



**Instituto  
Politécnico  
Nacional**

**UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERIA Y  
TECNOLOGIAS AVANZADAS.**

**BASES DE DATOS DISTRIBUIDAS**  
**Tarea 3**

**Equipo 4:**

Axel Iván Rossano Medina

Alan González Morales

Erik Bravo Pérez

**Docente:** Carlos de la Cruz Sosa

3 de junio del 2025



upiita-ipn

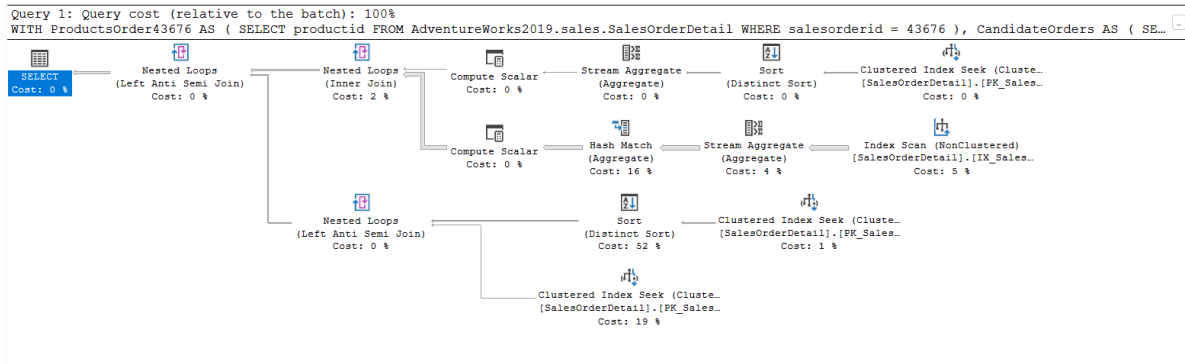
The logo for upiita-ipn is located in the bottom right corner. It consists of a stylized white script 'a' above the text 'upiita-ipn' in a white sans-serif font.

## Consulta 1

- Consulta:

```
WITH ProductsOrder43676 AS (  
    SELECT productid  
    FROM AdventureWorks2019.sales.SalesOrderDetail  
    WHERE salesorderid = 43676  
), CandidateOrders AS (  
    SELECT salesorderid, productid  
    FROM AdventureWorks2019.sales.SalesOrderDetail  
    WHERE salesorderid <> 43676  
)  
SELECT salesorderid  
FROM CandidateOrders  
GROUP BY salesorderid  
HAVING COUNT(DISTINCT productid) >= (  
    SELECT COUNT(DISTINCT productid)  
    FROM ProductsOrder43676  
)  
AND NOT EXISTS (  
    SELECT productid  
    FROM ProductsOrder43676  
    EXCEPT  
    SELECT productid  
    FROM CandidateOrders c  
    WHERE c.salesorderid = CandidateOrders.salesorderid  
);
```

- Plan de ejecución:

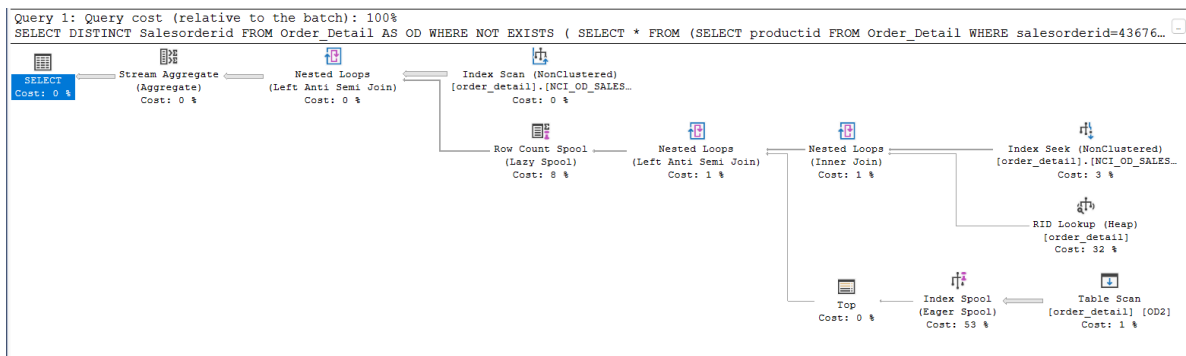


## Consulta 2

- Consulta:

```
SELECT DISTINCT Salesorderid
FROM Order_Detail AS OD
WHERE NOT EXISTS (
    SELECT *
    FROM (SELECT productid
          FROM Order_Detail
          WHERE salesorderid=43676) as P
    WHERE NOT EXISTS (
        SELECT *
        FROM Order_Detail AS OD2
        WHERE OD.salesorderid = OD2.salesorderid
              AND OD2.productid = P.productid
    )
);
```

- Plan de ejecución:



## Conclusión

Llegamos a la conclusión de que la consulta óptima es la primera, esto porque la primera consulta utiliza expresiones comunes de tabla (CTE) y el operador EXCEPT, que es más eficiente que el NOT EXISTS de la segunda consulta porque requiere múltiples escaneos de tabla. Por lo mismo, también presenta un plan de ejecución más optimizado.

El plan de ejecución usa de forma eficiente los índices disponibles por medio de búsquedas directas (*Index Seek*) y evita operaciones costosas como escaneos completos de tablas o *lookups*.

Por el uso de sentencias no optimizadas, la segunda consulta presenta un mayor consumo de recursos y cuellos de botella debido a la reevaluación frecuente de subconsultas para cada fila del conjunto principal, cosa que la primera consulta evita con el uso de WITH.