

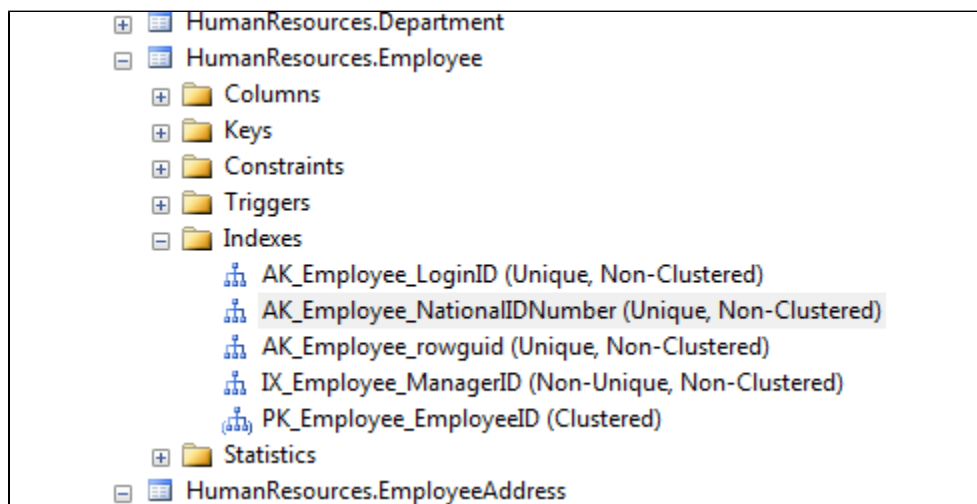
Implicit conversion recommendation

It is common for implicit data type conversions to be performed throughout TSQL. Implicit conversion occurs when data from one object is moved to, compared with, or combined with data from another object of a different data type. Generally, implicit conversions cause an unnoticeable performance impact on query performance so most TSQL developers do not worry about the data type mismatches that lead to the implicit conversion.

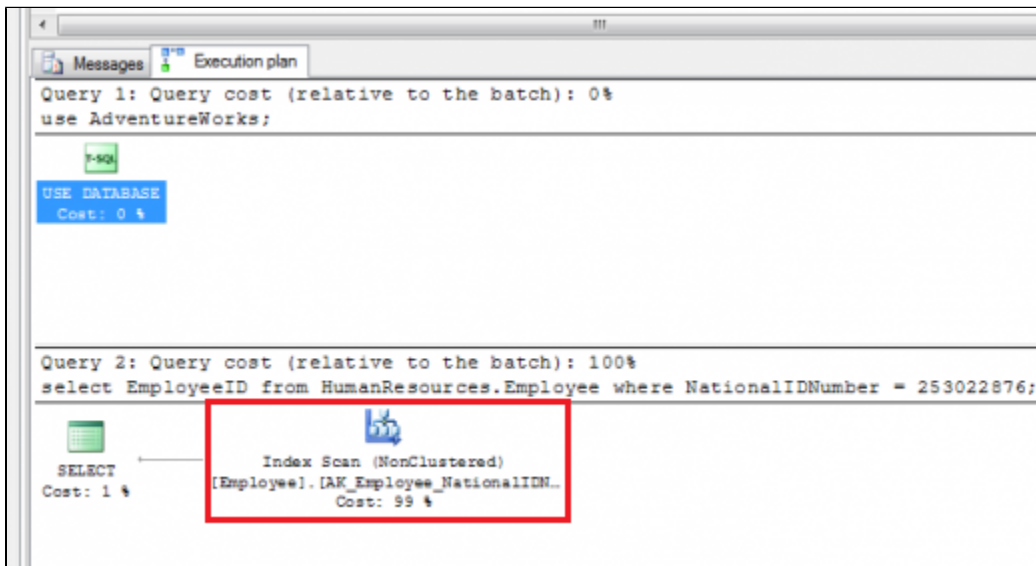
The lack of attention to data type mismatches can have a significant negative impact on query performance. One case, in particular, is when the data type mismatch leads to an implicit conversion being performed on a column used in a search predicate. This implicit conversion prevents the use of an index seek and ensures that an index scan or table scan is performed to satisfy the query predicate. In this case, the implicit conversion is caused by the query optimizer by seeing the column with the implicit conversion as an expression rather than a column.

For example, see the following implicit conversion occurring in a search predicate and the result:

```
use AdventureWorks;
select EmployeeID from HumanResources.Employee
where NationalIDNumber = 253022876;
```



You would expect an index seek to occur on the 'AK_Employee_NationalIDNumber' index; however, the execution plan clearly shows that an index scan is performed instead of an index seek.



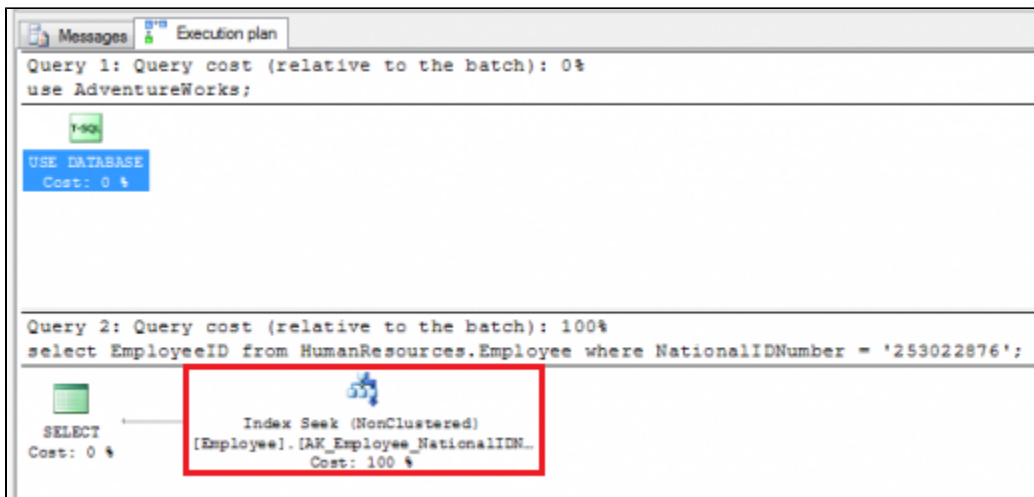
To determine the cause of having an index scan, mouse over the index scan operation of the execution plan to display the tooltip containing additional details:

