USE CASES (MSRP+MSRP-R Stack):

# License

Copyright (c) 2008 João André Pereira Antunes.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

# Status of this document

At the moment this is an incomplete document, the implementation decisions correspond to the work done as explained on: http://web.ist.utl.pt/~L54457/GSoC2008.

# Introduction

From the Abstract of my Google Summer of Code application:

*“Following a previously made academic project that consisted on developing an IMS (IP Multimedia Subsystem) that included an PTS (Push To Send) feature this project seemed like a very good follow up in which i already have some knowhow and experience.*

*What i will do is:*

*Implement and publish an open source MSRP Stack focused on the SIP Communicator needs but designed to be fully extensible therefore planting the seed to have a full implementation of the Stack available to the open source community, thus filling a gap there is on the open source community where no implementations in Java of the protocol are available and also none ongoing effort (as far as i can tell).*

*Use the MSRP Stack to implement the fully working functionality of file send and receive on SIP-Communicator.*

*[Optional] - fully enhance the MSRP Stack.”*

This is one of the products of the work of developing the MSRP Stack, for more info please visit http://web.ist.utl.pt/~L54457/GSoC2008.

# From the MSRP – R RFC[[1]](#endnote-2):

### Client connecting to relays acting on client’s behalf[[2]](#footnote-2):

Detailed description:

|  |  |
| --- | --- |
| **Client connecting to relays acting on client’s behalf (principal scenario)** | |
| **Pre condition:** | API already knows the relay’s hierarchical list that will act on their behalf addresses  API already knows if the client has defined an expires for the uri or not for a safety matter |
| **Description:** | 1. This use case starts when the client requests this method of communicating 2. If the relays given are mrsps then the stack uses the TLS certificate to authenticate the identity of the first relay    1. If not what happens?! ATTENTION!![[3]](#footnote-3) 3. stack sends an AUTH method to the relay he wishes to authenticate to 4. it receives an 401 unauthorized response 5. it then reauthenticates seen that it knows the nonce 6. it receives an ok response with an use-path and eventually an expires for the use path 7. using the first relay as a tunnel it authenticates with all other relays using tunneled auth requests |
| **Alternative Path:** | If the authentication fails in any way (AUTH method or TLS) then it should throw an error (exception maybe?!) |
| **Possible external modules:** | * TLS connection establishment * HTTP digest authentication |
| **Post condition:** | API gives away the Use-path used in the To-Path for use in the SDP negotiation |

|  |  |
| --- | --- |
| **Client connecting to relays acting on client’s behalf (Main scenario)** | |
| **Pre condition:** |  |
| **Description:** |  |
|  |  |
|  |  |
| **Post condition:** |  |

# From the MSRP RFC[[4]](#endnote-3):

|  |  |
| --- | --- |
| **Method SEND of a generic message** | |
| **Pre condition:** | MSRP URIs known by the API  Accept types known by the API  Client already has a message defined thus: (Success-Report of the message known to the API, Failure-Report of the message known to the API, The hole size of the message is known to the API) |
| **Description:** | 1. Client generates a new message 2. The headers of the MSRP message are generated taking into account    * the already known URI’s    * the message id that is provided along with the chunk    * the Byte-range that is provided along with the chunk    * The message chunk and thus the message has a value of reporting associated with it that is used 3. In case we don’t have a connection already established with the host on the specified port on the URI:    * Extend: Create a connection. 4. Chunk of message plus headers get sent to the connection scheduler 5. The transaction id of the message along with other necessary info (message id?!)gets stored for tracking purposes |
|  |  |
| **Non functional requirements** | Storing the MSRP URI’s in a secure way so that it’s difficult to access by other clients |
| **Post condition:** | The message gets passed to the corresponding connection scheduler |

|  |  |
| --- | --- |
| **Method SEND of a generic chunk of a message(Main scenario)** | |
| **Pre condition:** | MSRP URIs known by the API  Accept types known by the API  Chunk of a message already created |
| **Description:** | 1. Include: Transaction ID generation 2. The headers of the MSRP message are generated taking into account    * the already known URI’s    * the message id that is provided along with the chunk    * the Byte-range that is provided along with the chunk    * The message chunk and thus the message has a value of reporting associated with it that is used 3. In case we don’t have a connection already established with the host on the specified port on the URI:    * Extend: Create a connection. 4. Chunk of message plus headers get sent to the connection scheduler 5. The transaction id of the message along with other necessary info (message id?!)gets stored for tracking purposes |
|  |  |
| **Non functional requirements** | Storing the MSRP URI’s in a secure way so that it’s difficult to access by other clients |
| **Post condition:** | The message chunk along with the header are sent through the network |

