What do Graph Tables bring to the table

Hasan Savran

DATA PLATFORM SUMMIT 2020
VIRTUAL | ROUND-THE-CLOCK
Empowering Data & AI Transformation

WHAT DO GRAPH TABLES BRING TO THE TABLE

Hasan Savran

BI MANAGER

Microsoft® MVP Most Valuable Professional

About Me

- MS Data Platform MVP
- From Cleveland, USA
- BI Manager at Progressive Insurance
- 15 years Web Development
- 7 years Business Intelligence

https://h-savran.blogspot.com/

hasansavran

SavranWeb

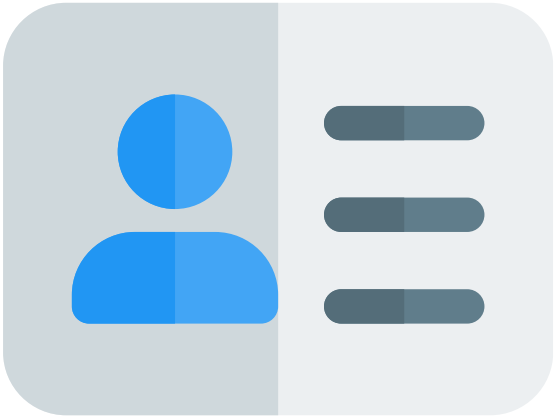
GRAPH DATABASE TABLES

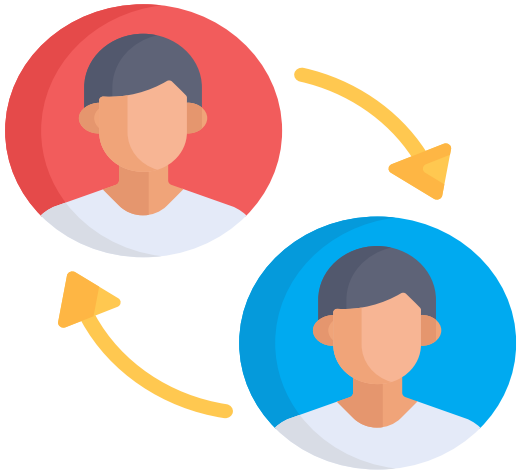
Who uses Graph Tables?

GRAPH DATABASE TABLES VS HIERARCHYID

HIERARCHYID	GRAPH DATABASE TABLES
Highly Structured Data	No fixed boundaries
Multiple Parents are not allowed	Multiple parents are possible
You can have only one root	Root doesn't exist
One to Many Relationships	Many to Many Relationships
Indexable	Indexable
CLR Functions to find data	MATCH Clause
SQL Server 2008+	SQL Server 2017+

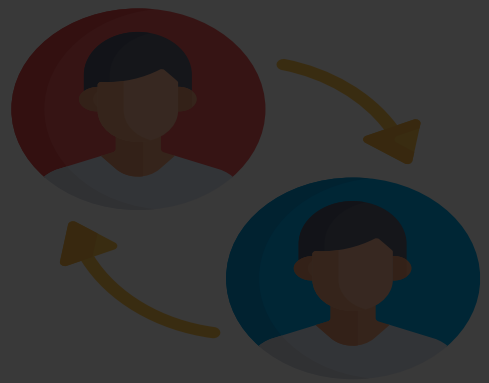
GRAPH DATABASE TABLE TYPES





NODE TABLES	EDGE TABLES
Keep Entity Data	Keep Relation Data
$nodeId is the identity of an Entity	$edgeId is the identity of a relation
They can have any type of columns	They can have any type of columns
	$from_id and $to_id controls the relation
	Relations can not be updated
	Directions are important.

