XEP-0457: Message Fancying

Georg Lukas
mailto:georg@op-co.de
xmpp:georg@yax.im

2021-04-01
Version 1.0.0

Status	Type	Short Name
Active	Humorous	fancying

This specification defines a Unicode-formatted fancy text syntax for use in instant messages.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 by the XMPP Standards Foundation (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the ”Specification”), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

NOTE WELL: This Specification is provided on an ”AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF’s Intellectual Property Rights Policy (a copy of which can be found at <https://xmpp.org/about/xsf/ipr-policy> or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

Contents

1 Introduction 1

2 Requirements 1

3 Use Cases 1

4 Business Rules 2
- 4.1 Preformatted Text . . . 2
- 4.2 Emphasis . . . 2
- 4.3 Strong Emphasis . . . 2
- 4.4 Very Strong Emphasis . . . 2
- 4.5 Strike through . . . 2

5 Implementation Notes 2

6 Accessibility Considerations 3

7 Internationalization Considerations 3

8 Security Considerations 3

9 IANA Considerations 3

10 XMPP Registrar Considerations 3

1 Introduction

XMPP-based Instant Messages has suffered from a lack of proper message styling mechanisms. After abandoning XHTML-IM (XEP-0071) [1] for its horrible security track record, Message Styling (XEP-0393) [2] and Message Markup (XEP-0394) [3] were proposed as replacements, but neither was able to gain sufficient traction to become the default styling mechanism.
This specification fills the gap by formally specifying the Unicode-based formatting that is currently fancied on Social Media and micro-blogging as an XMPP standard for Instant Messaging. As XMPP is based on UTF-8, messages formatted with this mechanism are immediately visible on all receiving clients. Simple third-party tools can be used for writing messages until this specification is integrated into the IM message input boxes.

2 Requirements

This specification relies on Unicode features introduced in the 3.1 specification. As XMPP implementations are required to support Unicode 3.2 for StringPrep during SASL authentication, no new requirements arise.

3 Use Cases

- As a user sending an instant message to a friend, I want to be able to emphasize an important part of my message.
- As a software developer, I want to be able to send code as pre-formatted, monospace, block or inline text to another developer.
- As a user of non-Latin script, I want to be surprised by inconsistent transitions and missing range checks resulting in characters from random Unicode planes.
- As a visually impaired user, I want my screen reader to emit long sequences of technical names of Unicode special characters to really convey the point of the intended emphasis, character by character!
- As a multi-user chat user I want to add context to my reply by quoting an earlier message in the chat.

[1] XEP-0071: XHTML-IM <https://xmpp.org/extensions/xep-0071.html>.
[2] XEP-0393: Message Styling <https://xmpp.org/extensions/xep-0393.html>.
[3] XEP-0394: Message Markup <https://xmpp.org/extensions/xep-0394.html>.

4 Business Rules

4.1 Preformatted Text

To format a string as Preformatted Text, the individual code-points need to be converted to their integer representation, then increased by 0x1D62F (capital letters) or 0x1D629 (lowercase letters) and then converted back to characters.
Preformatted text □ □□□□□□□□□□□□ □□□□

4.2 Emphasis

To format a string with Emphasis, the individual code-points need to be converted to their integer representation, then increased by 0x1D3F3 (capital letters) or 0x1D3ED (lowercase letters) and then converted back to characters.
Emphasis □ □□□□□□□□

4.3 Strong Emphasis

To format a string with Strong Emphasis, the individual code-points need to be converted to their integer representation, then increased by 0x1D3BF (capital letters) or 0x1D3B9 (lowercase letters) and then converted back to characters.
Strong Emphasis □ □□□□□□ □□□□□□□□

4.4 Very Strong Emphasis

To format a string with Very Very Strong Emphasis, the individual code-points need to be converted to their integer representation, then increased by 0x1D427 (capital letters) or 0x1D421 (lowercase letters) and then converted back to characters.
Very Strong Emphasis □ □□□□ □□□□□□ □□□□□□□□

4.5 Strike through

To format a string as struck through, each code-point needs to be extended with a U-0336 COMBINING LONG STROKE OVERLAY postfix.
Strike through □ S̶t̶r̶i̶k̶e̶ ̶t̶h̶r̶o̶u̶g̶h̶□

5 Implementation Notes

Some clients will use a serif-less font instead of a serif font to display instant messages, or even allow the user to choose an arbitrary font. The formatting defined in this document assumes the use of serif fonts, however. If a formatted text is to be displayed in a context that is not supposed to render serifs, the displaying implementation MUST add 0x1D4 to the respective integer representations of the fancy code-points.
On the wire, the serif representation always MUST be used.

Determining whether a given display font is serif or sans-serif is out of scope of this specification.

6 Accessibility Considerations

□□□□

7 Internationalization Considerations

□□□□

8 Security Considerations

The rendering of fonts is a complex task performed in the most inner guts of the operating system. Over the last decades, various exploits have been discovered in all major operating systems. Unfortunately, fixing those issues is far beyond the abilities of a simple XMPP client.

9 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [4].

10 XMPP Registrar Considerations

None.

[4] The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.