Normative actions based on the Success and Failure report values:

|  |  |
| --- | --- |
| **Transaction ID generation** | |
| **Pre condition:** | The whole body of the message is known |
| **Description:** | 1. A TID with “ at least 64 bits of randomness” get’s generated 2. Seven hyphens along with the TID that got generated and a continuation flag is searched for in the body of the message    1. If the previous combination was found, repeat the proccess |
| **Implementation Decisions references:** | TODOTODO page 18 |
|  |  |
| **Post condition:** | Returns a valid TID for this body |

# Further issues:

|  |  |
| --- | --- |
| Q. 1 – UDP and other protocol support | |
| **From:** | RFC 4975 – Page 15: “Since this document only specifies MSRP over TCP, all MSRP URIs herein use the "tcp" transport parameter. Documents that provide bindings on other transports should define respective parameters for those transports.  ” |
| **Question:** | What other protocols should we support? |
| **Further Action:** | Research to see if there is any document (RFC) describing the use of MSRP with UDP and if it is analyze it’s relevance. |
| **Rationale:** | The need to build a generic MSRP stack obliges me to analyze all the possible uses of the protocol. |

|  |  |
| --- | --- |
| Q. 2 – Usual accept-type to transfer files | |
| **From:** | RFC 4975 – Page 32: “**8.6. Content Type Negotiation**” |
| **Question:** | What are the content types used for file transfer?! (assume that our MSRP stack will be data agnostic) |
| **Further Action:** | Research to see if in the draft about transfer of files there is something there about this. (What is the CPIM format ref. 12 and 27 of the RFC 4975 does it has any significance to implement file transfer?!) |
| **Rationale:** | Even if our stack is data agnostic one will need this kind of info to implement file transfer in SIP-Communicator. |

|  |  |
| --- | --- |
| Q. 3 – HTTP Digest method library | |
| **From:** | RFC 4976 – Page 12: “**HTTP Digest authentication procedure (from** [**RFC 2617**](http://tools.ietf.org/html/rfc2617) **[**[**1**](http://tools.ietf.org/html/rfc4976#ref-1)**]).** ” |
| **Question:** | What library will one use to implement the HTTP Digest method?! |
| **Further Action:** | Will need to use a library that has the feature of HTTP Digest method. Check with fellow developers to assert exactly what is the best to use here, based on the limitations specified on pages 25/26 of RFC 4976, and consult RFC 2617 as needed. |
| **Rationale:** |  |

|  |  |
| --- | --- |
| Q. 4 – Relay testing | |
| **From:** | RFC 4976 |
| **Question:** | What should one do to test the compatibility of this MSRP stack with relays?,. |
| **Further Action:** | Check out the <http://msrprelay.org> they have a code for the relay check if one can use it to test the relay compatibility of this stack or not. |
| **Rationale:** | Seen that this implementation won’t provide for relays deployment but however it should be compatible with them one needs a way to test for this compatibility. |

|  |  |
| --- | --- |
| Q. 4 – HTTP Digest method how to use it | |
| **From:** | RFC 4976 – Page 25/26 |
| **Question:** | After choosing the library how should one do the methdo |
| **Further Action:** | Study pages 25/26, and go to RFC 2617 as needed |
| **Rationale:** |  |

|  |  |
| --- | --- |
| Q. 5 – TLS connections | |
| **From:** | RFC 4975 and 4975 |
| **Question:** | What library to implement the TLS connections and how to implement them |
| **Further Action:** | Ask fellow developers the best way to go about implementing this. |
| **Rationale:** | One could maybe have a tendency to use a very big library or maybe if Sip Comm already uses one library for this one should use it aswell to reduce the overall imports of SC |