GRAPH DATABASE TABLES <CREATE>



NODE TA

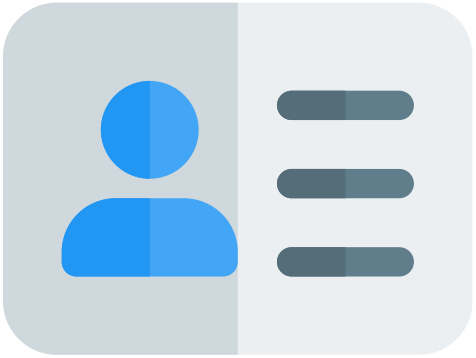
EDGE TABLES

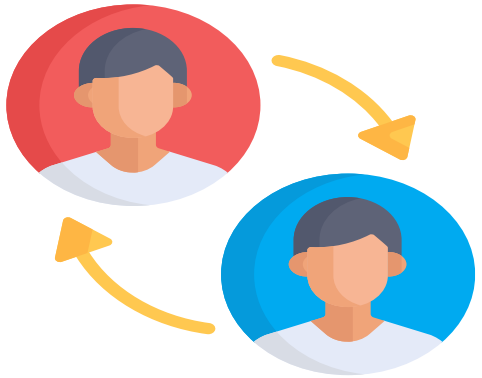
```
CREATE TABLE Products (
Id int PRIMARY KEY Iden
Brand varchar(50) not n
ProductName varchar(200
InStock bit not null,
InvoicePrice money not
RetailPrice money not n
Active bit not null,
Rating tinyint not null
) AS NODE
```

```
BLE AlikeProduct (
 datetime default GetDate()
```

- Graphs
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
      - dbo.AlikeProduct
      - dbo.HR
      - dbo.inprocess
      - dbo.instock
      - dbo.Products
      - dbo.ReportsTo
      - dbo.Users
      - dbo.warehouses
      - dbo.Wished
    - dbo.DistanceDemo
    - dbo.Employees (System-Versioned)
    - dbo.Grocery_UPC_Database
  - Views

GRAPH DATABASE TABLES <INSERT>





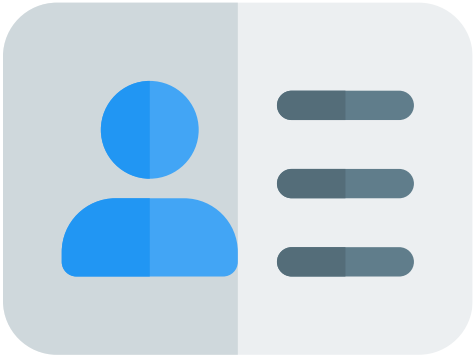
NODE TABLES

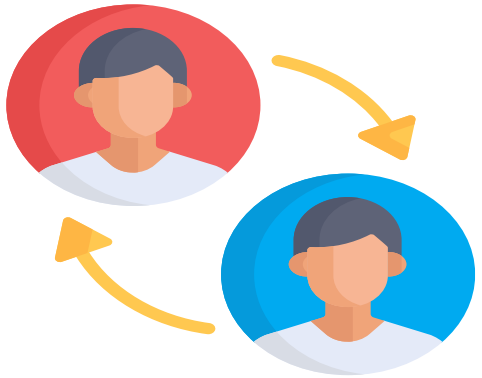
```
INSERT INTO Products (Brand, ProductName,
InStock, InvoicePrice, RetailPrice, Active,
Rating)
VALUES
('Amazon','Kindle Fire 7',1,30,49,1,4),
('Amazon','Kindle Fire HD 8',1,50,79,1,5),
('Amazon','Kindle Fire 7 Kids',1,55,99,1,4)
```

EDGE TABLES

```
INSERT INTO AlikeProduct VALUES(
(SELECT $node_id FROM Products WHERE Id = 1),
(SELECT $node_id FROM Products WHERE Id = 2),
GetDate())
```

GRAPH DATABASE TABLES <UPDATE>





NODE TABLES

```
UPDATE Products
SET InStock = 0
WHERE ProductId = 1
```

EDGE TABLES

```
UPDATE AlikeProduct
SET CreatedDt = GETDATE() -1
FROM Products p1, AlikeProduct ap, Products p2
WHERE MATCH(p1-(ap)->p2) and p1.ProductId =1
```

