# Data Modeling for the SQL API



Leonard Lobel
CTO, SLEEK TECHNOLOGIES
lennilobel.wordpress.com

### What is a Document Database?

### JSON documents

Hierarchical key-value pairs

```
"id": 10,
"business id": "PK6aS
                        "name": "Cleme: {
                        "username": "M
                                             "id": "1242160000000072038",
"full_address": "400
                                             "description": "3",
                        "email": "Rev.
"hours": {},
                        "address": {
                                             "website": "3",
"open": true,
                                             "numberOfEmployees": "3",
                          "street": "K
"categories": [
                          "suite": "Su
                                             "phone": "3",
  "Burgers",
                                             "name": "account3",
                          "citv": "Leb
  "Fast Food",
                          "zipcode": "
                                             "shippingAddress": {
  "Restaurants"
                                                  "country": "3",
                          "geo": {
],
                                                 "stateOrProvidence": "3",
                            "lat": "-3
"city": "Homestead",
                                                  "city": "3",
                            "lna": "57
"review count": 5,
                                                 "postalCode": "3",
"name": "McDonald's"
                                                  "street1": "3"
"neighborhoods": [
                        "phone": "024-
  "Homestead"
                                             "billingAddress": {
                        "website": "am
                                                  "country": "3",
                        "company": {
"longitude": -79.9100
                                                  "stateOrProvidence": "3",
                          "name": "Hoe
"state": "PA",
                                                 "city": "3",
                          "catchPhrase
"stars": 2,
                                                 "postalCode": "3",
                          "bs": "targe
                                                  "street1": "3"
"latitude": 40,412086
"attributes": {
  "Take-out": true,
  "Wi-Fi": "free",
  "Drive-Thru": true,
                                       "date": "07/04/2016",
  "Good For": {
                                       "time": "09:30",
    "dessert": false,
                                       "from": "New York",
                                       "to": "Chicago",
    "latenight": false,
                                       "seat": "57B"
    "lunch": false,
    "dinner": false.
                                    "passenger": {
    "breakfast": false,
                                       "name": "John Smith"
    "brunch": false
                                   },
  },
                                   "price": 1234.25,
  "Caters": false.
                                   "comments": ["Lunch & dinner incl.", "\"Have a nice dav!\""]
  "Noise Level": "average",
  "Takes Reservations": false
  "Dolivory", falco
```

### What is a Document Database?

JSON documents

Hierarchical key-value pairs

Two API choices

1) SQL API (aka core API)

2) MongoDB API

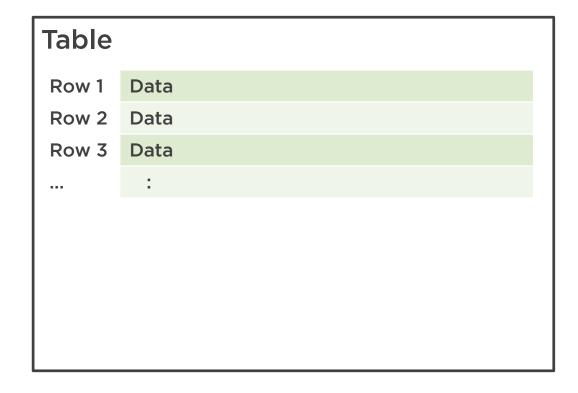


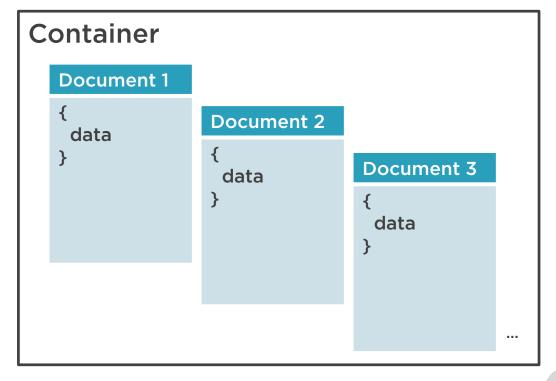
**Relational Database** 

**Document Database** 



| Relational Database | Document Database |  |
|---------------------|-------------------|--|
| Rows                | Documents         |  |







| Relational Database | Document Database |  |
|---------------------|-------------------|--|
| Rows                | Documents         |  |
| Columns             | Properties        |  |

| Col1 | Col2 | Col3 | Col4 |
|------|------|------|------|
| Data | Data | Data | Data |
| Data | Data | Data | Data |
| Data | Data | Data | Data |