|  |  |
| --- | --- |
| Q. 6 – Expires field | |
| **From:** | RFC 4976 |
| **Question:** | What happens when the session expires?! What if one is in the middle of a send of file?! |
| **Further Action:** | Recheck that the RFC 4976 doesn’t have any mention to it. After rechecking send an e-mail to the RFC author’s |
| **Rationale:** | It is an important question, however not an urgent one at this point seen that one will need to know this to implement the Relays functionality |

|  |  |
| --- | --- |
| Q. 7 – CPIM | |
| **From:** | **draft-ietf-mmusic-file-transfer-mech-08** |
| **Question:** | The draft shows as an example the use of the message/CPIM as the wrapper for the file. What is this format? Does the implementation of this format changes the stack implementation?! |
| **Further Action:** | Check the message/CPIM specification (Done RFC 3860 that got me specifically to RFC 3862 this is a wrapper with metadata to MIME messages)  Check what protocol is used to allow transfer of messages through SC. (sent a mail to Ivov) |
| **Rationale:** | Since we are building a generic implementation of MSRP but that which one of the purposes is to offer file transfer for S.C. one need to check if the implementation of the stack couldn’t be altered to better allow the implementation of this format. |
| **Results:** | it alters nothing if the stack keeps itslef data agnostic. (as expected) however i'm curious about what protocol is used in SC for messages (because alterations to the stack could be made in order to make it more compatible with SC in case SC already uses Message/CPIM format) |

# Implementation Decisions:

|  |  |
| --- | --- |
| ID 1 – Non-SEND (non-normative yet) requests | |
| **From:** | RFC 4975 – Page 17: “Non-SEND requests are not intended to carry message content, and are therefore not interruptible. Non-SEND request bodies MUST NOT be larger than 10240 octets.  Although this document does not discuss any particular usage of bodies in non-SEND requests, they may be useful in the future for carrying security or identity information, information about a message in progress, etc. The 10K size limit was chosen to be large enough for most of such applications, but small enough to avoid the fairness issues caused by sending arbitrarily large content in non-interruptible method bodies.” |
| **Question:** | Should the stack have support for this non-SEND requests? How should them be supported? |
| **Decision:** | I think it should have support for them. How?! Well, generalizing the design of the stack (choosing a good design pattern) so that adding other methods besides SEND and REPORT becomes easy for newer developers.  So this decision should be taken into consideration when defining the Class diagrams and objects of the stack |
| **Rationale:** | The objective of this work is to plant a seed to form a very complete stack of the MSRP and MSRP Relay implementation it would be going against that objective if we didn’t took into account the possibility of new methods. |

|  |  |
| --- | --- |
| ID 2 – body validation due to transaction id | |
| **From:** | RFC 4975 – Page 17: “Some implementations may choose to scan for the closing sequence as they send the body, and if it is encountered, simply interrupt the chunk at that point and start a new transaction with a different transaction identifier to carry the rest of the body. Other implementations may choose to scan the data and ensure that the body does not contain the transaction identifier before they start sending the transaction.” |
| **Question:** | Should our implementation scan the body on the fly or assume we have a previously known body when we generate the TID? |
| **Decision:** | We should assume we have a previously already known body when we generate the TID. |
| **Rationale:** | The on the fly implementation would be more complicated and would only make sense if we get some content which we are getting on the fly, like a stream, even in that context one can break it into pieces and generate a TID for the already known content, however not probably without having to relax some time requirements.  This work is intended to result in a very generalized stack implementation however having a TID generated on the fly seems to only be of some use to some specific implementations of the protocol and would add increased complexity to the code. |

|  |  |
| --- | --- |
| ID 3 – timer attached to the success report?! | |
| **From:** | RFC 4975 – Page 19: “If success reports are requested, i.e., the value of the Success-Report header field is "yes", the sending device MAY wish to run a timer of some value that makes sense for its application and take action if a success report is not received in this time. There is no universal value for this timer. For many IM applications, it may be 2 minutes while for some trading systems it may be under a second. Regardless of whether such a timer is used, if the success report has not been received by the time the session is ended, the device SHOULD inform the user.” |
| **Question:** | Should we consider such a timer for implementation? If so how does the value of it should be chosen? If it’s not chosen by the implementation where does our responsibilities end? |
| **Decision:** | The design of the classes should consider this.  The value of the timer should be left for the programmer to choose and be assigned to the message success report.  Also we need a way to tell when “the session is over” so there is a new concept of session that must be considered in the implementation |
| **Rationale:** | This is a generalized implementation of the MSRP so we should really allow the use of such a timer, however and like is said the value of it should really be left for the programmer to decide it depends a lot on the context of the application. |

