IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 802.11 Editors Guide | | | | |
| Date: 2015-09-15 | | | | |
| Author(s): | | | | |
| Name | Company | Address | Phone | email |
| Adrian Stephens | Intel Corporation |  |  | Adrian.p.stephens@intel.com |
| Dan Gal | Alcatel-Lucent |  |  | dan.gal@alcatel-lucent.com |

Abstract

This document contains material relevant to the job of being an 802.11 editor.

It is recommended that editors read this material before they start, as it may avoid them needlessly re-inventing the wheel.

R1: Updated by Henry Ptasinski re OpenOffice/LibreOffice. Updated by Adrian Stephens in response to review comments from Henry.

R4: More thoughts on choice of editing tool.

# Introduction

## Purpose of Document

This document is intended to help 802.11 working/task group editors perform their job more effectively.

It describes things an editor needs to know and processes to be followed.

## Duties of a Task Editor

An 802.11 TG editor has the following responsibilities:

* Implement approved comment resolutions in a timely fashion
* (optional, to be decided by TG chair) Look after comment resolution data in the task group
* Provide draft sources to TG editor on each version balloted (this is for disaster recovery purposes)
* Attend the Editor’s Meeting (7:00am Tuesday on the week of an 802.11 session)
* Related to 11MC (The 802.11 Mandatory Coordination process)
  + With the WG editor, to identify when 11MC is needed and request the TG editor to perform 11MC
  + To co-author, review and respond to the 11MC report with a goal of reaching a consensus that is likely to be acceptable to the TG
  + To bring the 11MC report to the TG for discussion and approval of changes identified in it
* Related to MEC
  + With the WG editor, to identify when MEC is needed
  + To respond to MEC comments from the IEEE-SA
  + To make any changes required by the IEEE-SA prior to entry to sponsor ballot
  + To document the MEC response from the IEEE-SA and the resulting changes made.
* Related to publication
  + To provide sources of the last draft to the IEEE-SA in a timely fashion
  + To review changes proposed by the publication editor and respond with any issues
  + To respond to questions from the publication editor

## Duties of a Working Group Editor

In addition to multiple task group editors, the working group appoints (strictly, the WG chair appoints, and the WG confirms) one or more WG editors to perform the following duties:

* Manage the operation of the TG editors.
* Provide status reports to the WG chair
* Attend the WG chair’s CAC meeting
* Run the Editor’s meeting
* Interface between the IEEE-SA editors and TG editors:
  + Submit drafts for MEC
  + Submit final draft sources for publication editing
  + Ensure that the TG editor responds appropriately during the publication editing process
  + Communicate any changes in IEEE-SA rules or style to the TG editors

## IEEE 802.11 Editor’s Meeting

The editor’s meeting takes place at 7:00am on Tuesdays during an 802.11 session. It is the one venue when all the 802.11 editors get together.

Attendance credit is given for attending the meeting.

It is a requirement that all editors attend this meeting, if at all possible.

Others may attend the meeting.

The agenda is typically:

* Check contact details
* Review status of each group
* Make any adjustment to amendment ordering
* Discuss and resolve any editorial issues
* Review matters of style or process development
* Help new editors get up to speed and understand the 802.11 editorial processes

## Relationship with IEEE-SA

IEEE 802.11 editors operate as volunteers, subject to the policies and procedures of the working group (802.11), the LMSC (802) and the IEEE-SA.

For an editor, there are the following areas of interaction with the IEEE-SA:

* Complying with IEEE-SA style – you must read the style guide!
* Responding to MEC comments
* Participating in the publication editing process as reviewer

# Style

Style is great. Like hygiene, it’s easiest to recognize when it is absent :0).

Style is there to ensure clarity and lack of ambiguity in the way normative requirements are stated, as well as to ensure the “least surprise” when somebody opens a new IEEE standard. That is why certain things have to be done in certain ways. Certain words used in certain contexts, and others avoided.

## IEEE-SA Style

Refer to the [IEEE-SA style guide](https://development.standards.ieee.org/myproject/Public/mytools/draft/styleman.pdf).

There really is no substitute for reading this document. If you don’t understand it, you will be doing things wrong, and that will cost you time and effort to put right later in the process. So, if you have any questions, ask the WG editors for explanation.

## WG11 Style

The [WG11 style document](https://mentor.ieee.org/802.11/documents?is_dcn=1034&is_year=2009) (11-09/1034) describes stylistic conventions that WG11 has developed over a number of years.