### **Document 1**

```
{
  "prop1": data,
  "prop2": data,
  "prop3": data,
  "prop4": data
}
```

### Document 2

```
{
  "prop1": data,
  "prop2": data,
  "prop3": data,
  "prop4": data
}
```

### Document 3

```
{
  "prop1": data,
  "prop2": data,
  "prop3": data,
  "prop4": data
}
```



| Relational Database    | Document Database |  |
|------------------------|-------------------|--|
| Rows                   | Documents         |  |
| Columns                | Properties        |  |
| Strongly typed schemas | No defined schema |  |

| ID | Name       | IsActive | Dob       |
|----|------------|----------|-----------|
| 1  | Dan Smith  | True     | 8/30/1964 |
| 2  | Sue Jones  | False    | 2/18/2002 |
| 3  | Adam Stark | True     | 7/13/1987 |

```
Tocument 1

{
    "id": 1,
    "name": "Dan Smith",
    "isActive": true,
    "dob": "1964-30-08"
}
```



| Relational Database    | Document Database |
|------------------------|-------------------|
| Rows                   | Documents         |
| Columns                | Properties        |
| Strongly typed schemas | No defined schema |

| ID | Name       | IsActive | Dob       |
|----|------------|----------|-----------|
| 1  | Dan Smith  | True     | 8/30/1964 |
| 2  | Sue Jones  | False    | 2/18/2002 |
| 3  | Adam Stark | True     | 7/13/1987 |

| Document 1   |
|--|
| <pre>{   "id": 1,   "name": "Dan Smith",   "isActive": true,   "dob": "1964-30-08" }</pre> |

```
Tocument 2

{
    "id": 2,
    "fullName": "Sue Jones",
    "dob": "2002-02-18"
}
```

```
Pocument 3

{
    "id": 3,
    "fullName":
    {
        "first": "Adam",
        "last": "Stark"
    },
    "isActive": true,
    "dob": "2015-04-19"
}
```

| Relational Database    | Document Database |
|------------------------|-------------------|
| Rows                   | Documents         |
| Columns                | Properties        |
| Strongly typed schemas | No defined schema |

| ID | Name       | IsActive | Dob       |
|----|------------|----------|-----------|
| 1  | Dan Smith  | True     | 8/30/1964 |
| 2  | Sue Jones  | False    | 2/18/2002 |
| 3  | Adam Stark | True     | 7/13/1987 |

# Tocument 1 { "id": 1, "name": "Dan Smith", "isActive": true, "dob": "1964-30-08", "type": "user"

```
Tocument 2

{
    "id": 2,
    "fullName": "Sue Jones",
    "dob": "2002-02-18",
    "type": "user"
}
```

```
Pocument 3

{
    "id": 3,
    "fullName":
    {
        "first": "Adam",
        "last": "Stark"
    },
    "isActive": true,
    "dob": "2015-04-19",
    "type": "user"
}
```

| Relational Database    | Document Database |
|------------------------|-------------------|
| Rows                   | Documents         |
| Columns                | Properties        |
| Strongly typed schemas | No defined schema |

| ID | Name       | IsActive | Dob       |
|----|------------|----------|-----------|
| 1  | Dan Smith  | True     | 8/30/1964 |
| 2  | Sue Jones  | False    | 2/18/2002 |
| 3  | Adam Stark | True     | 7/13/1987 |

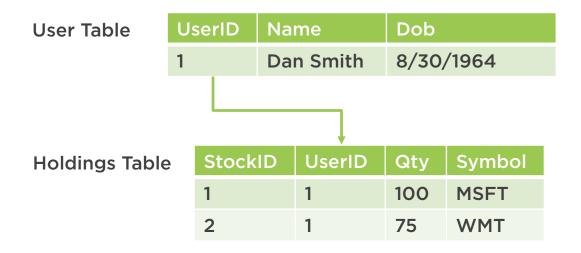
| Document 1           |
|----------------------|
| {                    |
| "id": 1,             |
| "name": "Dan Smith", |
| "isActive": true,    |
| "dob": "1964-30-08", |
| "type": "user",      |
| "version": 1         |
|                      |

```
Document 2

{
    "id": 2,
    "fullName": "Sue Jones",
    "dob": "2002-02-18",
    "type": "user",
    "version": 2
}
```

```
Pocument 3

{
    "id": 3,
    "fullName":
    {
        "first": "Adam",
        "last": "Stark"
    },
    "isActive": true,
    "dob": "2015-04-19",
    "type": "user",
    "version": 3
}
```