YOU CANNOT UPDATE RELATIONS
$FROM_ID or $TO_ID

GRAPH DATABASE TABLES <SELECT>

NODE TABLES

Results | Messages

	$node_id_6ACA178D479F4B5FBB88C806CB64C7AF	ProductId	Brand	ProductName	InStock	InvoicePrice	RetailPrice	Active	Rating
1	{"type":"node","schema":"dbo","table":"Products","id":...	1	Amazon	Kindle Fire 7	1	30.00	49.00	1	4
2	{"type":"node","schema":"dbo","table":"Products","id":...	2	Amazon	Kindle Fire HD 8	1	50.00	79.00	1	5
3	{"type":"node","schema":"dbo","table":"Products","id":...	3	Amazon	Kindle Fire 7 Kids	1	55.00	99.00	1	4
4	{"type":"node","schema":"dbo","table":"Products","id":...	4	Amazon	Kindle PaperWhite	1	87.00	120.00	1	4

EDGE TABLES

Results | Messages

	$edge_id_AD5111F3447F4...	$from_id_6E22AE584BB6428A831ED3BEFCB84CF0	$to_id_22754BBCE410471B99F8EBCFA83D5A97	CreatedDt
1	{"type":"edge","schema":"...	{"type":"node","schema":"dbo","table":"Products","id":0}	{"type":"node","schema":"dbo","table":"Products","id":1}	2020-10-15 10:23:17.603
2	{"type":"edge","schema":"...	{"type":"node","schema":"dbo","table":"Products","id":1}	{"type":"node","schema":"dbo","table":"Products","id":5}	2020-10-15 10:23:17.603
3	{"type":"edge","schema":"...	{"type":"node","schema":"dbo","table":"Products","id":6}	{"type":"node","schema":"dbo","table":"Products","id":7}	2020-10-15 10:23:17.603
4	{"type":"edge","schema":"...	{"type":"node","schema":"dbo","table":"Products","id":6}	{"type":"node","schema":"dbo","table":"Products","id":8}	2020-10-15 10:23:17.603

MATCH(graph_search_pattern)

- Specifies the pattern to search
- Pattern needs to go from one entity to another using a relationship
- Arrow character specifies the direction.
- Parenthesis remarks the relationships

```
SELECT U.Name, W.CreatedDT, P.ProductId, P.Brand, P.ProductName
FROM Products P,  Wished W,  Users U
WHERE MATCH(U-(W)->P) and U.UserId = 1
```

QUERYING GRAPH DATABASE TABLES <MATCH>

Fire 7 tablet (7" display, 16 GB) - Black

Brand: Amazon

104,134 ratings | 1000+ answered questions

Amazon's Choice for "kindle fire"

Best Deal

Price: $49.99 prime FREE One-Day
or 5 monthly payments of $10.00

In Stock.

FREE delivery: **Tomorrow**
Order within 5 hrs and 7 mins Details

Ships from and sold by Amazon.com Services LLC.