These conventions ensure homogeneity between an amendment and its baseline such that there should be no apparent discontinuity of style between material that has been there since the founding fathers, and material that is hot off the IEEE-SA press.

Editors are strongly advised to familiarize themselves with the WG11 style document, as compliance to this style document is reviewed during the 11MC process, and errors will need to be addressed before entry to sponsor ballot. It is better to avoid these mistakes than correct them half way through the lifetime of the draft.

# Choice between Frame, Word, Openoffice/Libreoffice

The IEEE-SA will accept documents in Frame, Word or OpenOffice formats.

Frame was initially chosen by the IEEE-SA probably because Word was unsuitable for technical documentation of this kind.   Specifically, Word did not handle long documents well.  It may be that these reasons are less applicable these days.  But the IEEE-SA document processes (e.g., amendment numbering style) have been designed to match the capabilities of Frame.

Frame also allows control of the visibility of conditional text and global changes to the style of cross references, which can be useful when preparing a draft for ballot.

Amendments with embedded equations should be done in frame, because any such equations have to be manually re-created in frame on rolling-in the Amendment. This can introduce error.

The IEEE 802.11 base standard is a Frame document.

The downside of using Frame is that there's a fair amount of manual copy and paste (and adjustment) required to get text from approved submissions into the draft, and it is even harder for submission authors to get material from the draft.

The advantage of using Word is that it is more commonly available, and incorporation of submission text is easier. The disadvantages of word include:

* Word will not generate amendment-style numbering “3.3a.1”. This means that cross references to these items are invariably manual text inserts, which means they stand a good chance of becoming outdated.
* Word native graphics and equations have to be manually re-drawn when the amendment is rolled in to the revision.

On balance, it is recommended that Frame is used for all amendments of length > 100 pages, or amendments with any equations.

The comments for Word apply also to OpenOffice. Acceptance of documents in OpenOffice format is a recent (writing in 2011) change by the IEEE-SA. Note: Libreoffice is a fork of the original OpenOffice development. Both projects are continuing development. Unless otherwise noted, all comments about OpenOffice apply to Libreoffice and any other derivatives.

## Which source file format versions are acceptable to IEEE-SA?

Frame support in the IEEE-SA languished at version 7.2 for many years. They have recently adopted Frame 8 and then Frame 9 in short order. They can take Frame in any of these versions, but will now give it back only in Frame 9 format.

Unless there is a good reason, the latest version of Frame should be used.

There are no IEEE-SA rules about the Word document format (“doc” vs “docx”) is acceptable. Both should be acceptable because Office 2007 is now widely deployed.

There are likewise no rules about versions / types of OpenOffice/LibreOffice document files. Any format that can be read by the current version should be acceptable.

# Creating your first draft

Creating your first draft will mean taking one or more submission documents and changing the formatting to the appearance of a draft.

Understand that this first draft is likely to require significant editorial work to adjust syntax, grammar, style. If this is also your first experience with Frame, there is a learning curve to go through.

## Resources available

Your most valuable resource is the other editors. If you need help getting through these learning curves, ask. We’ve all been there and all been baffled.

The IEEE-SA have quite a lot to say on developing standards. See: <http://standards.ieee.org/develop/>

### Templates

The IEEE-SA provides templates for both [Frame](https://development.standards.ieee.org/myproject/Public/mytools/draft/fm7temp.zip) and [Word](https://development.standards.ieee.org/myproject/Public/mytools/draft/wordtemp.zip) versions.

It is essential that the draft is created based on these templates, because they include styles for headings, body text and tables that match the IEEE style guide.

The Word templates contain an annoying set of macros to set up the front matters. It is recommended to use these matters to create an initial skeleton, and then cut and paste this into an empty file so that the macros no longer intrude on normal editing.

### 802.11 sources

If a TG draft is using Frame, the WG editor will provide, on request, access to the sources of the current revision, or last published, standard. This can provide a number of benefits:

1. Discover how certain visual effects are achieved (i.e. which of the template styles to use in which context).
2. Provide baseline text and graphics for subsequent modification

Note, these sources are provided only to support the development of the TG draft and shall not be used for any other purpose.

## Transitioning from Word into Frame

Transitioning from Word to Frame is non-trivial. Transitioning, at some point, may be necessary if the Initial Draft approved by the Task Group is in Word (e.g. assembled from various 802.11 submissions), but the target format is Frame (e.g. because it is > 100 pages).