|  |  |
| --- | --- |
| ID 4 – Connection Scheduler | |
| **From:** | RFC 4975 – Page 21: “SEND requests larger than 2048 octets MUST be interrupted if the sender needs to send pending responses or REPORT requests. If multiple SEND requests from different sessions are concurrently being sent over the same connection, the device SHOULD implement some scheme to alternate between them such that each concurrent request gets a chance to send some fair portion of data at regular intervals suitable to the application.” |
| **Question:** | Should this be implemented? How? |
| **Decision:** | Yes it should.  Every SEND request larger than 2048 octets will have a \* character in the end-field byte-range header (that makes it interruptible)  From this there is a new entity, per connection, let’s call it the connection scheduler, that will manage these requests and eventually interrupt the interruptible ones! |
| **Rationale:** | Having all the send requests having a size of less or equal than 2048 bytes would eliminate the need of having this Connection Scheduler, however for file transfers we would end up having a big overhead of bytes (headers and stuff) specially if we get a lot of hops as it is possible when using relays. |

|  |  |
| --- | --- |
| ID 5 – Stream mode support | |
| **From:** | RFC 4975 – Page 20: “The total field MAY contain a "\*" if the total size of the message is not known in advance.” |
| **Question:** | Should this work consider this kind of messages (streams)? |
| **Decision:** | Yes it should.  In a design point of view it should be considered.  In an implementation point of view to work with SIP-Communicator for file transfer is irrelevant |
| **Rationale:** | Not allowing this kind of messages to be sent would kill a feature of the MSRP protocol in this implementation that is the ability to send streams.  This kind of functionality is not needed by the SIP-Communicator. |

|  |  |
| --- | --- |
| ID 6 – REPORT with body?! | |
| **From:** | RFC 4975 – Page 22: “REPORT requests will normally not include a body, as the REPORT request header fields can carry sufficient information in most cases. However, REPORT requests MAY include a body containing additional information about the status of the associated SEND request. Such a body is informational only, and the sender of the REPORT request SHOULD NOT assume that the recipient pays any attention to the body.” |
| **Question:** | Should the body be considered when receiving a report or when generating one? |
| **Decision:** | No. Although some method could be considered in posterior implementations to retrieve and insert them. |
| **Rationale:** | There is really no apparent gain other than being more on implementing a feature that would allow us to inspect the body of a REPORT atm. |

|  |  |
| --- | --- |
| ID 7 – success reports granularity | |
| **From:** | RFC 4975 – Page 22: “The receiver MAY wait until it receives the last chunk of a message, and send a success report that covers the complete message.  Alternately, it MAY generate incremental success REPORTs as the chunks are received. These can be sent periodically and cover all the bytes that have been received so far, or they can be sent after a chunk arrives and cover just the part from that chunk.” |
| **Question:** | What should be the control that the API gives about this?! |
| **Decision:** | Well, at this point it seems that this kind of control is better to be implemented through a parameter. The parameter could be from which percentage variations of the size of the received message one should send the success report, another possibility, and these are not mutually exclusive is to also allow a definition from x to x kBytes received |
| **Rationale:** | This is a very useful feature that can really improve the user friendliness of the application that uses the protocol. Also the percentage seems like a very good method to define the granularity to maintain the same user friendliness despite the size of the message |

|  |  |
| --- | --- |
| ID 8 – receiving buffers of data | |
| **From:** | RFC 4975 – Page 26: “There are situations in which the receiver may not be able to give precedence to the last chunk received when chunks overlap. For example, the recipient might incrementally render chunks as they arrive. If a new chunk arrives that overlaps with a previously rendered chunk, it would be too late to "take back" any conflicting data from the first chunk. Therefore, the requirement to give precedence to the most recent chunk is specified at a "SHOULD" strength. This requirement is not intended to disallow applications where this behavior does not make sense.” |
| **Question:** | How should the stack handle this question?! How big should the buffer be?! |
| **Decision:** | I think it’s best to let this be a per session parameter but with a given default value that fits better! (The byte-size of the message could give a hint, however one should be aware of too big byte-sizes as advised on the Security Considerations of RFC 4975) |
| **Rationale:** | The API should be really flexible on this so that it can be used on several apps |

