Shape Command Sample Purpose

The purpose of this sample is to provide sample Shape commands that illustrate both a compound relation hierarchy (a hierarchy with depth greater than two) and a multiple relation hierarchy (a hierarchy with a parent table that has multiple child tables). The best summary document I've found that discusses Shape command syntax is the following Microsoft Knowledge Base article:

http://support.microsoft.com/kb/189657/en-us

Additional information about Shape commands can typically be found in any Active X Data Objects (ADO) reference.

Sample Shape Command Syntax

Sample SQL queries in a typical sample SQL document are bracketed as follows:

--Sample query begins…
one or more SQL statements
--Sample query ends.

This bracketing isn't possible for Shape commands because a Shape command doesn't allow a SQL comment outside the context of an actual SQL statement. Instead, each set of SQL statements within the following shape commands is bracketed as follows:

--SQL query begins…
SQL Select statement
--SQL query ends.

Note that each SQL Select statement within a Shape command must be enclosed in braces.

Keep in mind that a query script cannot include more than one Shape command, nor can it include a Shape command and additional SQL statements outside of a Shape command. One very negative ramification of this constraint is that a set of map unit keys for a large area of interest cannot be implemented as a local temporary table when using a Shape command. For additional information about implementing a set of map unit keys for a large area of interest as a local temporary table, please see the sample query document titled "Constraining a Query to an Area of Interest".

Sample 1 - Fundament Query as a Shape Command

The following shows a variant of the fundamental query (see the sample query document titled "Fundamental Query) implemented as a Shape command. The following query returns selected attributes from tables "sacatalog", "legend", "mapunit", "component" and "chorizon", for the SSURGO map units in a four square mile area of interest in Hall County Nebraska. For additional information about constraining a query to an area of interest, please see the sample query document titled "Constraining a Query to an Area of Interest".

Note that each Select statement in a Shape command must return the column or columns that permit the results of that Select statement to be joined to the results of any related Select statement. For example, the second Select statement below, which targets table "mapunit", returns "lkey" so that these results can be joined to the results of the first Select statement, and also returns "mukey" so that these results can be joined to the results of the third Select statement.

The tables referenced in the following Shape command represent a compound relation hierarchy (a hierarchy with depth greater than two).

```
SHAPE
{
  --SQL query begins…
  SELECT DISTINCT saversion, saverest,
  l.areasymbol, l.areaname, l.lkey

  FROM sacatalog sac
    INNER JOIN legend l ON sac.areasymbol = l.areasymbol
      INNER JOIN mapunit mu ON l.lkey = mu.lkey

  WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
  --SQL query ends.
}
APPEND
(
  (SHAPE
  {
    --SQL query begins…
    SELECT lkey, musym, muname, museq, mukey

    FROM mapunit mu

    WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
    --SQL query ends.
  }
  APPEND
  (
    (SHAPE
    {
      --SQL query begins…
      SELECT mu.mukey,
      comppct_r, compname, localphase, slope_r, c.cokey

      FROM mapunit mu
        LEFT OUTER JOIN component c ON mu.mukey = c.mukey

      WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
      --SQL query ends.
    }
    APPEND
    (
      {
        --SQL query begins…
        SELECT c.cokey,
        hzdept_r, hzdepb_r, ch.chkey

        FROM mapunit mu
          LEFT OUTER JOIN component c ON mu.mukey = c.mukey
            LEFT OUTER JOIN chorizon ch ON c.cokey = ch.cokey

        WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
        --SQL query ends.
      }
      AS chorizon RELATE cokey TO cokey)
    )
    AS component RELATE mukey TO mukey)
  )
  AS mapunit RELATE lkey TO lkey
)
```

Sample 2 - Returning all Geomorphology Related Information for an Area of Interest

The following Shape command retrieves all available geomorphology related information for the SSURGO map units in a four square mile area of interest in Hall County Nebraska. For additional information about constraining a query to an area of interest, please see the sample query document titled "Constraining a Query to an Area of Interest".

The tables referenced in the following Shape command represent both a compound relation hierarchy (a hierarchy with depth greater than two) and a multiple relation hierarchy (a hierarchy with a parent table that has multiple child tables).

Geomorphology related information is distributed between the following tables: "component", "cogeomordesc", "cosurfmorphgc", "cosurfmorphhpp", "cosurfmorphmr" and "cosurfmorphss". Tables "cosurfmorphgc", "cosurfmorphhpp", "cosurfmorphmr" and "cosurfmorphss" are all immediate child tables of "cogeomordesc".