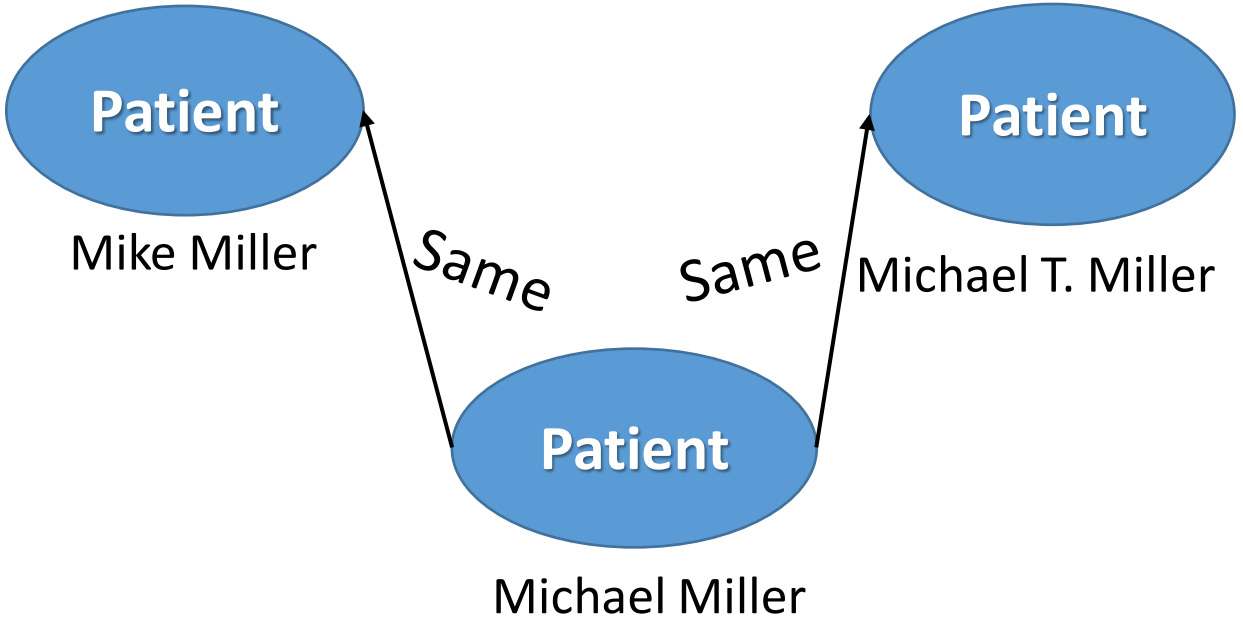
Digital Storage Capacity: **16 GB**

16 GB | 32 GB

Offer Type: **Ad-Supported**

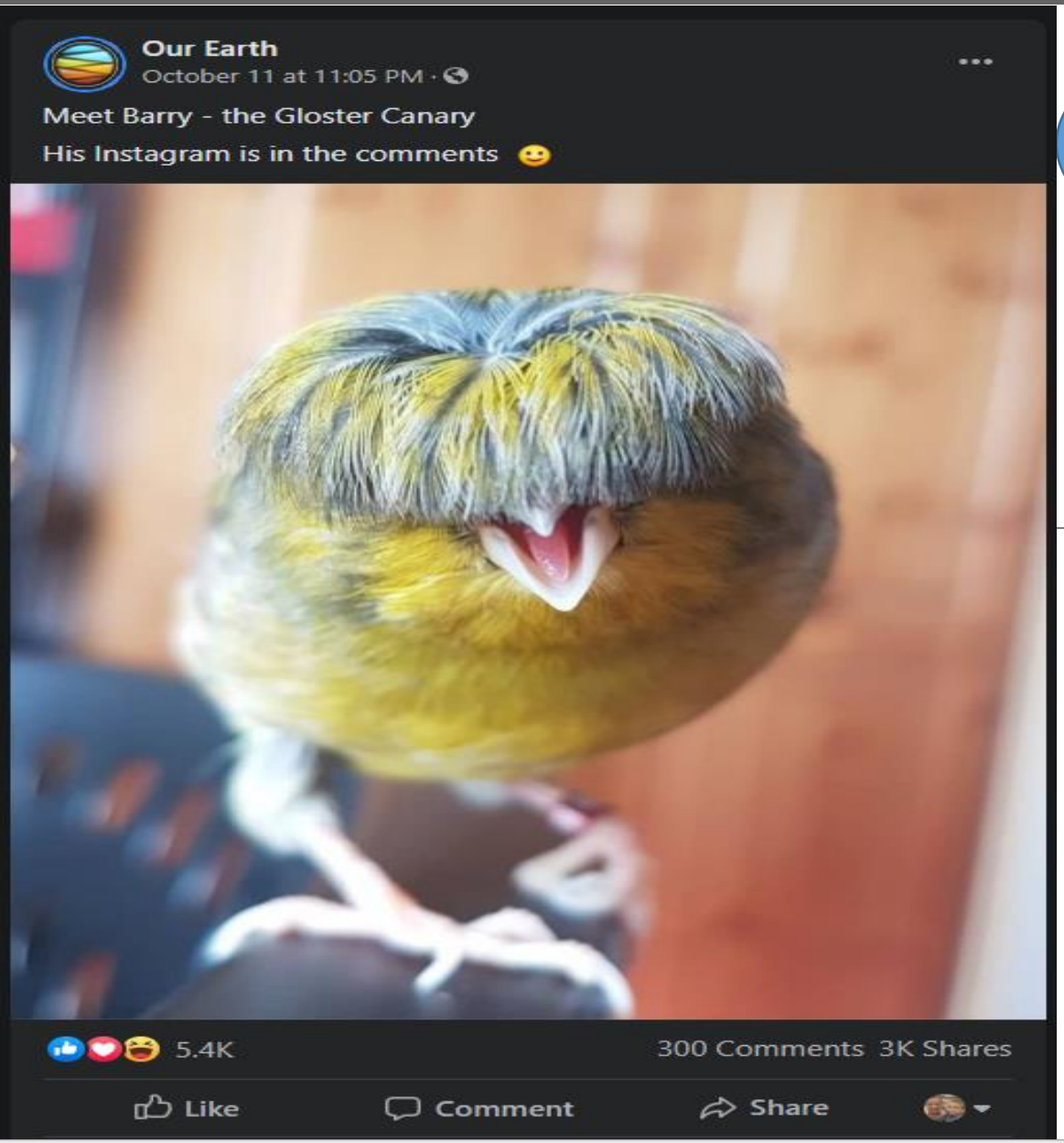
```
SELECT P.Name, P.Desc, P.Price, P2.Name, P2.Price, P2.Desc
FROM Products P,  Alike A,  Products P2