Index Scan (NonClustered)	
Scan a nonclustered index, entirely or only a range.	
Physical Operation	Index Scan
Logical Operation	Index Scan
Estimated I/O Cost	0.0038657
Estimated CPU Cost	0.000476
Estimated Number of Executions	1
Estimated Operator Cost	0.0043417 (99%)
Estimated Subtree Cost	0.0043417
Estimated Number of Rows	1
Estimated Row Size	32 B
Ordered	False
Node ID	1
Predicate	
CONVERT_IMPLICIT(int,[AdventureWorks].[HumanResources].[Employee].[NationalIDNumber],0)=(253022876)	
Object	
[AdventureWorks].[HumanResources].[Employee].[AK_Employee_NationalIDNumber]	
Output List	
[AdventureWorks].[HumanResources].[Employee].EmployeeID	

Under the 'Predicate' section of the tooltip see that the 'CONVERT_IMPLICIT(int, [AdventureWorks].[HumanResources].[Employee].[NationalIDNumber],0)=(253022876)' implicit conversion is being performed on the 'NationalIDNumber' column. This column value is being converted because the data type of the column mismatches the data type of the literal value being used in the compare. To resolve the data mismatch, change the data type of the literal value to match the data type of the column:

```
use AdventureWorks;
select EmployeeID from HumanResources.Employee
where NationalIDNumber = '253022876';
```

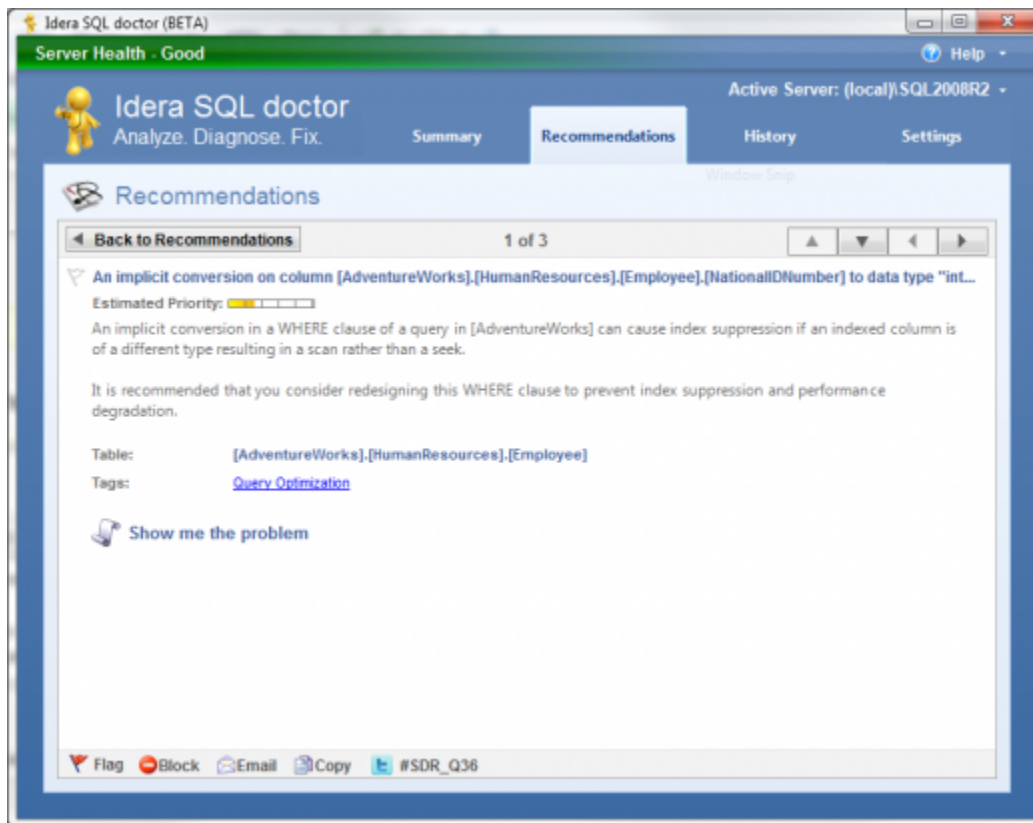
The new execution plan for the updated TSQL is the following:



Note that the index scan is replaced by an index seek operation, optimizing successfully the query.

Use SQL Doctor to identify implicit conversions

IDERA SQL Doctor collects and analyzes your heaviest TSQL to help you identify implicit conversions along with other problems that impact query performance. The following is the resulting recommendation generated by **IDERA SQL Doctor** for the example presented above (SDR-Q36 – Implicit conversion on a column may be causing index suppression):



You can click **Show me the problem** on the Recommendations window to display the TSQL.

[IDERA](#) | [Products](#) | [Purchase](#) | [Support](#) | [Community](#) | [Resources](#) | [About Us](#) | [Legal](#)