : 0.05 milliseconds
244. Client Side Metrics
245. Retry Count : 0
246. Request Charge : 35.35 RUs
247. Partition Execution Timeline
248. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
249. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
250. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
251. â 0â 01:27:58.301464â01:27:58.358922â 38â 0â
252. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
253. Scheduling Metrics
254. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
255. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
256. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
257. â 0â 0.03â 57.45â 0.06â 57.51â 1â
258. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
259. ]
260. Query: Find all games played in Nottingham
261. ---------------------------------------------------------------
262. [0, Retrieved Document Count : 38
263. Retrieved Document Size : 3,230,737 bytes
264. Output Document Count : 38
265. Output Document Size : 0 bytes
266. Index Utilization : 100.00 %
267. Total Query Execution Time : 14.13 milliseconds
268. Query Preparation Times
269. Query Compilation Time : 0.18 milliseconds
270. Logical Plan Build Time : 0.03 milliseconds
271. Physical Plan Build Time : 0.05 milliseconds
272. Query Optimization Time : 0.00 milliseconds
273. Index Lookup Time : 0.13 milliseconds
274. Document Load Time : 12.49 milliseconds
275. Runtime Execution Times
276. Query Engine Execution Time : 0.89 milliseconds
277. System Function Execution Time : 0.00 milliseconds
278. User-defined Function Execution Time : 0.00 milliseconds
279. Document Write Time : 0.06 milliseconds
280. Client Side Metrics
281. Retry Count : 0
282. Request Charge : 35.35 RUs
283. Partition Execution Timeline
284. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
285. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
286. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
287. â 0â 01:27:58.423408â01:27:58.519234â 38â 0â
288. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
289. Scheduling Metrics
290. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
291. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
292. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
293. â 0â 0.02â 95.83â 0.03â 95.85â 1â
294. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
295. ]
296. Query: Find all T20 games
297. ---------------------------------------------------------------
298. [0, Retrieved Document Count : 1,661
299. Retrieved Document Size : 53,600,914 bytes
300. Output Document Count : 1,661
301. Output Document Size : 0 bytes
302. Index Utilization : 100.00 %
303. Total Query Execution Time : 184.17 milliseconds
304. Query Preparation Times
305. Query Compilation Time : 0.10 milliseconds
306. Logical Plan Build Time : 0.03 milliseconds
307. Physical Plan Build Time : 0.08 milliseconds
308. Query Optimization Time : 0.00 milliseconds
309. Index Lookup Time : 0.10 milliseconds
310. Document Load Time : 170.40 milliseconds
311. Runtime Execution Times
312. Query Engine Execution Time : 11.04 milliseconds
313. System Function Execution Time : 0.00 milliseconds
314. User-defined Function Execution Time : 0.00 milliseconds
315. Document Write Time : 1.94 milliseconds
316. Client Side Metrics
317. Retry Count : 0
318. Request Charge : 792.47 RUs
319. Partition Execution Timeline
320. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
321. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
322. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
323. â 0â 01:27:58.528383â01:27:59.267309â 1661â 0â
324. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
325. Scheduling Metrics
326. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
327. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
328. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
329. â 0â 0.01â 738.92â 0.02â 738.94â 1â
330. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ
331. ]
332. Query: Find all T20 games
333. ---------------------------------------------------------------
334. [0, Retrieved Document Count : 1,661
335. Retrieved Document Size : 53,600,914 bytes
336. Output Document Count : 1,661
337. Output Document Size : 0 bytes
338. Index Utilization : 100.00 %
339. Total Query Execution Time : 185.51 milliseconds
340. Query Preparation Times
341. Query Compilation Time : 0.10 milliseconds
342. Logical Plan Build Time : 0.03 milliseconds
343. Physical Plan Build Time : 0.04 milliseconds
344. Query Optimization Time : 0.00 milliseconds
345. Index Lookup Time : 0.07 milliseconds
346. Document Load Time : 170.14 milliseconds
347. Runtime Execution Times
348. Query Engine Execution Time : 12.76 milliseconds
349. System Function Execution Time : 0.00 milliseconds
350. User-defined Function Execution Time : 0.00 milliseconds
351. Document Write Time : 1.89 milliseconds
352. Client Side Metrics
353. Retry Count : 0
354. Request Charge : 792.47 RUs
355. Partition Execution Timeline
356. ââââââââââââââ¬âââââââââââââââââ¬ââââââââââââââââ¬ââââââââââââââââââââ¬ââââââââââââ
357. âPartition IdâStart Time (UTC)âEnd Time (UTC) âNumber of DocumentsâRetry Countâ
358. ââââââââââââââ¼âââââââââââââââââ¼ââââââââââââââââ¼ââââââââââââââââââââ¼ââââââââââââ¤
359. â 0â 01:27:59.284387â01:28:00.926713â 1661â 1â
360. ââââââââââââââ´âââââââââââââââââ´ââââââââââââââââ´ââââââââââââââââââââ´ââââââââââââ
361. Scheduling Metrics
362. ââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬âââââââââââââââââââââ¬ââââââââââââââââââââââ
363. âPartition IdâResponse Time (ms) âRun Time (ms) âWait Time (ms) âTurnaround Time (ms)âNumber of Preemptionsâ
364. ââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼âââââââââââââââââââââ¼ââââââââââââââââââââââ¤
365. â 0â 0.01â 1642.32â 0.02â 1642.35â 1â
366. ââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´âââââââââââââââââââââ´ââââââââââââââââââââââ

