# 了解 DocumentDB 中的存储和可预测性能的预配信息

DocumentDB 是一个全面托管的，面向可扩展文档的针对 JSON 文档的 NoSQL 数据库服务。使用 DocumentDB，你无需租用虚拟机、部署软件、监控数据库或担心灾难恢复。DocumentDB 由 Microsoft 工程师操作和持续监视，以提供一流的可用性、性能和数据保护。

你可以通过在 [Azure 门户](https://portal.azure.cn/)中[创建数据库帐户](/documentation/articles/documentdb-create-account)来开始使用 DocumentDB。DocumentDB 使用固态硬盘 (SSD) 支持的存储和吞吐量的单位数来提供。这些单位数是通过在数据库帐户中创建数据库集合来设置的。每个集合都有预留的吞吐量。如果应用程序的吞吐量需求改变，你可以通过为每个集合设置[性能级别](/documentation/articles/documentdb-performance-levels)来动态更改此预留值。

当你的应用程序超出一个或多个集合的性能级别时，将按每个集合限制请求。这意味着一些应用程序请求可能成功，而另一些则可能受限制。

本文概述了可用于管理容量和计划数据存储的资源和指标。

## 数据库帐户和管理资源

作为 Azure 用户，你可以设置一个或多个 DocumentDB 数据库帐户。每个数据库帐户都有一个管理资源配额，包括数据库、用户和权限。这些资源的使用需要遵循[限制和配额](/documentation/articles/documentdb-limits)。如果你需要更多管理资源，请[与支持部门联系](/documentation/articles/documentdb-increase-limits)。

Azure 门户提供了用于监视数据库帐户、数据库和集合的使用情况指标。有关详细信息，请参阅[监视 DocumentDB 帐户](/documentation/articles/documentdb-monitor-accounts)。

## 数据库

一个 DocumentDB 数据库可以包含几乎无限量的文档存储空间，这些存储空间被分到各个集合中。集合提供性能隔离 - 每个集合都设置有吞吐量，此吞吐量不会与同一数据库或帐户中的其他集合共享。DocumentDB 数据库的大小是灵活的 – 从几个 GB 到几个 TB 的 SSD 支持的文档存储和设置的吞吐量。与传统的 RDBMS 数据库不同，DocumentDB 中的数据库不局限于一台计算机，而是可以跨越多个计算机或群集。

使用 DocumentDB，可根据你扩展应用程序的需求创建多个集合或数据库，或同时创建两者。

## 数据库集合

每个 DocumentDB 数据库可以包含一个或多个集合。集合是用于文档存储和处理的高可用的数据分区。每个集合可以存储具有各种架构的文档。DocumentDB 的自动索引和查询功能允许你轻松筛选并检索文档。集合提供了文档存储和查询执行的范围。集合还是它所包含的所有文档的事务域。根据集合的指定的性能级别为其分配吞吐量。可通过 Azure 门户或一个 SDK 来动态调整此吞吐量。

DocumentDB 自动将集合分区到一个或多个物理服务器。创建集合时，你可以指定预配吞吐量（根据每秒的请求单位数）和分区键属性。DocumentDB 会使用此属性的值将文档分布于分区和路由请求（例如查询）之间。分区键值还可作为存储过程和触发器的事务边界。每个集合都有该集合特定的保留吞吐量，且不会与相同帐户中的其他集合共享。因此，你可以在存储和吞吐量方面扩大你的应用程序。

可以通过 [Azure 门户](https://portal.azure.com/)或任意一个 [DocumentDB SDK](/documentation/articles/documentdb-sdk-dotnet) 创建集合。

## 请求单位数和数据库操作

在创建集合时，你可以根据每秒的请求单位 (RU) 数预留吞吐量。你可以把**请求单位**视为执行应用程序所请求的各种数据库操作和服务所需的资源的一个计量单位，而无需考虑和管理硬件资源。不管集合中存储的项数有多少，或者同时运行的并发请求数有多少，读取 1 KB 的文档都要使用 1 个 RU。所有针对 DocumentDB 的请求，包括复杂的操作如 SQL 查询，都有一个可预测的 RU 值，该值可在开发时间确定。如果你知道用于支持你的应用程序的文档的大小以及每个操作（读取、写入和查询）的频率，那么你可以设置恰到好处的吞吐量来满足应用程序需求，并且随着性能需求的变化增大和减小数据库的规模。

每个集合可以预留每秒 100 的倍数请求单位的数据块的吞吐量，每秒从几百到几百万个请求单位。可以在集合的整个生命周期内调整设置的吞吐量，以适应不断变化的应用程序的处理需求和访问模式。有关详细信息，请参阅 [DocumentDB 性能级别](/documentation/articles/documentdb-performance-levels)。

[AZURE.IMPORTANT] 集合是计费实体。集合的成本由集合的设置的吞吐量（以每秒的请求单位数来衡量）和总占用存储空间（以千兆字节为单位）决定。

一个特定操作如插入、删除、查询或存储过程执行会消耗多少个请求单位？ 请求单位是请求处理成本的规范化的度量。读取 1 kB 文档需要 1 个 RU，但是插入、替换或删除同一文档的请求却要占用服务的更多处理，因此需要更多请求单位。来自服务的每个响应包括一个自定义标头 (x-ms-request-charge)，其中报告了请求所要使用的请求单位数。此标头也可通过 [SDK](/documentation/articles/documentdb-sdk-dotnet) 防问。在 .NET SDK 中，RequestCharge 是 ResourceResponse 对象的属性。