|  |  |
| --- | --- |
| ID 9 – data agnostic?! | |
| **From:** | RFC 4975 – Page 26: “What is done with the body is outside the scope of MSRP and largely determined by the MIME Content-Type and Content-Disposition. The body MAY be rendered after the whole message is received or partially rendered as it is being received.” |
| **Question:** | How our implementation of MSRP should consider this?! |
| **Decision:** | Let’s be data agnostic to some level, just store the data in byte arrays along with the mime-type.  For sending files one can use external libraries like Java Mime Magic Library (<http://jmimemagic.sourceforge.net/>) or Mime Util (<http://sourceforge.net/projects/mime-util>) to find out the MIME type of the file. |
| **Rationale:** | We have to be flexible. |

|  |  |
| --- | --- |
| ID 10 – Threaded connection scheduler that can interrupt sessions | |
| **From:** | RFC 4975 – Page 27: “It is possible to receive a failure report or a failure transaction response for a chunk that is currently being delivered. In this case, the entire message corresponding to that chunk SHOULD be aborted, by including the "#" character in the continuation field of the end-line.” |
| **Question:** | How should the connection scheduler be modified? |
| **Decision:** | Let’s make it read and write on the socket as long as data is present and make possible for SEND requests to be interrupted |
| **Rationale:** | It only makes sense |

|  |  |
| --- | --- |
| ID 11 – not valid REPORT requests | |
| **From:** | RFC 4975 – Page 27: “It is possible that an endpoint will receive a REPORT request on a session that is no longer valid. The endpoint's behavior if this happens is a matter of local policy.” |
| **Question:** | What should be our “local policy” |
| **Decision:** | Silently ignore those REPORTs. |
| **Rationale:** | It’s easier. |

|  |  |
| --- | --- |
| ID 12 – SDP: m-line | |
| **From:** | RFC 4975 – Page 27: “m-line for purposes of backwards compatibility with conventional SDP usages. While MSRP could theoretically carry any media-type, "message" is appropriate..” |
| **Question:** | What to do about the media content lines “m-line” |
| **Decision:** | If the stack will ever generate the SDP only retrieve m-lines with message types. If you are going to validate it don’t allow it to be any other type. |
| **Rationale:** | No real need to alter the type of it, subsequent needed alterations to this field shouldn’t be difficult to implement. |

|  |  |
| --- | --- |
| ID 13 – SDP: Update method | |
| **From:** | RFC 4975 – Page 31: “Either peer may initiate an updated exchange at any time. The endpoint that sends the new offer assumes the role of offerer for all purposes. The answerer MUST respond with a path attribute that represents a valid path to itself at the time of the updated exchange. This new path may be the same as its previous path, but may be different. The new offerer MUST NOT assume that the peer will answer with the same path it used previously.” |
| **Question:** | How to implement this update of the session?! |
| **Decision:** | How about having something that alters the session and then gives back the new SDP to be exchanged. |
| **Rationale:** |  |

|  |  |
| --- | --- |
| ID 14 – Allowing for a denial of a receipt of a message with a reason | |
| **From:** | RFC 4975 – Page 38: Response code descriptions |
| **Question:** | How to deny some of the existing session inputs |
| **Decision:** | Implement a method to the API where you have to give a reason to deny a certain message or session that will map to a response code. |
| **Rationale:** | It would be a very bad principle to ignore the fact that one has a lot of response codes that could give good hints of why things are not working |

|  |  |
| --- | --- |
| ID 15 – CPIM compatibility | |
| **From:** | RFC 4975 – Page 49: 13. CPIM Compatiblity |
| **Question:** | What is CPIM should the stack be compatible?! When they say that they must recognize some things like the From To etc, does that need to be implemented on the stack if we want the CPIM compatibility? |
| **Decision:** | (Go investigate further) |
| **Rationale:** |  |

|  |  |
| --- | --- |
| ID 16 – Allow the confirmation of a TLS peer to peer session with use of the fingerprint exchanged via SIP | |
| **From:** | RFC 4975 – Page 52: “When used with SDP and SIP, the correct  certificate can be verified by passing a fingerprint of the  certificate in the SDP and ensuring that the SDP has suitable  integrity protection.  “ |
| **Question:** | How should be implemented in our stack?! |
| **Decision:** | When setting up a new session one could pass the fingerprint to that method. (…) (-how about setting up a connection receiving the SDP?! Also replying with an SDP-) |
| **Rationale:** |  |