Creating a redline between a Word version and a Frame version is nigh on impossible.

The conversion from Word to Frame includes the following steps:

* Reworking the front matter using the Frame template
* Building a book file with generated tables of content, figures, tables
* Adjusting paragraph numbering style for most headings
* Reworking embedded equation objects from some other program into native frame equations
* Replacing embedded graphics objects with references to external metafile or tiff files.

Not all of these have to happen initially. The last two steps are likely to happen gradually during the lifetime of the draft, but have to be completed before it is submitted for publication.

It is not really feasible to transition during the WG or sponsor ballot processes, because of the difficulty of creating the redline. There is no point in transitioning after sponsor ballot .

So the recommended places to transition are:

* Between D0.1 and D1.0, i.e., during a TG review process.
  + There are no particular issue with this. Although if the TG want to see incremental redlines, the conversion will have to wait until other changes from the TG have been incorporated.
  + The TG editor should allow time to convert the draft (1 week / 200 pages + 1 week), plus time to have the new draft reviewed against the predecessor into the timing of the start of WG ballot.
* Between WG ballot and sponsor ballot.
  + The main issue with this is that you have to enter sponsor ballot with the “same draft” you approved in WG ballot.
  + There is no particular problem entering with a higher revision number provided any changes are clearly editorial, although in practice it is better to minimize such changes to avoid any argument as to what is editorial and what is not.
  + However, to avoid any unintended changes, the editor will need to employ multiple pairs of eyes to do a page-by-page visual comparison of the two drafts.

# Editing a draft

## Editorial Notes

The editor may need to note in the draft issues discovered while attempting to incorporate new material or execute approved editing instructions.

It is recommended to include a section of the document to describe the format of these notes using something similar to the following language:

|  |
| --- |
| **Editor’s notes**  The editor’s notes do not form a part of this standard. They will be removed before publication. Please do not comment on editor’s notes in any ballot on the draft, as these comments would have no effect on the published standard.  *EDITOR’s NOTE -- Editor’s Notes in the body of the standard appear like this. They will be removed before publication. They may highlight some issue that the editor has had to address during the implementation of a change. Where there may be any technical impact from an editing issue, the editor will raise a technical letter ballot comment. There is no need for voters to comment on such issues unless they have a specific resolution they wish to present.* |

Trawl through a draft before posting and remove any now-superfluous notes.

## Tagging changes

It is recommended that changes made in an approved draft are tagged so that the reason for any change can be determined. This has two benefits:

1. It enables reviewers who are checking changes to quickly locate changes
2. It enables those writing comment resolutions to discover earlier comment resolutions that changed text in the scope of a current comment resolution. This may help them understand the rationale for an earlier change and be able to respond in a more informed fashion. In particular, it can help detect “thrash” between polar opposites.

The format of the tags is not hugely significant, provided that they are visually unambiguously tags. The Editors Notes section should describe the purpose and format using something like:

|  |
| --- |
| Tags  Tags are placed in this draft near changes to identify the source of the change. Changes resulting from incorporation of an amendment are shown like this: (11<letters>). Changes resulting from incorporation of an approved comment resolution are shown like this: (#<number>) where <number> identifies the comment resolution (i.e. is its CID). These tags will be hidden in versions of the draft sent out to letter ballot - i.e. they are present only to assist the editorial review panel in checking that changes have been properly applied.  Tags are shown close to the point of change. When a whole subclause is new, the heading is tagged. Otherwise, when a whole paragraph is new, the paragraph is tagged. Otherwise, tags are placed after a section of changes within a para, or at the end of the para if the changes are substantial.  New tables are tagged in the table caption (if there is one), or the introductory paragraph. Otherwise, new rows in existing tables are tagged only in the first column, to avoid distraction. Otherwise a modified cell is tagged.  Any other changes made by the editor (e.g. for grammar, language, style & consistency with other comment resolutions) are tagged (Ed) |

In Frame, the tags can be set as conditional text based on the “Condition Tag”=”Comment”.

In openoffice, conditional text can be used as described in: <http://help.libreoffice.org/Writer/Conditional_Text>.[[1]](#footnote-1)

Tags should not be visible in a draft sent to ballot. This means that they should either be removed or hidden. In Frame, the tags can be hidden by hiding the Condition Tag “Comment”.