[AZURE.NOTE] 用于 1 KB 文档的 1 个请求单位基准与使用会话一致性的简单的文档 GET 命令相对应。

影响针对某个 DocumentDB 数据库帐户的操作所使用的请求单位数的因素有多个。这些因素包括：

* 文档大小。随着文档大小的增加，用来读取或写入数据的单位数也随之增加。
* 属性计数。假设默认为所有属性创建索引，用于写入文档的单位数将随着属性计数的增加而增加。
* 数据一致性。当使用“强”或“有限过时”的数据一致性级别时，将占用更多单位数来读取文档。
* 已创建索引的属性。每个集合的索引策略都可确定默认情况下要进行索引的属性类别。通过限制已创建索引的属性的数量，可以减少请求单位的消耗。
* 文档索引。默认情况下，将自动为每个文档创建索引，如果你选择不为其中一些文档创建索引，则将占用更少的请求单位。

有关详细信息，请参阅 [DocumentDB 请求单位](/documentation/articles/documentdb-request-units)。

例如，下表显示了三种不同的文档大小（1 KB、4 KB 和 64 KB）和两个不同的性能级别（500 次读取/秒 + 100 次写入/秒和 500 次读取/秒 + 500 次写入/秒）所要设置的请求单位数。数据一致性配置为会话级别，索引策略设置为无。

<tbody>  
 <tr>  
 <td valign="top"><p><strong>文档大小</strong></p></td>  
 <td valign="top"><p><strong>读取数/秒</strong></p></td>  
 <td valign="top"><p><strong>写入数/秒</strong></p></td>  
 <td valign="top"><p><strong>请求单位</strong></p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>1 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>100</p></td>  
 <td valign="top"><p>(500 × 1) + (100 × 5) = 1000 RU/秒</p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>1 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>(500 × 5) + (100 × 5) = 3,000 RU/秒</p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>4 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>100</p></td>  
 <td valign="top"><p>(500 × 1.3) + (100 × 7) = 1,350 RU/秒</p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>4 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>(500 × 1.3) + (500 × 7) = 4,150 RU/秒</p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>64 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>100</p></td>  
 <td valign="top"><p>(500 × 10) + (100 × 48) = 9,800 RU/秒</p></td>  
 </tr>  
 <tr>  
 <td valign="top"><p>64 KB</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>500</p></td>  
 <td valign="top"><p>(500 × 10) + (500 × 48) = 29,000 RU/秒</p></td>  
 </tr>  
</tbody>

查询、存储过程和触发器是根据所执行的操作的复杂性来使用请求单位的。在开发应用程序时，检查请求费用标头，以更好地了解每个操作消耗请求单位容量的方式。

## 一致性级别的选择和吞吐量

默认一致性级别的选择会影响吞吐量和延迟。你可以编程方式和通过 Azure 门户设置默认的一致性级别。你还可以重写单个请求的一致性级别。默认情况下采用会话一致性级别，该级别提供单向读取/写入和读取最新写入内容的保证。会话一致性非常适用于以用户为中心的应用程序，可提供一致性和性能的完美平衡。

有关在 Azure 门户上更改一致性级别的说明，请参阅[如何管理 DocumentDB 帐户](/documentation/articles/documentdb-manage-account#consistency)。或者，有关一致性级别的详细信息，请参阅[使用一致性级别](/documentation/articles/documentdb-consistency-levels)。

## 设置的文档存储和索引开销

DocumentDB 支持创建单个分区和已分区的集合。DocumentDB 中的每个分区支持高达 10 GB 的 SSD 支持的存储空间。10 GB 文档存储空间包含文档和索引的存储。默认情况下，DocumentDB 集合配置为自动为所有文档创建索引，且没有明确要求任何二级索引或架构。根据使用 DocumentDB 的应用程序，索引开销通常介于 2-20%。DocumentDB 使用的索引技术可以确保无论属性值是多少，索引开销都不会超过具有默认设置的文档大小的 80% 以上。

默认情况下，DocumentDB 为所有文档自动创建索引。但是，如果你想要精细调整索引开销，那么你可以在插入或替换文档时选择不对某些文档创建索引，如 [DocumentDB 索引策略](/documentation/articles/documentdb-indexing-policies)中所述。你可以将 DocumentDB 集合配置为不对集合中的所有文档创建索引。你还可以将 DocumentDB 集合配置为有选择的只对 JSON 文档的某些具有通配符的属性或路径创建索引，如[配置集合的索引策略](/documentation/articles/documentdb-indexing-policies#configuring-the-indexing-policy-of-a-collection)中所述。不对某些属性或文档创建索引还可以提高写入吞吐量 - 这意味着将消耗更少的请求单位。 ## 后续步骤 有关在 Azure 门户中监视性能级别的说明，请参阅[监视 DocumentDB 帐户](/documentation/articles/documentdb-monitor-accounts)。

有关选择集合的性能级别的详细信息，请参阅 [DocumentDB 中的性能级别](/documentation/articles/documentdb-performance-levels)。