|  |  |
| --- | --- |
| ID 17 – idle connection to the relay’s expiration time. | |
| **From:** | RFC 4976 – Page18: “For connections to relays, the client should leave a connection up until no sessions have used it for a locally defined period of time, which defaults to 5 minutes for foreign relays and one hour for the client's relays.“ |
| **Question:** | Should we provide a method to change this default time? |
| **Decision:** | Yes, will provide one that allows to change both parameters, the idle connection time to a intra relay, to a foreign and also to a peer (comments needed, need to be careful because this could allow for the increased security risk of DoS as described in RFC 4975 [using up one’s TCP ports]) |
| **Rationale:** | Want to implement a very generic Stack that supports MSRP and MSRP relays so ideally one should change all of the parameters. |

|  |  |
| --- | --- |
| ID 18 – stack that will allow relays?!. | |
| **From:** | RFC 4976 |
| **Question:** | Should the stack also support for the deployment of MSRP relays |
| **Decision:** | No |
| **Rationale:** | The primary idea is to allow support of MSRP endpoints, also the project timeline is tricky as it is. One should not say that this code couldn’t be extended to be used by an MSRP relay however it is completely out of the scope ATM, for test purposes with relays see Q4 |

# Glossary:

Session – in this context it’s considered that each complete URI represents a session.

Connection – in this context one connection can be shared among various sessions, the connection is represented by the part of the URI that has the protocol, host and port more concretely: protocol://host:port/RestOfURI. In the MSRP protocol two protocol identifiers are accepted: msrps and msrp one corresponding to the secure version of the protocol and the other to the non secure version, respectively.

# GNU Free Documentation License

GNU Free Documentation License

Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other

functional and useful document "free" in the sense of freedom: to

assure everyone the effective freedom to copy and redistribute it,

with or without modifying it, either commercially or noncommercially.

Secondarily, this License preserves for the author and publisher a way

to get credit for their work, while not being considered responsible

for modifications made by others.

This License is a kind of "copyleft", which means that derivative

works of the document must themselves be free in the same sense. It

complements the GNU General Public License, which is a copyleft

license designed for free software.

We have designed this License in order to use it for manuals for free

software, because free software needs free documentation: a free

program should come with manuals providing the same freedoms that the

software does. But this License is not limited to software manuals;

it can be used for any textual work, regardless of subject matter or

whether it is published as a printed book. We recommend this License

principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that

contains a notice placed by the copyright holder saying it can be

distributed under the terms of this License. Such a notice grants a

world-wide, royalty-free license, unlimited in duration, to use that

work under the conditions stated herein. The "Document", below,

refers to any such manual or work. Any member of the public is a

licensee, and is addressed as "you". You accept the license if you

copy, modify or distribute the work in a way requiring permission

under copyright law.

A "Modified Version" of the Document means any work containing the

Document or a portion of it, either copied verbatim, or with

modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of

the Document that deals exclusively with the relationship of the

publishers or authors of the Document to the Document's overall subject

(or to related matters) and contains nothing that could fall directly

within that overall subject. (Thus, if the Document is in part a

textbook of mathematics, a Secondary Section may not explain any

mathematics.) The relationship could be a matter of historical

connection with the subject or with related matters, or of legal,

commercial, philosophical, ethical or political position regarding

them.

The "Invariant Sections" are certain Secondary Sections whose titles

are designated, as being those of Invariant Sections, in the notice

that says that the Document is released under this License. If a

section does not fit the above definition of Secondary then it is not

allowed to be designated as Invariant. The Document may contain zero

Invariant Sections. If the Document does not identify any Invariant

Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed,

as Front-Cover Texts or Back-Cover Texts, in the notice that says that

the Document is released under this License. A Front-Cover Text may

be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy,

represented in a format whose specification is available to the

general public, that is suitable for revising the document

straightforwardly with generic text editors or (for images composed of

pixels) generic paint programs or (for drawings) some widely available

drawing editor, and that is suitable for input to text formatters or

for automatic translation to a variety of formats suitable for input

to text formatters. A copy made in an otherwise Transparent file

format whose markup, or absence of markup, has been arranged to thwart

or discourage subsequent modification by readers is not Transparent.