#### **Document**

```
{
  "postid": "1",
  "title": "My blog post",
  "body": "Post content...",
  "comments": [
    "comment #1",
    "comment #2",
    "comment #3",
    "comment #4",
    :
    "comment #1598873",
    :
```



#### **Document**

```
{
  "postid": "1",
  "title": "My blog post",
  "body": "Post content..."
}
```

```
Compant

Document

{
    "postid": "1",
    "comment": "comment #3"
    }
}
```

#### **Document**

```
{
  "postid": "1",
  "title": "My blog post",
  "body": "Post content...",
  "comments": [
      "comment #1",
      "comment #2",
      :
      "comment #100"
]
}
```



#### **Document**

```
"id": "1",
"prop1": "Typically read",
"prop2": "Typically read",
"prop3": "Typically read",
"prop4": "Typically read",
"prop5": "Typically read",
"prop6": "Typically read",
"prop7": "Typically read",
"prop8": "Typically read",
"prop9": "Typically read",
"prop10": "Typically read",
"prop200": "Typically read",
"prop201": "Typically updated",
"prop202": "Typically updated",
"prop203": "Typically updated",
"prop204": "Typically updated",
"prop205": "Typically updated"
```

#### **Document**

```
"id": "1-1",
  "prop1": "Typically read",
  "prop2": "Typically read",
  "prop3": "Typically read",
  "prop4": "Typically read",
  "prop5": "Typically read",
  "prop6": "Typically read",
  "prop7": "Typically read",
  "prop8": "Typically read",
  "prop9": "Typically read",
  "prop10": "Typically read",
  :
  "prop200": "Typically read"
}
```

#### **Document**

```
{
  "id": "1-2",
  "prop201": "Typically updated",
  "prop202": "Typically updated",
  "prop203": "Typically updated",
  "prop204": "Typically updated",
  "prop205": "Typically updated"
}
```

```
Document
 "code": "BC",
 "desc": "Black chair",
 "info": "This comfortable seat is..."
   Document
    "code": "RT",
    "desc": "Red table".
    "info": "This sturdy table top is..."
      Document
        "code": "YC",
        "desc": "Yellow clock",
        "info": "This distinctive timepiece is..."
```

```
Document
 "name": "Dan Smith",
 "orderDate": "2015-30-03",
 "details": Γ
   { "qty": 5, "code": "BC" },
   { "qty": 1, "code": "RT" },
   { "qty": 2, "code": "YC" }
      Document
        "name": "Sue Jones,
        "orderDate": "2015-16-04",
        "details": Γ
          { "qty": 1, "code": "PL" },
           { "atv": 3, "code": "YC" }
             Document
              "name": "Adam Stark",
              "orderDate": "2015-06-05",
              "details": Γ
                 { "qty": 2, "code": "BC" },
                 { "qty": 1, "code": "YC" }
```

```
Document
                                                                  Document
 "code": "BC",
                                                                    "name": "Dan Smith",
 "desc": "Black chair",
                                                                    "orderDate": "2015-30-03",
 "info": "This comfortable seat is..."
                                                                   "details": Γ
                                                                      { "qty": 5, "code": "BC", "desc": "Black chair" },
   Document
                                                                      { "qty": 1, "code": "RT", "desc": "Red table" },
                                                                      { "qty": 2, "code": "YC", "desc": "Yellow clock" }
    "code": "RT".
                                                                         Document
    "desc": "Red table",
    "info": "This sturdy table top is..."
                                                                           "name": "Sue Jones,
      Document
                                                                           "orderDate": "2015-16-04",
                                                                           "details": Γ
                                                                             { "qty": 1, "code": "PL", "desc": "Purple lamp" },
        "code": "YC",
                                                                             { "qty": 3, "code": "YC", "desc": "Yellow clock" }
       "desc": "Yellow clock",
       "info": "This distinctive timepiece is..."
                                                                                Document
                                                                                 "name": "Adam Stark",
                                                                                 "orderDate": "2015-06-05",
                             Change Feed
                                                                                 "details": [
                                                                                    { "qty": 2, "code": "BC", "desc": "Black chair" },
```

https://docs.microsoft.com/en-us/azure/cosmos-db/change-feed

{ "qty": 1, "code": "YC", "desc": "Yellow clock" }

### Data Migration Tool

### Open source project

Microsoft Download Center

http://www.microsoft.com/en-us/download/details.aspx?id=46436

Documentation and How-To

https://docs.microsoft.com/en-us/azure/cosmos-db/import-data

Source code on GitHub

https://github.com/azure/azure-documentdb-datamigrationtool

### Import from

- SQL Server
  - JSON
  - CSV
- MongoDB
- Azure Table Storage
  - and more...



