setTransactionID

| **Syntax** | ```
setTransactionID() = aString
``` |
| **Semantics** | Sets the current transaction ID to the specified **String** value.

ⓘ The transaction ID has a maximum length of 40 bytes. Longer values are cut by the xUML Runtime.
If your transaction ID contains multibyte characters, this limitation may lead to the problem that characters are cut somewhere in between (see also the hint below: How to Limit the Size of a Transaction ID). |
| **Examples** | ```
setTransactionID(a_TID_from_caller);
setTransactionID(concat(a_TID_from_caller, aStringAttribute));
``` |
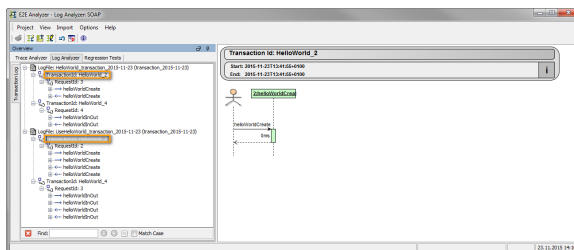
The **Transaction ID** identifies a transaction. It is a unique number used to trace service calls through the call stack of multiple service calls.

- Runtime 2019.9 Clients calling a service running on the Bridge can provide a transaction ID in HTTP header **X-Transaction-ID** or **xTransactionId** (in JMS context).
- SOAP clients can also use the SOAP headers to provide a transaction ID.
- If an xUML service is called without providing a transaction ID, the xUML Runtime will generate such an ID.

This ID will be passed on through the call stack of the xUML service, so that the whole transaction can be traced. This can be useful, when analyzing the log file in case of error.

For more information on how to provide a transaction ID in the SOAP header refer to SOAP Adapter Reference.

In the Analyzer, the transaction log of an xUML service will be sorted by transaction ID.

How to Limit the Size of a Transaction ID

Given a transaction ID **aTransactionID**, for example taken from the process ID of a persistent state instance

```
local aTransactionID = self.getProcessID();
```

You can make sure that the transaction ID is a proper at-most-40-byte-sequence by doing the following preprocessing:

1. Cut the transaction ID after 40 bytes and take care that multibyte characters are not split in between. This is done by calculating the amount of superfluous bytes in a 40 character sequence.

    ```
    local superBytes = aTransactionID.substring(0, 40).transcodeToBlob
    ('utf-8').blobLength() - 40;
    ```

2. Next, reduce the transaction ID by the calculated amount of characters.

```
setTransactionID(aTransactionID.substring(0, 40 - superBytes));
```