It is recommended that the contents of the tag relate to approved changes in some way:

1. A reference to an approved motion in the TG (e.g. a motion number or date)
2. A reference to an approved submission (e.g. a document number)
3. A reference to an approved comment resolution (e.g. a unique comment identifier)

Be aware that, as editor, you have multiple options for keeping tags:

1. Keep all tags ever inserted in the document from its very birth to its publication
2. Clear out all tags before D1, and before the draft that goes to sponsor ballot
3. Clear out all tags before each balloted revision

**Historical Note:** The use of tags was introduced to 802.11 by Bill Marshall in 802.11r.

## Cross-reference style

The standard IEEE-SA cross reference style is shown below:

* See 3.4.5
* In Figure 8-125
* See Equation 9-3
* See Table 7-65

During all ballots except those near the end of sponsor ballot, an alternative style can be used:

* See 3.4.5 (Description of a BSS)
* In Figure 8-125 (Format of ACK frame)
* …

This enables reviewers/voters to more easily check that cross references are correct.

Note that the standard IEEE-SA cross reference style should be used in final sponsor ballots.

In Frame the switching can be achieved quite easily and quickly by editing the cross-reference style. If a draft has multiple documents, the style needs to be made in one of them and then imported into the others.

You can also have two blank documents each holding a different cross-reference style and then import (only) the cross reference formats from one of the two documents into all the book documents to effect a global switch.

It is not possible to easily switch or customize cross-reference style in OpenOffice. Therefore, it is recommended to start with the official IEEE-SA style and stick to it throughout the life of the draft.

## Creating a draft

The draft .pdf should ensure that all fonts are embedded. That way, it will look the same no matter on which system it is viewed. The following box shows how to do this with the Adobe Acrobat printer.

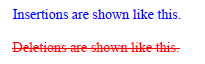
|  |
| --- |
| Update your Adobe Printer    Right click on the printer  Select printer preferences    Make sure Rely on system fonts only; do not use document fonts in unchecked  Click on Edit  Make sure compatibility under file options is set to Acrobat 8    Select the Fonts folder and highlight all of the fonts listed under the “never embed” tab and select “remove”  Finally…..”save as” |

## Creating a redline

### Redline Format

The IEEE-SA rules require us to identify changes in a recirculated draft, but don’t describe how those changes should be communicated.

We have evolved a WG11 style, which is readily achieved in both Frame and Word thus:



* Insertions are blue, without any added underline.
* Deletions are red, with strikeout
* These are the only colours and formats used – i.e., do not attempt to show different series of changes using a different colour. This is all too easy to obtain by mistake in Word if different people edit the draft.
* The gutter should show a change-bar at the point of change.

### Creating a Redline in Frame

There are many ways to skin a cat, particularly if the cat is written in Frame.

Here’s what I do. Others may skin their cats in a different way.

1. Copy all files from a release into a temporary directory. This involves copying all figures so that relative links to graphics files don’t get broken. I run the following bash script [[2]](#footnote-2) to do this:

|  |
| --- |
| # prepare\_cmp.bash  # prepare for comparison in CMP directory  fmfiles=`ls \*.fm`  mkdir -p CMP/Figures  rm -f CMP/\*.book  (cd CMP; rm -f $fmfiles)  rm -f CMP/Figures/\*  cp \*.fm CMP  cp \*.book CMP  cp Figures/\* CMP/Figures  rm -f CMP/\*.auto.fm  rm -f CMP/\*.lck |

1. Open the reference (older) .book file
2. Open the newer .book file
3. In Frame do File/Utilities/Compare Books
   1. Wait until this completes
   2. This might take a long time to complete. In REVmb, after some restructuring of the 200-page Annex C (MIB), a compare of the document and its predecessor failed to complete after 24 hours running time on a fast desktop machine. I think the running time is a polynomial of the file size.
      1. You can create a redline by exporting to Word, doing a document compare and then importing to Frame. This requires some manual work, but the execution time for the same compare as above was 2 seconds.
      2. Or you can split the document into multiple parts. REVmb ended up with Annex C split into 4 separate files.
4. Exit Frame
   1. Doesn’t matter about saving summary.fm
5. Frame when comparing two “fred.fm” files, Frame creates the redline in “fredCMP.fm”. You need to delete the new “fred.fm” and rename the \*CMP.fm file to “fred.fm”. That way you can re-open the book, and it now refers to the redline file. The following script does the renaming:

|  |
| --- |
| #!/bin/bash  ls \*CMP.fm | ( while read file  do  echo "Processing: $file"  rm -f "${file/CMP/}"  mv "$file" "${file/CMP/}"  done  ) |

1. You can now open the newer .book file, and it will show the redline contents.
2. At this point, create the .pdf from frame.
   1. Do not update the tables of contents or references at this time.

