Using Sequences

This feature is available in midPoint 3.3 and later.

Sequences

Sequences are persistent objects in midPoint repository that maintain counters. They are mostly equivalent in functionality to database auto-increment columns. The sequences can be used to assign unique identifiers to large number of midPoint objects in a very efficient and reliable fashion. The sequences are excellent tool for automatic assignment of UNIX uid and gid numbers, persistent user identifiers, etc.

The sequence in it self is a very simple object:

```
<sequence>
    <name>Unix UID numbers</name>
    <counter>1001</counter>
    <maxUnusedValues>10</maxUnusedValues>
</sequence>
```

The sequence remembers just one important value: the counter. The counter value is assigned to the next object, then the counter value is (atomically) incremented.

Sequences in Expressions

There is a special-purpose expression evaluator that works with sequences. It is `sequentialValue` evaluator:

```
            <mapping>
                <name>sequenceUID</name>
                <strength>weak</strength>
                <expression>
                    <sequentialValue>
                        <sequenceRef oid="7d4acb8c-65e3-11e5-9ef4-6382ba96fe6c"/>
                    </sequentialValue>
                </expression>
                <target>
                    <path>extension/uidNumber</path>
                </target>
            </mapping>
```

This mapping will take a counter from the sequence and place it in the target property. The mapping will also make sure that the value is returned to the sequence for reclamation in case that the processing fails. See Using Sequences page for more details.

Sequences in Script Expressions

There is also library function `getSequenceCounter` that can be used to retrieve value from a sequence:

```
                    <mapping>
                <name>sequenceGID</name>
                <strength>weak</strength>
                <expression>
                    <script>
                        <code>
                            1000 - midpoint.getSequenceCounter("02cb7caa-6618-11e5-87a5-7b6c6776a63e")
                        </code>
                    </script>
                </expression>
                <target>
                    <path>extension/gidNumber</path>
                </target>
            </mapping>
```

This method gives better control over how the sequence value is computed or formatted. The example above shows how to assign Unix GID numbers starting from 999 and going down to 998, 997, etc. The sequences are always increasing, therefore mathematic expression in the script in needed in this case.

Value Stickiness and Reclamation

Sequence values are retrieved just once for each operation. The values are kept in Model Context and re-used every time the expression is re-evaluated during that operation. This is a mechanism to avoid taking multiple values from the sequence. However, the assigned value is not explicitly remember across operations. Therefore is is strongly recommended to use sequences in *weak* mappings.

The operation that retrieved a sequence value may fail before it can persistently store the object. In such a case midPoint will take care of returning the value back to the sequence for reclamation. The sequence value is reused on the following attempt. This is the mechanism that conserves the sequence values and this is what is suitable for most of the cases. However, this may lead to a sequence that is not strictly monotonous. If a monotonous sequence is required then this behavior can be disabled by setting the `maxUnusedValues` property of the sequence to zero.

Limitations

Taking a value from the sequence is an operation that is not entirely reversible and sequences can become depleted of the identifiers. Also, it is difficult to explicitly remember sequence values that were taken - and midPoint does NOT do that. The value is only "remembered" in the target property value. Therefore we strongly recommend to use sequence expression in **weak mappings**. Weak mappings ensure that the sequence value is only assigned once for each object.

The sequence expressions are currently (midPoint 3.3) supported only in object templates and in focus mappings in roles. The use of sequence expressions in outbound, inbound or any other place is not supported.

Example: UNIX systems

See Unix Story Test for a complete example that is using sequences. The test is using sequences to assign Unix UID and GID numbers.

See Also

- Sequences
- Mappings and Expressions
- Unix Story Test