# Demo



Importing data from SQL Server



```
"id": "Sample",
"familyName": "Jones",
"address": {
   "addressLine": "789 Harbor Boulevard",
   "city": "Chicago",
   "state": "IL",
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
" etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

| Property       | Value                  |
|----------------|------------------------|
| id             | User-defined unique ID |
| user-definable | Partition key          |

```
"id": "Sample",
"familyName": "Jones",
"address": {
   "addressLine": "789 Harbor Boulevard",
   "city": "Chicago",
   "state": "IL",
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
" etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

| Property       | Value                  |
|----------------|------------------------|
| id             | User-defined unique ID |
| user-definable | Partition key          |
| _rid           | Resource ID            |

```
"id": "Sample",
"familyName": "Jones",
"address": {
   "addressLine": "789 Harbor Boulevard",
   "city": "Chicago",
   "state": "IL",
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
" etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

```
id User-defined unique ID

user-definable Partition key

_rid Resource ID

self URI path to the resource
```

```
"id": "Sample",
"familyName": "Jones",
                                      self
"address": (
   "addressLine": "789 Harbor Boulevard",
   "city": "Chicago",
   "state": "IL",
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
"_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

Property

id

```
user-definable Partition key

_rid Resource ID

_self URI path to the resource

GUID (optimistic concurrency)
```

Value

User-defined unique ID

```
"id": "Sample",
"familyName": "Jones",
"address": [
   "addressLine": "789 Harbor Boulev. _etag
   "city": "Chicago",
   "state": "IL",
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
"_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

Property

```
id User-defined unique ID

user-definable Partition key

_rid Resource ID

_self URI path to the resource

_etag GUID (optimistic concurrency)

_attachments URI suffix to the attachments
```

Value

```
"id": "Sample",
"familyName": "Jones",
"address": [
    "addressLine": "789 Harbor Bouleva
   "city": "Chicago",
   "state": "IL",
    "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAA==/",
"_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```



Value

Property

"\_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",

" attachments": "attachments/",

" ts": 1562511939

```
id
                                                    User-defined unique ID
                                   user-definable Partition key
                                                    Resource ID
                                  _rid
"id": "Sample",
"familyName": "Jones",
                                                    URI path to the resource
                                  _self
"address": [
                                  _etag
                                                    GUID (optimistic concurrency)
   "addressLine": "789 Harbor Bouleva
   "city": "Chicago",
                                  _attachments URI suffix to the attachments
   "state": "IL",
                                                    Last updated timestamp (epoch)
                                  ts
   "zipCode": "60603"
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAAA==/",
```