An image format is not Transparent if used for any substantial amount

of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain

ASCII without markup, Texinfo input format, LaTeX input format, SGML

or XML using a publicly available DTD, and standard-conforming simple

HTML, PostScript or PDF designed for human modification. Examples of

transparent image formats include PNG, XCF and JPG. Opaque formats

include proprietary formats that can be read and edited only by

proprietary word processors, SGML or XML for which the DTD and/or

processing tools are not generally available, and the

machine-generated HTML, PostScript or PDF produced by some word

processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself,

plus such following pages as are needed to hold, legibly, the material

this License requires to appear in the title page. For works in

formats which do not have any title page as such, "Title Page" means

the text near the most prominent appearance of the work's title,

preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose

title either is precisely XYZ or contains XYZ in parentheses following

text that translates XYZ in another language. (Here XYZ stands for a

specific section name mentioned below, such as "Acknowledgements",

"Dedications", "Endorsements", or "History".) To "Preserve the Title"

of such a section when you modify the Document means that it remains a

section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which

states that this License applies to the Document. These Warranty

Disclaimers are considered to be included by reference in this

License, but only as regards disclaiming warranties: any other

implication that these Warranty Disclaimers may have is void and has

no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either

commercially or noncommercially, provided that this License, the

copyright notices, and the license notice saying this License applies

to the Document are reproduced in all copies, and that you add no other

conditions whatsoever to those of this License. You may not use

technical measures to obstruct or control the reading or further

copying of the copies you make or distribute. However, you may accept

compensation in exchange for copies. If you distribute a large enough

number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and

you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have

printed covers) of the Document, numbering more than 100, and the

Document's license notice requires Cover Texts, you must enclose the

copies in covers that carry, clearly and legibly, all these Cover

Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on

the back cover. Both covers must also clearly and legibly identify

you as the publisher of these copies. The front cover must present

the full title with all words of the title equally prominent and

visible. You may add other material on the covers in addition.

Copying with changes limited to the covers, as long as they preserve

the title of the Document and satisfy these conditions, can be treated

as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit

legibly, you should put the first ones listed (as many as fit

reasonably) on the actual cover, and continue the rest onto adjacent

pages.

If you publish or distribute Opaque copies of the Document numbering

more than 100, you must either include a machine-readable Transparent

copy along with each Opaque copy, or state in or with each Opaque copy

a computer-network location from which the general network-using

public has access to download using public-standard network protocols

a complete Transparent copy of the Document, free of added material.

If you use the latter option, you must take reasonably prudent steps,

when you begin distribution of Opaque copies in quantity, to ensure

that this Transparent copy will remain thus accessible at the stated

location until at least one year after the last time you distribute an

Opaque copy (directly or through your agents or retailers) of that

edition to the public.

It is requested, but not required, that you contact the authors of the

Document well before redistributing any large number of copies, to give

them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under

the conditions of sections 2 and 3 above, provided that you release

the Modified Version under precisely this License, with the Modified

Version filling the role of the Document, thus licensing distribution

and modification of the Modified Version to whoever possesses a copy

of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct

from that of the Document, and from those of previous versions

(which should, if there were any, be listed in the History section

of the Document). You may use the same title as a previous version

if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities

responsible for authorship of the modifications in the Modified

Version, together with at least five of the principal authors of the

Document (all of its principal authors, if it has fewer than five),

unless they release you from this requirement.

C. State on the Title page the name of the publisher of the

Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications

adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice

giving the public permission to use the Modified Version under the

terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections

and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add

to it an item stating at least the title, year, new authors, and

publisher of the Modified Version as given on the Title Page. If

there is no section Entitled "History" in the Document, create one

stating the title, year, authors, and publisher of the Document as

given on its Title Page, then add an item describing the Modified

Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for

public access to a Transparent copy of the Document, and likewise

the network locations given in the Document for previous versions

it was based on. These may be placed in the "History" section.