WHERE MATCH(P-(A)->P2) AND P.Id = 1
```

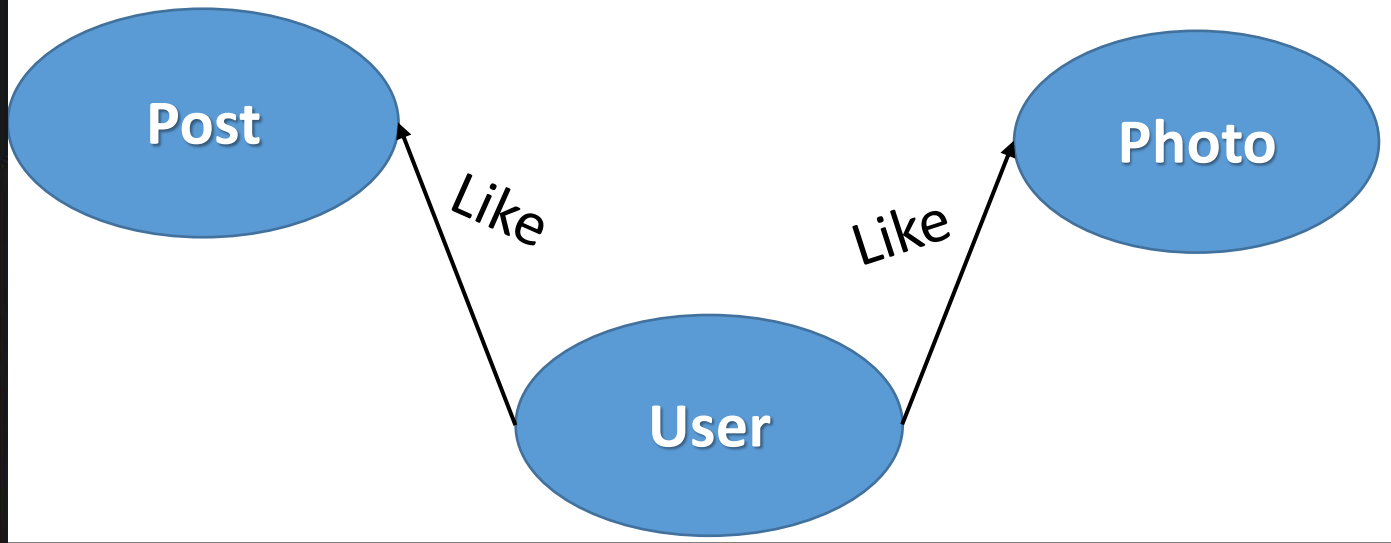
QUERYING GRAPH DATABASE TABLES <MATCH>




```
SELECT P1.Name, P1.MRN, P2.Name,
P2.MRN
FROM Patient P1, Same S,
Patient P2
WHERE MATCH(P1-(S)->P2)
and P1.Id= 1
```

