Using Deferred Segment Creation

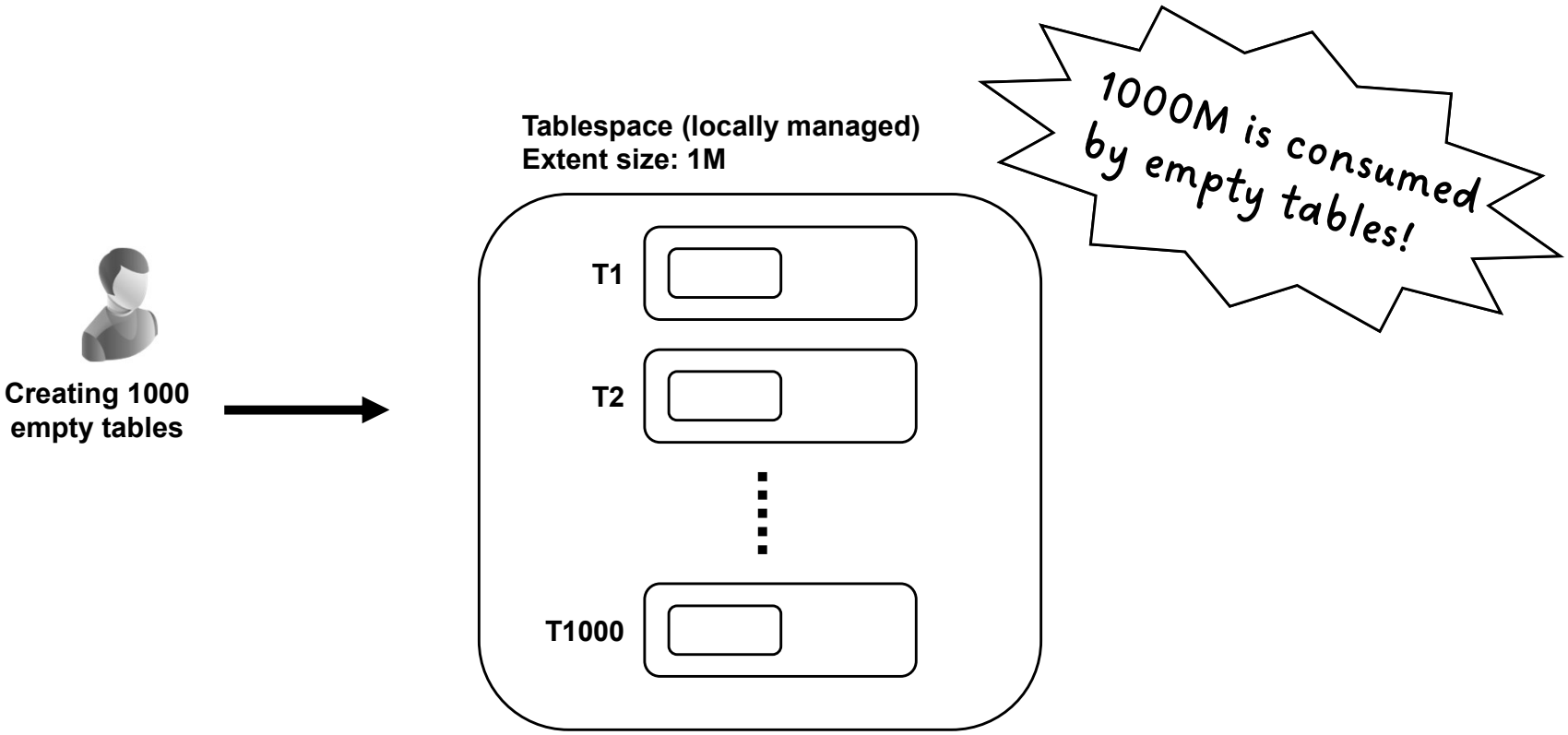
By Ahmed Baraka

Objectives

By the end of this lecture, you should be able to perform the following:

- Understand and use deferred segment creation
- Materialize database segments

Before 11g: Space Consumed by Empty Tables



About the Deferred Segment Creation

- The object is created but its segment is not, until the first row is inserted
- The tables appear in the **`*_TABLES`** but not in **`*_SEGMENTS`**
- Benefits:
  - Saves disk space usage when deploying new system
  - Reduces application installation time

When Segment Creation is Deferred?

- When the **DEFERRED_SEGMENT_CREATION** parameter (system or session) is set to **TRUE** (default), tables are created with deferred option.
- To supersede the database setting:

```
CREATE TABLE ... [SEGMENT CREATION DEFERRED | IMMEDIATE]
```

- When a segment is created, the **SEGMENT_CREATED** column is updated in ***_TABLES**, ***_INDEXES**, and ***_LOBS** views for nonpartitioned tables, and in ***_TAB_PARTITIONS**, ***_IND_PARTITIONS**, and ***_LOB_PARTITIONS** views for partitioned tables.

Materializing Database Segments

- To materialize segments for tables, table partitions, and dependent objects created with deferred segment creation enabled:

```
DBMS_SPACE_ADMIN.MATERIALIZE_DEFERRED_SEGMENTS (
 SCHEMA_NAME    IN VARCHAR2 DEFAULT NULL,
 TABLE_NAME     IN VARCHAR2 DEFAULT NULL,
 PARTITION_NAME IN VARCHAR2 DEFAULT NULL);
```

- Example:

```
BEGIN
DBMS_SPACE_ADMIN.MATERIALIZE_DEFERRED_SEGMENTS(
 SCHEMA_NAME => 'SOE',
 TABLE_NAME => 'ORDERS');
END;
```

Summary

In this lecture, you should have learnt how to perform the following:

- Understand and use deferred segment creation
- Materialize database segments