You may omit a network location for a work that was published at

least four years before the Document itself, or if the original

publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications",

Preserve the Title of the section, and preserve in the section all

the substance and tone of each of the contributor acknowledgements

and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document,

unaltered in their text and in their titles. Section numbers

or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section

may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements"

or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or

appendices that qualify as Secondary Sections and contain no material

copied from the Document, you may at your option designate some or all

of these sections as invariant. To do this, add their titles to the

list of Invariant Sections in the Modified Version's license notice.

These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains

nothing but endorsements of your Modified Version by various

parties--for example, statements of peer review or that the text has

been approved by an organization as the authoritative definition of a

standard.

You may add a passage of up to five words as a Front-Cover Text, and a

passage of up to 25 words as a Back-Cover Text, to the end of the list

of Cover Texts in the Modified Version. Only one passage of

Front-Cover Text and one of Back-Cover Text may be added by (or

through arrangements made by) any one entity. If the Document already

includes a cover text for the same cover, previously added by you or

by arrangement made by the same entity you are acting on behalf of,

you may not add another; but you may replace the old one, on explicit

permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License

give permission to use their names for publicity for or to assert or

imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this

License, under the terms defined in section 4 above for modified

versions, provided that you include in the combination all of the

Invariant Sections of all of the original documents, unmodified, and

list them all as Invariant Sections of your combined work in its

license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and

multiple identical Invariant Sections may be replaced with a single

copy. If there are multiple Invariant Sections with the same name but

different contents, make the title of each such section unique by

adding at the end of it, in parentheses, the name of the original

author or publisher of that section if known, or else a unique number.

Make the same adjustment to the section titles in the list of

Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History"

in the various original documents, forming one section Entitled

"History"; likewise combine any sections Entitled "Acknowledgements",

and any sections Entitled "Dedications". You must delete all sections

Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents

released under this License, and replace the individual copies of this

License in the various documents with a single copy that is included in

the collection, provided that you follow the rules of this License for

verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute

it individually under this License, provided you insert a copy of this

License into the extracted document, and follow this License in all

other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate

and independent documents or works, in or on a volume of a storage or

distribution medium, is called an "aggregate" if the copyright

resulting from the compilation is not used to limit the legal rights

of the compilation's users beyond what the individual works permit.

When the Document is included in an aggregate, this License does not

apply to the other works in the aggregate which are not themselves

derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these

copies of the Document, then if the Document is less than one half of

the entire aggregate, the Document's Cover Texts may be placed on

covers that bracket the Document within the aggregate, or the

electronic equivalent of covers if the Document is in electronic form.

Otherwise they must appear on printed covers that bracket the whole

aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may

distribute translations of the Document under the terms of section 4.

Replacing Invariant Sections with translations requires special

permission from their copyright holders, but you may include

translations of some or all Invariant Sections in addition to the

original versions of these Invariant Sections. You may include a

translation of this License, and all the license notices in the

Document, and any Warranty Disclaimers, provided that you also include

the original English version of this License and the original versions

of those notices and disclaimers. In case of a disagreement between

the translation and the original version of this License or a notice

or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements",

"Dedications", or "History", the requirement (section 4) to Preserve

its Title (section 1) will typically require changing the actual

title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except

as expressly provided for under this License. Any other attempt to

copy, modify, sublicense or distribute the Document is void, and will

automatically terminate your rights under this License. However,

parties who have received copies, or rights, from you under this

License will not have their licenses terminated so long as such

parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions

of the GNU Free Documentation License from time to time. Such new

versions will be similar in spirit to the present version, but may

differ in detail to address new problems or concerns. See

http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number.

If the Document specifies that a particular numbered version of this

License "or any later version" applies to it, you have the option of

following the terms and conditions either of that specified version or

of any later version that has been published (not as a draft) by the

Free Software Foundation. If the Document does not specify a version

number of this License, you may choose any version ever published (not

as a draft) by the Free Software Foundation.

1. http://tools.ietf.org/html/rfc4976 [↑](#endnote-ref-2)
2. Section 5.1 of RFC 4976 (<http://tools.ietf.org/html/rfc4976#section-5>) [↑](#footnote-ref-2)
3. Responses here: http://msrprelay.org/chrome/site/files/msrprelay-messageflows.pdf [↑](#footnote-ref-3)
4. http://tools.ietf.org/html/rfc4975 [↑](#endnote-ref-3)