ETSI ES 203 022 V1.2.1 (2018-05)

Methods for Testing and Specification (MTS);

The Testing and Test Control Notation version 3;

TTCN-3 extension: Advanced Matching

**ETSI Standard**

Reference

RES/MTS-203022ed121

Keywords

conformance, testing, TTCN-3

***ETSI***

650 Route des Lucioles

F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C

Association à but non lucratif enregistrée à la

Sous-Préfecture de Grasse (06) N° 7803/88

***Important notice***

The present document can be downloaded from:
<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at <https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:
<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

***Copyright Notification***

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2018.

All rights reserved.

**DECT**TM, **PLUGTESTS**TM, **UMTS**TM and the ETSI logo are trademarks of ETSI registered for the benefit of its Members.
**3GPP**TM and **LTE**TM are trademarks of ETSI registered for the benefit of its Members and
of the 3GPP Organizational Partners.
**oneM2M** logo is protected for the benefit of its Members.
**GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

# 5.5 Encoding Mutation Annotation

In value templates, it is allowed to add a mutation annotation to parts of the value which is applied during encoding of the annotated part as a post-processing step to the original result produced by the encoder for that part.

***Syntactical Structure***

[ *TemplateInstance* ] (**@mutation** | **@mutation\_o** | **@mutation\_unichar** [ "(" *StringEncoding* ")" ]) *Expression*

***Semantic Description***

The family of mutation annotations **@mutation**, **@mutation\_o** and **@mutation\_unichar** are value templates that can be part of special value templates whose only purpose is to be encoded.

If the mutation annotation keyword occurs to the right of a *TemplateInstance*, then the *Expression* on the right side of the mutation annotation keyword can use the keyword **value** as an implicit formal parameter to reference the encoded value of that *TemplateInstance*. If the *Expression* does not need to reference the encoded value, then the *TemplateInstance* may be omitted.

If the **@mutation** keyword is used, then the **value** keyword refers to an expression of type **bitstring** and the *Expression* shall evaluate to a value of type **bitstring**.

If the **@mutation\_o** keyword is used, then the **value** keyword refers to an expression of type **octetstring** and the *Expression* shall evaluate to a value of type **octetstring**.

If the @**mutation\_unichar** keyword is used, then the **value** keyword refers to an expression of type **universal charstring** and the *Expression* shall evaluate to a value of type **universal charstring**. If a different string encoding than the default “UTF-8” is used for the universal charstring, then this string encoding is given as an additional *StringEncoding* operand in parenthesis after the **@mutation\_unichar** keyword.

When an encoder processes a value template that is a mutation annotation with a *TemplateInstance*, it will first encode that *TemplateInstance* into a sub-message. It will then transform that sub-message into a TTCN-3 string value of the appropriate type (depending on which mutation annotation is used) and then invoke the evaluation of the mutation *Expression*, using the transformed string value as an actual parameter of the formal parameter **value**. The result of the evaluation is transformed back to a sub-message which is then used instead of the original sub-message as part of the resulting message.

When an encoder processes a value template that is a mutation annotation without a *TemplateInstance*, it will evaluate the mutation *Expression* and transform the resulting value to a sub-message which is then used as the part of the message corresponding to the encoded value.

If the **@mutation\_o** keyword is used, the sub-message is transformed into a left-aligned **octetstring** before transformation, so that if the sub-message does not have a bit-length divisible by 8, the appropriate amount of padding bits are the least significant bits of the least significant octet of the **octetstring**. The bit-content of the whole octetstring that is the result of the evaluation will be used as the resulting sub-message.

If the **@mutation\_unichar** keyword is used, the sub-message is transformed depending on the given *StringEncoding* into a **univeral charstring**. The transformed sub-message must be byte-aligned and have a bit-length that is consistent with the given *StringEncoding* and otherwise an error will be produced. The result of the evaluation is a **universal charstring** that is transformed into a sub-message by using the given *StringEncoding* to encode it into a byte-aligned binary representation.

***Restrictions***

1. The only operations allowed on templates using mutation annotations are encoding and sending operations. They shall not be used for matching, decoding or receiving operations.
2. The *Expression* shall conform to the restrictions given in clause 16.4.1 and shall not use any functions with a runs on clause.