QUERYING GRAPH DATABASE TABLES <MATCH>

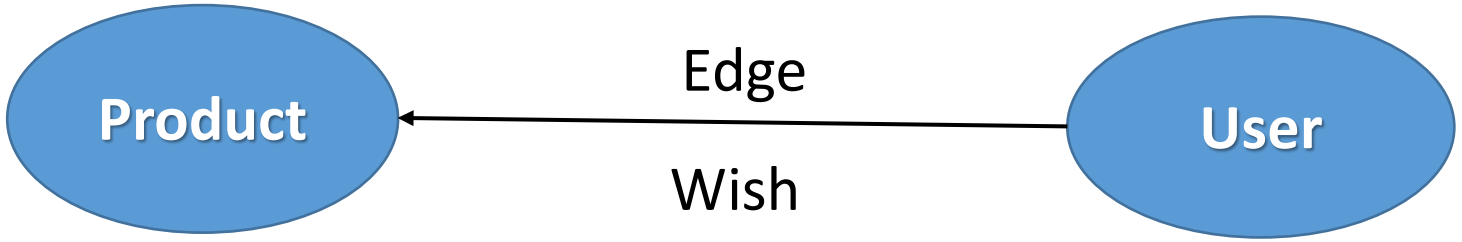





```
SELECT U.Name, L.OccuredDt, P.Id,
P.Text
FROM Users U,  Like L,  Posts P
WHERE MATCH(U-(L)->P) AND P.Id = 1
```

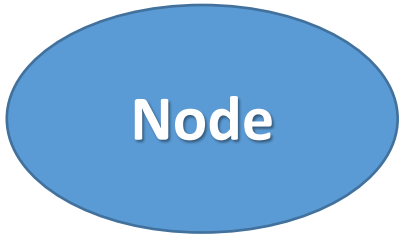
GRAPH DATABASE TABLES <EDGE CONSTRAINTS>

2019 OR Azure SQL

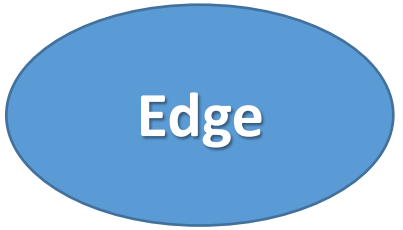



```
ALTER TABLE Wished ADD CONSTRAINT EC_Wish
CONNECTION (Users TO Products)

ALTER TABLE Wished ADD CONSTRAINT EC_Wish
CONNECTION (Users TO Products, Guests TO Products)

ALTER TABLE Wished ADD CONSTRAINT EC_Wish
CONNECTION (Users TO Products) ON DELETE CASCADE

ALTER TABLE Wished ADD CONSTRAINT EC_Wish
CONNECTION (Users TO Products) ON DELETE NO ACTION
```

