UBIDO
BUSINESS SOLUTIONS GMBH

Partitioning

Basics

- Only in Enterprise Edition!
- Split tables and indexes into smaller parts
- All SQL 2005 tables are partitioned
- Easier to manage
  - Load
  - Drop
  - Store
  - Manage (index rebuild, defrag, etc.)
- Improve query performance
  - Joining
    - Use the same partition function for both tables!
  - Use multiple disk drives

Functions and metadata views

- Query the partition number for any value
  - $PARTITION

```
SELECT $PARTITION.RangePF1 (10)
```

- sys.partition_functions
- sys.partition_parameters
- sys.partition_range_values
- sys.partition_schemes
- sys.data_spaces
- sys.destination_data_spaces
- sys.partitions

How?

Initially create partitioned object

```
CREATE PARTITION FUNCTION partition_function_name
( input_parameter_type )
AS RANGE [ LEFT | RIGHT ]
FOR VALUES ( [ boundary_value [ ,...n ] ] )
```

- Create partition function
- Assign partition to filegroup
- Create partition scheme

```
CREATE PARTITION SCHEME partition_scheme_name
AS PARTITION partition_function_name [ ALL ]
TO ( { file_group_name | [ PRIMARY ] } [ ,...n ] )
```

- Create table, reference partition function

```
CREATE TABLE ... table_name (...)
ON partition_scheme_name ( partition_column_name ) ...
```

- Create index, reference partition function

```
CREATE ... INDEX index_name ON ...
ON partition_scheme_name ( column_name ) ...
```

(Re-)Partition an existing table

- Create a new partitioned table, copy data with *insert ... select*
- Create (and/or alter) a partitioned clustered index
- Performe a sequence of *alter partition function* statements

Alter partition

- Alter partition scheme

```
ALTER PARTITION SCHEME partition_scheme_name NEXT USED [ filegroup_name ]
```

- Alter partition function

```
ALTER PARTITION FUNCTION partition_function_name()
{ SPLIT RANGE ( boundary_value ) | MERGE RANGE ( boundary_value ) }
```

Partition switching

- Does not move data physically; just metadata changes
- For what?
  - Assigning a table as a partition to an already existing partitioned table.
  - Switching a partition from one partitioned table to another.
  - Reassigning a partition to form a single table.
- Also valid for constraints!
- Requirements
  - Equal structure of source and target
  - Target and source must exist
  - Target must be empty
  - Additional requirements see msdn

```
ALTER TABLE ... table_name ...
SWITCH [ PARTITION source_partition_number_expression ]
TO [ schema_name. ] target_table [ PARTITION target_partition_number_expression ] }
```