For a description of these tables, and the geomorphology related attributes in these tables, please see the metadata reports titled "Tables and Columns Report" and "Table Column Descriptions Report" at:

http://sdmdataaccess.nrcs.usda.gov/QueryHelp.aspx

```
SHAPE
{
  --SQL query begins…
  SELECT DISTINCT saversion, saverest,
  l.areasymbol, l.areaname, l.lkey

  FROM sacatalog sac
    INNER JOIN legend l ON sac.areasymbol = l.areasymbol
      INNER JOIN mapunit mu ON l.lkey = mu.lkey

  WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
  --SQL query ends.
}
APPEND
(
  (SHAPE
  {
    --SQL query begins…
    SELECT lkey, musym, muname, museq, mukey

    FROM mapunit mu

    WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
    --SQL query ends.
  }
  APPEND
  (
    (SHAPE
    {
      --SQL query begins…
      SELECT mu.mukey,
      comppct_r, compname, localphase, slope_r, geomdesc, c.cokey

      FROM mapunit mu
        LEFT OUTER JOIN component c ON mu.mukey = c.mukey

      WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
      --SQL query ends.
    }
    APPEND
    (
      (SHAPE
      {
        --SQL query begins…
        SELECT c.cokey,
        geomftname, geomfname, geomfmod, geomfeatid, existsonfeat,
rvindicator, cgd.cogeomdkey

        FROM mapunit mu
          LEFT OUTER JOIN component c ON mu.mukey = c.mukey
            LEFT OUTER JOIN cogeomordesc cgd ON c.cokey = cgd.cokey

        WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
        --SQL query ends.
      }
      APPEND
      (
        {
          --SQL query begins…
          SELECT cgd.cogeomdkey,
          geomposmntn, geomposhill, geompostrce, geomposflats

          FROM mapunit mu
            LEFT OUTER JOIN component c ON mu.mukey = c.mukey
              LEFT OUTER JOIN cogeomordesc cgd ON c.cokey = cgd.cokey
                LEFT OUTER JOIN cosurfmorphgc cgdgc ON cgd.cogeomdkey =
cgdgc.cogeomdkey
```

```
          WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
          --SQL query ends.
        }
        AS cosurfmorphgc RELATE cogeomdkey TO cogeomdkey
      ),
      (
        {
          --SQL query begins…
          SELECT cgd.cogeomdkey,
          hillslopeprof

          FROM mapunit mu
            LEFT OUTER JOIN component c ON mu.mukey = c.mukey
              LEFT OUTER JOIN cogeomordesc cgd ON c.cokey = cgd.cokey
                LEFT OUTER JOIN cosurfmorphhpp cgdhpp ON cgd.cogeomdkey =
cgdhpp.cogeomdkey

          WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
          --SQL query ends.
        }
        AS cosurfmorphhpp RELATE cogeomdkey TO cogeomdkey
      ),
      (
        {
          --SQL query begins…
          SELECT cgd.cogeomdkey,
          geomicrorelief

          FROM mapunit mu
            LEFT OUTER JOIN component c ON mu.mukey = c.mukey
              LEFT OUTER JOIN cogeomordesc cgd ON c.cokey = cgd.cokey
                LEFT OUTER JOIN cosurfmorphmr cgdmr ON cgd.cogeomdkey =
cgdmr.cogeomdkey

          WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
          --SQL query ends.
        }
        AS cosurfmorphmr RELATE cogeomdkey TO cogeomdkey
      ),
      (
        {
          --SQL query begins…
          SELECT cgd.cogeomdkey,
          shapeacross, shapedown

          FROM mapunit mu
            LEFT OUTER JOIN component c ON mu.mukey = c.mukey
              LEFT OUTER JOIN cogeomordesc cgd ON c.cokey = cgd.cokey
                LEFT OUTER JOIN cosurfmorphss cgdss ON cgd.cogeomdkey =
cgdss.cogeomdkey

          WHERE mu.mukey IN

('107559','107646','107674','107682','107707','107794','107853','107854','
107865','107867','107869','107870','107871')
          --SQL query ends.
        }
        AS cosurfmorphss RELATE cogeomdkey TO cogeomdkey
      )
      )
      AS cogeomordesc RELATE cokey TO cokey)
    )
    AS component RELATE mukey TO mukey)
  )
  AS mapunit RELATE lkey TO lkey
)
```