]

1. In File Explorer, double-click **MatchesPartitionedByHomeTeam.txt**. Compare the values of the **Run Time (ms)** for each query of those in the **Matches.txt** file. For the queries **Find T20 games for England at home** and **Find ODIs played in England between England and Australia**, the timings should be faster due to the way in which the data is partitioned
2. In File Explorer, double-click **MatchesPartitionedByAwayTeam.txt**. The partitioning should have decreased the run time required for the query **Find T20 games for England away**, although the queries **Find T20 games for England at home** and **Find ODIs played in England between England and Australia** should have slowed down.

**Note:** Remember that when you specify a partition key, you define a **logical** partition. Internally, Cosmos DB can choose to store several logical partitions in the same **physical**partition, depending on the size and number of logical partitions. The set of logical partitions in a physical partition is referred to as the **partition key range**. When the data was partitioned by home team or away team, it was all stored in the same physical partition. As these logical partitions grow, and new logical partitions are added, then Cosmos DB may choose to add more physical partitions and reorganize the data. There are a larger number of cities than international cricketing countries, so partitioning by city created a larger number of logical partitions, and Cosmos DB decided to distribute these logical partitions across multiple physical partitions. Partitioning by city improves the response time of the query that finds all matches played in Nottingham.

1. In File Explorer, double-click **MatchesPartitionedByMatchType.txt**. In this case, the data is all stored in the same physical partition. Look at the metrics for the query **Find all T20 games**. As described earlier, partitioning by match type does not offer much advantage in performance due to the low number of distinct values.
2. Close all open instances of Notepad.
3. Close Visual Studio 2017.

Task 4: Examine the Physical to Logical Mapping of Partitions

1. In the Azure portal, navigate to your Cosmos DB instance, under **MONITORING** click **Metrics**.
2. On the **Metrics** blade, on the **Storage** tab, in the **Database(s)** drop-down list, click **Cricket**.
3. In the **Collection(s)** drop-down list, click **MatchesPartitionedByHomeTeam**.
4. In the **Data + Index storage consumed per partition key range** pane, click the bar for the only partition that is displayed. This partition should be named **partition 0**.
5. In the **Showing partition keys for partition 0** pane, you should see the top three partition keys defining the logical partitions in partition 0. These should be labelled **England**, **Australia**, and **India**.

Summary

In this exercise, you evaluated the impact of different partitioning and indexing strategies on the performance of some common operations in Cosmos DB.



 2018 SKILL ME UP AND OPSGILITY, LLC. ALL RIGHTS RESERVED

* **GET STARTED**
* [**HOME**](https://skillmeup.com/)
* [**ABOUT SKILLMEUP**](https://skillmeup.com/Home/About)
* [**TERMS OF USE**](https://skillmeup.com/Home/TermsAndConditions)
* [**REGISTER**](https://skillmeup.com/Account/Register)
* [**LOGIN**](https://skillmeup.com/Account/Login)
* [**HELP AND TUTORIALS**](https://skillmeup.com/tutorials)
* **COURSES AND LABS**
* [**ALL LEARNING PATHS**](https://skillmeup.com/course/SearchTrackOrCourse?onlyLabs=false&onlyCourses=false&onlyTracks=true)
* [**ALL COURSES**](https://skillmeup.com/course/SearchTrackOrCourse?onlyLabs=false&onlyCourses=true&onlyTracks=false)
* [**ALL LABS**](https://skillmeup.com/course/SearchTrackOrCourse?onlyLabs=true&onlyCourses=false&onlyTracks=false)
* [**UPCOMING COURSES AND LABS**](https://skillmeup.com/TrainingSchedule/RoadMap)
* **LEARN MORE**
* [**CONTACT US**](https://skillmeup.com/ContactUs)
* [**OPEN A SUPPORT TICKET**](https://skillmeup.com/SupportTicket)
* [**AUTHORS**](https://skillmeup.com/Author)
* [**BECOME AN AUTHOR**](https://skillmeup.com/Author/SkillMeUpAuthor)
* [**PRIVACY**](https://skillmeup.com/home/termsandconditions)