INDEXING GRAPH DATABASE



$node_id

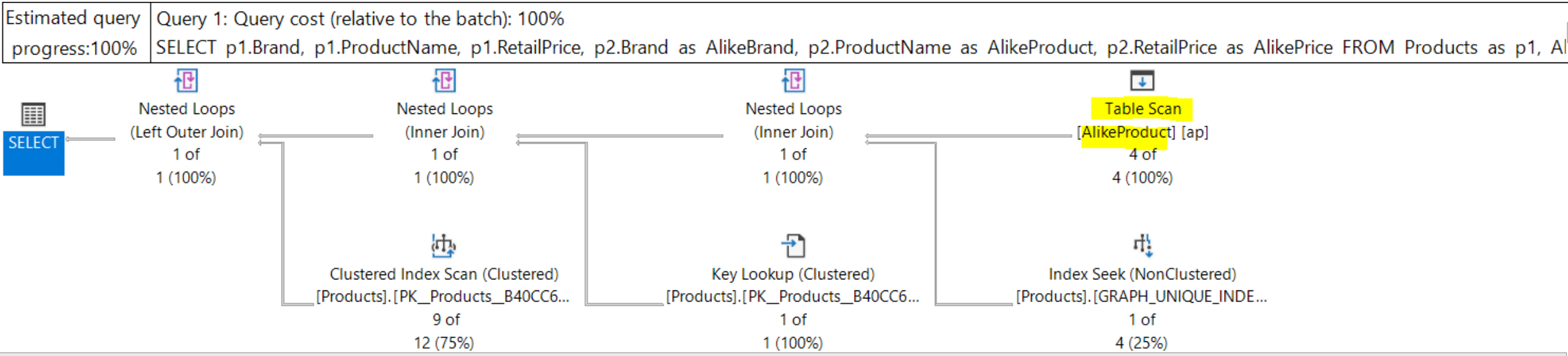


$edge_id
$from_id
$to_id

```
SELECT p1.Brand, p1.ProductName, p1.RetailPrice, p2.Brand as
AlikeBrand, p2.ProductName as AlikeProduct, p2.RetailPrice as
AlikePrice
FROM Products as p1, AlikeProduct as ap, Products p2
WHERE MATCH(p1-(ap)->p2)
AND p1.ProductId = 1
```

INDEXING GRAPH DATABASE

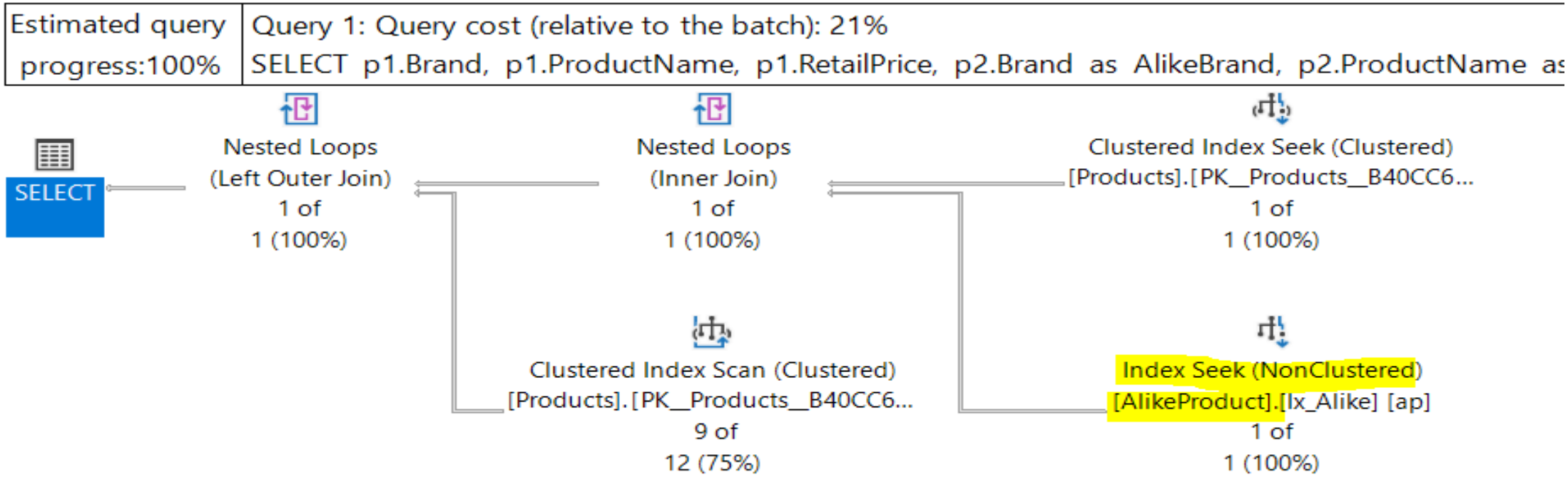
```
SELECT p1.Brand, p1.ProductName, p1.RetailPrice, p2.Brand as
AlikeBrand, p2.ProductName as AlikeProduct, p2.RetailPrice as
AlikePrice
FROM Products as p1, AlikeProduct as ap, Products p2
WHERE MATCH(p1-(ap)->p2)
AND p1.ProductId = 1
```