Reminder – only do this in a temporary directory. If you do it on your working sources the renaming in step 6 will trash them.

Frame (D8 onwards) does have “track changes” feature, however:

* We have no experience of using it in anger, so we don’t know what the gotchas are
* In D8 and D9, there is no way to select the format of the tracked changes, and they do not match the WG evolved style. Until somebody finds a workaround for this issue it cannot be used.

### Creating a Redline in Word

Word has two mechanisms to create a redline. Either work better and more conveniently than the Frame feature.

* Do editing with tracked changes switched on. Hide the tracked changes to produce the “clean” draft, and unhide them to produce the redline. To do that, open the Review menu tab, click open the Show Markup command item, then, open Reviewers, and check or uncheck the All Reviewers box to show or hide all the marked up changes.
* Accept all changes once a ballot is started, in preparation for the next draft.
* Do editing with Tracked Changes switched off. Compare previous and current revision to create a redline.

### Creating a Redline in Openoffice

Openoffice, like word, has two mechanisms to create a redline. Either work better and more conveniently than the Frame feature.

* Do editing with tracked changes switched on. Hide the tracked changes to produce the “clean” draft, and unhide them to produce the redline. Note that the ToC, List of Tables and List of Figures will not get updated automatically, so after hiding or showing changes, use Tools->Update->Update All to make sure the tables reflect what's currently being displayed. Also note that when changes are shown, these tables show both the old and new text for any changed items.
* Do editing with tracked changes switched off. Compare previous and current revision to create redline[[3]](#footnote-3).

## Treatment of graphics

Figures and Equations usually are entered as various flavours of embedded objects in Word submissions.

In the document submitted to the IEEE-SA, Figures and Equations are required to be[[4]](#footnote-4) one of various types of external file (e.g. TIFF, EPS, WMF, Visio ), or to use the Frame native figure and equation drawing features.

Note that amendments will be merged into the 802.11 baseline, which uses Frame. This explains why Word/OpenOffice/LibreOffice native equation/figures features should not be used as they do not import straightforwardly into Frame and would require the 802.11 editor to “mess around” with the objects to get them into the roll-up.

Note that you should keep the sources for external equations/figures and supply them to the 802.11 editor [[5]](#footnote-5) so that they can be re-created and edited once rolled into 802.11.

Recommendations:

* Use Visio for drawings
  + The native Frame drawing tool is very limited. I find it difficult to use.
  + Slowly convert embedded graphics objects to external .wmf files. It is much better to use .wmf than .tiff because it is a vector format, and doesn’t create aliasing artefacts between the .tiff pixel size and the printer screen size/angle.
  + I keep a directory for figures and another one for figure sources – i.e., each figure has its own .wmf file and its own .vsd file. To update a figure, edit the visio file and then do “save as” and select .wmf.
  + This can be automated using the following script: makefile and windows scripting host script:

|  |
| --- |
| # update\_figures.bash  # Creates wmf files from vsd sources  make -f update\_figures.makefile |

update\_figures.makefile

|  |
| --- |
| # 20090105 Adrian Stephens  # Note - requires Windows XP and cygwin make  # Note - figure filenames cannot contain embedded spaces  figdir = ../Figures/  wscript = /cygdrive/c/WINDOWS/system32/wscript  sources = $(shell ls \*.vsd)  pptsources = $(shell ls \*.ppt)  wmfs = $(sources:%.vsd=$(figdir)%.wmf) $(pptsources:%.ppt=$(figdir)%.wmf)  make\_only\_if\_missing = true  # If this variable is true, unlike in a normal makefile, only want to rebuild \*missing\* output files,  # regardless of the date on the sources. We do this by creating a  # "stamp" pseudo target that can be made either from the target or the source. The rule for  # making it from the source has the side effect of creating the target.  # The stamp file never actually exists.  stamps = $(sources:%.vsd=%.stamp) $(pptsources:%.ppt=%.stamp)  ifeq ($(make\_only\_if\_missing), true)  all: $(stamps)  else  all: $(wmfs)  endif    clean:  rm -f $(wmfs)  ifeq ($(make\_only\_if\_missing), true)  %.stamp: $(figdir)%.wmf    %.stamp: %.vsd  $(wscript) vsdtowmf.vbs $< $(<:%.vsd=$(figdir)%.wmf)  %.stamp: %.ppt  $(wscript) ppttowmf.vbs $< temp.wmf  mv temp/slide1.WMF $(<:%.ppt=$(figdir)%.wmf)  rmdir temp  else  $(figdir)%.wmf: %.vsd  $(wscript) vsdtowmf.vbs $< $@  $(figdir)%.wmf: %.ppt  $(wscript) ppttowmf.vbs $< temp.wmf  mv temp/slide1.WMF $@  rmdir temp  endif |