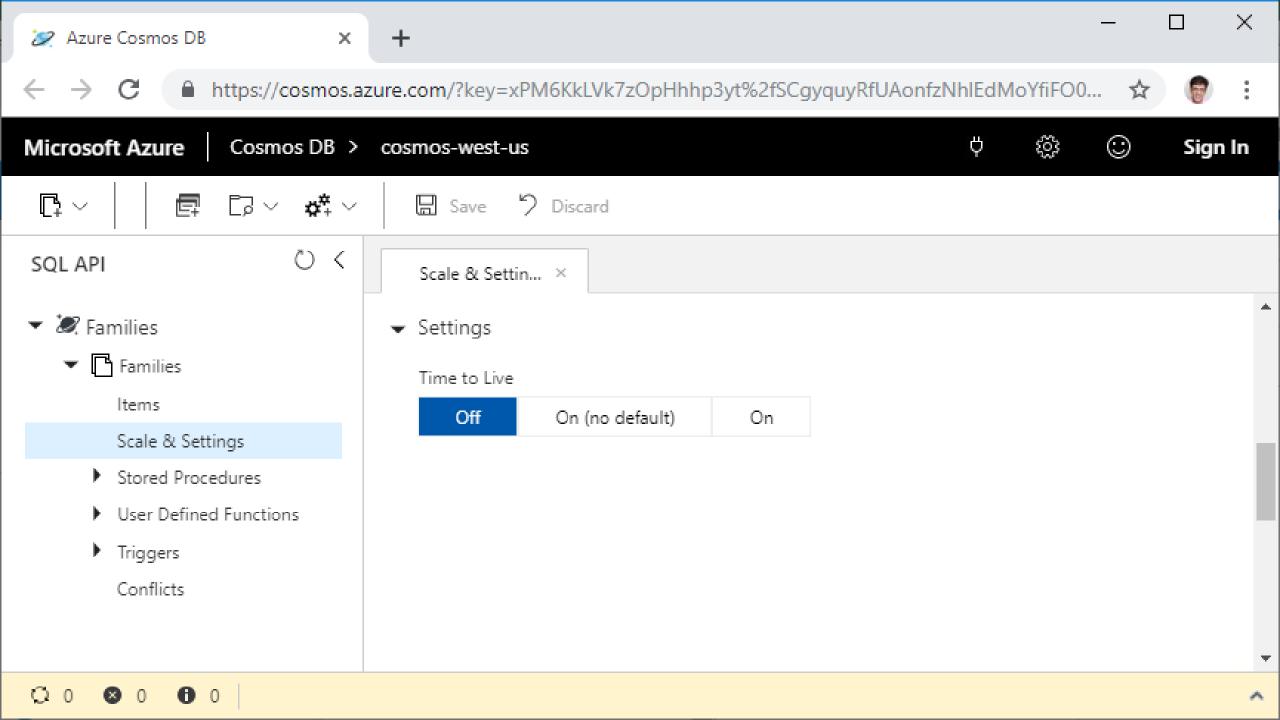
Value

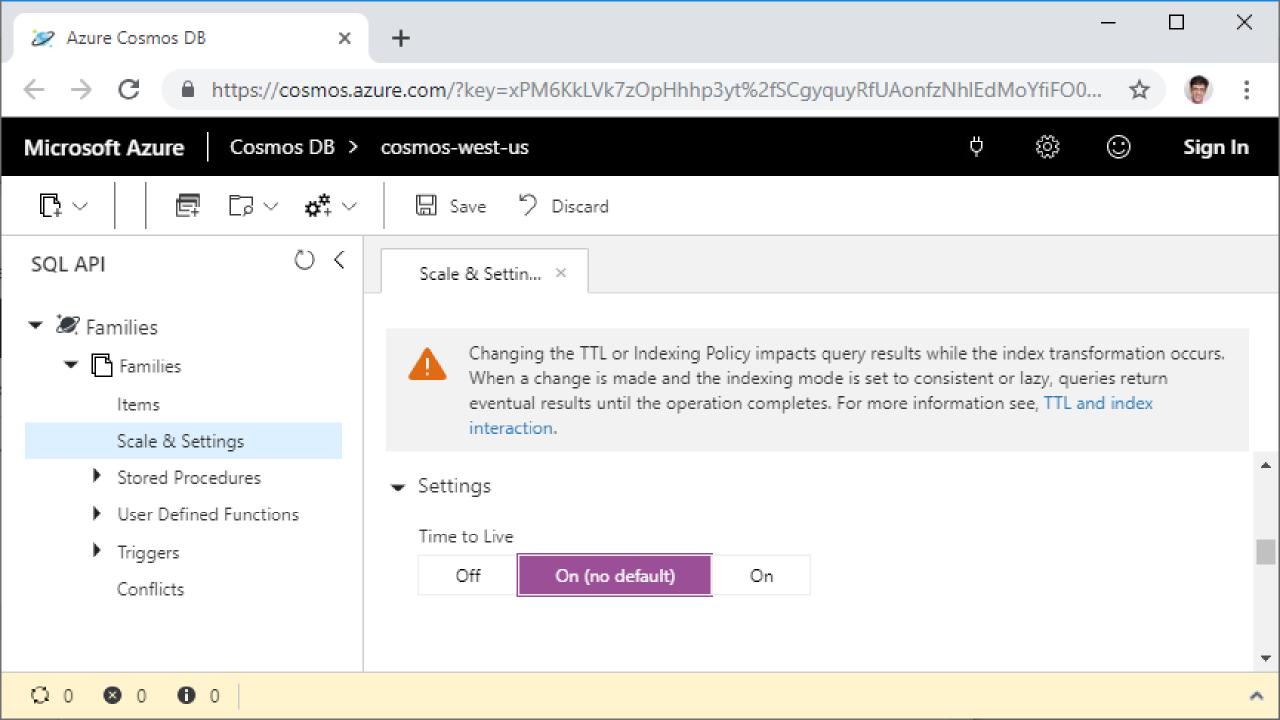
User-defined unique ID

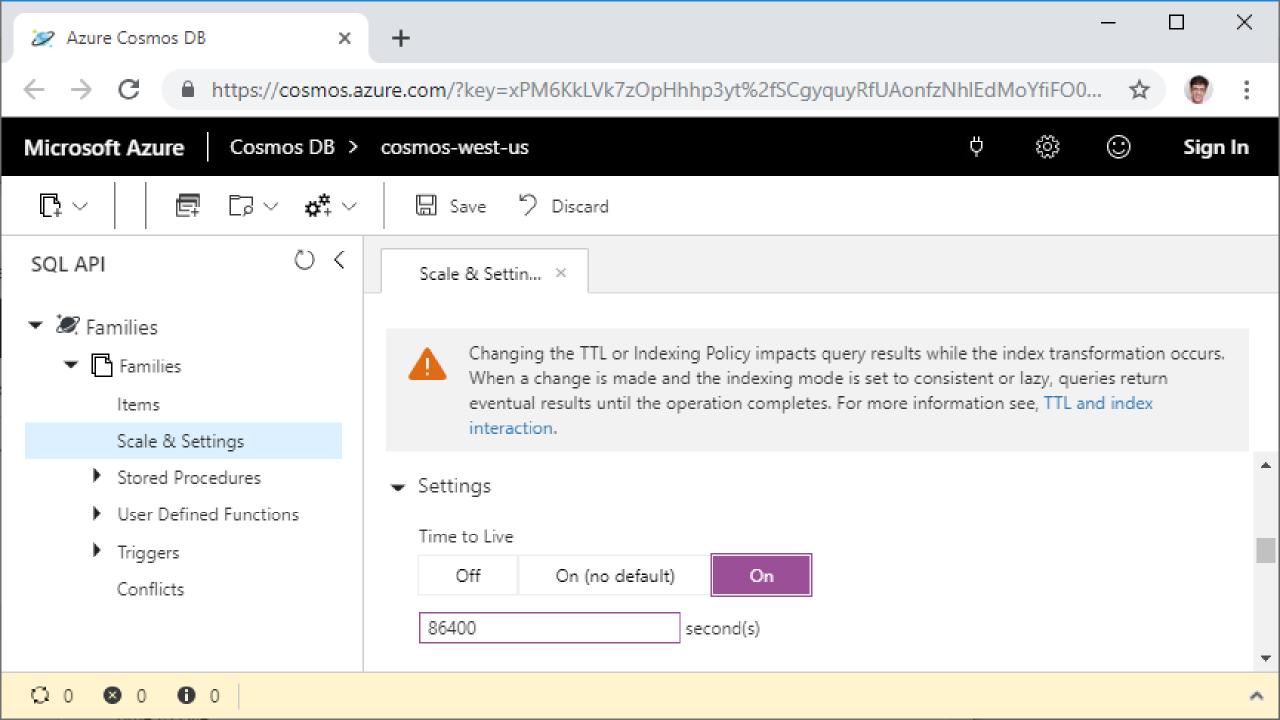
Property

id

```
user-definable Partition key
                                                     Resource ID
                                   _rid
"id": "Sample",
"familyName": "Jones",
                                                     URI path to the resource
                                   _self
"address": [
                                   _etag
                                                     GUID (optimistic concurrency)
   "addressLine": "789 Harbor Bouleva
   "city": "Chicago",
                                   _attachments URI suffix to the attachments
   "state": "IL",
                                                     Last updated timestamp (epoch)
                                   ts
   "zipCode": "60603"
                                   ttl
                                                     Time to Live (expiration)
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAAA==/",
"_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```







Value

User-defined unique ID

Property

id

```
user-definable Partition key
                                                     Resource ID
                                   _rid
"id": "Sample",
"familyName": "Jones",
                                                     URI path to the resource
                                   _self
"address": [
                                   _etag
                                                     GUID (optimistic concurrency)
   "addressLine": "789 Harbor Bouleva
   "city": "Chicago",
                                   _attachments URI suffix to the attachments
   "state": "IL",
                                                     Last updated timestamp (epoch)
                                   ts
   "zipCode": "60603"
                                   ttl
                                                     Time to Live (expiration)
"_rid": "IwdPAIDPIS0BAAAAAAAAA==",
" self": "dbs/IwdPAA==/colls/IwdPAIDPIS0=/docs/IwdPAIDPIS0BAAAAAAAAA==/",
"_etag": "\"09008ef9-0000-0700-0000-5d220a430000\"",
" attachments": "attachments/",
" ts": 1562511939
```

### Summary



**Document database** 

Relational vs. document

Denormalized data model

**Data migration tool** 