INDEXING GRAPH DATABASE

```
CREATE UNIQUE NONCLUSTERED INDEX Ix_Alike on AlikeProduct($From_Id, $To_Id)
CREATE UNIQUE NONCLUSTERED INDEX Ix_Wished on Wished ($From_id, $To_id)
```

```
SELECT p1.Brand, p1.ProductName, p1.RetailPrice, p2.Brand as AlikeBrand,
p2.ProductName as AlikeProduct, p2.RetailPrice as AlikePrice
FROM Products as p1, AlikeProduct as ap, Products p2
WHERE MATCH(p1-(ap)->p2)
AND p1.ProductId = 1
```



QUERYING GRAPH DATABASE TABLES <SHORTEST PATH>

- Finds shortest path between two entities.
- Use in MATCH

SELECT {**Graph Path Agg Funcs**}
FROM {**FOR PATH**}
WHERE MATCH(){**Arbitrary Length**}

QUERYING GRAPH DATABASE TABLES <SHORTEST PATH>

SELECT GRAPH PATH AGG FUNCTION WITHIN ORDER CLAUSE

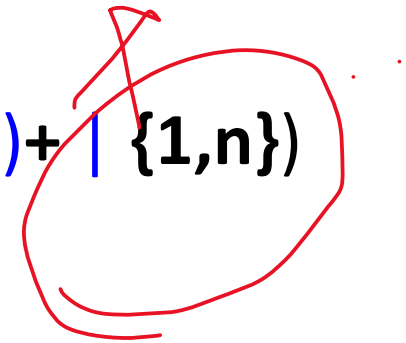
STRING_AGG()
LAST_VALUE()
SUM()
COUNT()
AVG()
MIN()
MAX()

GROUP (GRAPH PATH)

+ : Repeat 1 or more times
{1,n}: Repeat the pattern 1 to n times

FROM Node Or Edge Tables FOR PATH

WHERE MATCH(SHORTEST_PATH(graph search pattern)**+** | **{1,n}**)




DEBUG

Hasan Savran

BI MANAGER

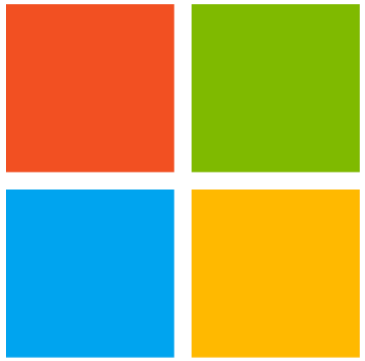
Thank you!

https://h-savran.blogspot.com/

hasansavran

SavranWeb

Special Thanks To



Microsoft

for supporting

DataPlatformGeeks & SQLServerGeeks

Community Initiatives