vsdtowmf.vbs:

|  |
| --- |
| ' Save vsd drawing to wmf format  dim vsdFile  dim wmfFile  vsdFile=Wscript.Arguments(0)  wmfFile=Wscript.Arguments(1)  dim WshShell  set WshShell=WScript.CreateObject("WScript.Shell")  ' Cygwin to windows filename conversion, and make references absolute  vsdFile=WshShell.CurrentDirectory + "\" + Replace(vsdFile,"/","\")  wmfFile=WshShell.CurrentDirectory + "\" + Replace(wmfFile,"/","\")  set fso=WScript.CreateObject("Scripting.FileSystemObject")  set visio=WScript.CreateObject("Visio.Application")  set document=visio.Documents.Open(vsdFile)  if fso.FileExists(wmfFile) then  fso.DeleteFile wmfFile  end if  visio.ActiveWindow.Page.Export wmfFile  visio.quit |

* Use native Frame Equations for equations
  + This tool is quite powerful, although counterintuitive and frustrating to drive

## Identify your baseline

An amendment is amending a baseline, which consists of either a published standard (IEEE Std 802.11-<year>) or an in-progress revision (P802.11REVm<letter>), plus any preceding amendments.

All numbering and quoted baseline text is based upon this baseline, i.e., potentially a long sequence of documents.

The specific versions of the baseline documents that an amendment changes are declared at the start of the amendment. This declared baseline might include a mixture of published and draft documents such as (.11s D5.0):

“(This amendment is based on IEEE Std 802.11™-2007, as amended by IEEE Std 802.11k™-2008, IEEE Std 802.11r™-2008, IEEE Std 802.11y™-2008, IEEE P802.11w™-2009, IEEE P802.11n™-2009, IEEE P802.11z D7.0, IEEE P802.11p D11.0, IEEE P802.11v D9.0, and IEEE P802.11u D8.03.)”

When the document is sent for approval, its baseline can consist of only published documents, e.g. (.11s D12):

“(This amendment specifies enhancements to IEEE Std 802.11™. It is based on IEEE Std 802.11™-2007, as amended by IEEE Std 802.11k™-2008, IEEE Std 802.11r™-2008, IEEE Std 802.11y™-2008, IEEE Std 802.11w™-2009, IEEE Std 802.11n™-2009, IEEE Std 802.11p™-2010, IEEE Std 802.11z™-2010, IEEE Std 802.11v™-2011, and IEEE Std 802.11u™-2011.)”

### Amendment ordering

The amendment order is determined by the Editor’s Meeting at interim meetings based on any updated timelines from task groups in the previous session.

Changing amendment ordering is a big deal. It can result in weeks of work as all affected drafts have to rework numbering and quoted baseline material. It can result in material moving from one draft to another due to dependency on that material.

It therefore behoves the editors to avoid needless swapping of ordering. They can do this within their own groups by ensuring that timescales are considered and reasonable – i.e., the group has a plan that has broad support outside the task group, and is consistent with previous execution history of the group.

See the [timeline](http://www.ieee802.org/11/Reports/802.11_Timelines.htm) for historical data.

Also if one draft has a dependency on material in another draft, and it is not within the scope of the first draft to embody that new material, the draft will need to be artificially delayed so that is it published after the draft on which it depends.

### When can I update my declared baseline?

Updating your baseline carries a cost because you need to review the redlines from your baseline in text you quote to make sure you reproduce any changes in your baseline text.

However, not updating your baseline also carries a cost as you may reference obsolete or known to be inaccurate information from your baselines. If different predecessor amendments quote wildly different baselines, it becomes increasingly difficult to accurately determine your own baseline.

It is recommended that you update your baseline to the last balloted revision of each of your dependencies once per ballot. A good time to do this is just after your ballot, when you are looking at editorial comment resolutions. Consider yourself resolving an implicit comment “update baseline to latest”.

Note, I would limit “latest” to mean the “latest balloted” revision. That should ensure that a proper redline is available from the previous balloted revision of one of your dependencies, and avoids any editing defects caught by review on that dependency between ballots.

### What changes in my baseline do I have to track?

Any change in your baseline that affects your draft must be reflected in your draft. There are two main areas:

1. Numbering (discussed in the next subclause)
2. Contents.

Examples of contents change are one of your dependencies adds a parameter to an MLME primitive. You now have to edit your draft and show that new parameter. The change is not shown as an editing instruction, because you are merely reflecting the current status of your baseline.

The change will, however, show up in the redline. This is the reason why the redline format does not include an underline, so that inserted baseline changes (blue, no underline) in your draft will show up in a “change” instruction in blue, no underline, while inserted text in a “change” instruction (in your draft) will show up as blue, underline.

## Numbering of Headings, Figures, Tables and Equations

Definitions:

* Dependency – something your draft is dependent on. This is the Standard plus any amendments preceding your draft in the current amendment order.
* Dependent – a draft coming after yours in the current amendment order.

### Numbering of Figures, Tables and Equations

In 802.11, numbering of Figures, Tables and Equations is reset per Clause (or Section), and includes the clause (section) number, plus a hyphen. This format limits the “ripple effect” from any change to be within a single clause (section).

Examples: Figure 8-127, Table 9-26, and Equation 19-12.

### Numbering style for amendments

In amendments, numbering follows IEEE-SA amendment style. Items inserted at the end of a numbering sequence in your baseline are given “regular” numbers. Don’t forget, your baseline includes all prior amendments in your sequence.

Examples:

* Subclause 3.3a inserted between baseline 3.3 and 3.4.
* Figure 8-127a inserted between baseline Figure 8-127 and 8-128
* Subclause 8.4.2.101 inserted after baseline clause 8.4.2.100 – there is no baseline clause of this number.
* Subclause 10.3.3.0a inserted before baseline subclause 10.3.3.1.

### Private vs “reasonable effort” numbering

In drafts sent to sponsor ballot, the editor should make a “reasonable effort” to ensure that the numbers are correct.

In drafts prior to working group ballot, groups often adopt a “private numbering space” – e.g. Figure 8-zz1. This is a reasonable starting point. The editor may not initially understand the amendment numbering rules, and his contributors more likely won’t. The publication order may be unclear or disputed.

However, at some point the editor will have to adjust the numbering to use “reasonable effort” numbering.

Also note that any private numbers used should match the number of dots/dashes in the “reasonable effort” number. Example:

* Private number: Figure 8-zz1, not Figure 26.
* Private number: 6.3.aa1, to be inserted somewhere in 6.3.x, not subclause aa1

Not sticking to this rule will make it very difficult to determine the correct numbering, when the time comes to make that transition.

### Relationship to the 802.11 Mandatory Draft Review (MDR) process

One of the review items in the MDR is an updated numbering spreadsheet.

If not already done so, a draft exiting the MDR process will have “reasonable effort” numbering, and the numbering spreadsheet will be updated.

### Numbering spreadsheet

#### Purpose of the spreadsheet

The numbering spreadsheet is the tool that we use to determine “reasonable effort” numbering. The data it holds can be used to determine unambiguously the correct numbering for a draft.

Of course, this is only correct if the data in the spreadsheet is correct. So editors have a distributed duty to maintain this data.

#### Contents of the spreadsheet

Some examples from 11-08-644r25, Clauses tab:





Here’s what the columns mean:

1. There are lots of columns to the left not shown.
2. The first column shows the number from .11s D12. This number bears no relation to the number it will have in REVmb when rolled in, shown in the next column, because of reordering of Clauses done in REVmb, and deletion of a number of subclauses by REVmb, which were part of the .11s baseline.
3. The second number is our best estimate of what the number will be when .11s is rolled into REVmb. (there’s an error, it should read D9, not D8).
4. The next column shows .11ae D3. This was still using a private numbering space.
5. The next columns shows .11ae D4. This has been adjusted for REVmb D8.
6. The next column (the “pink” column) shows an entry where the draft number does not match the value it should be based on all the current dependencies in the spreadsheet. In this case, because REVmb was updated to D9, which deleted a subclause 6.3.x, the numbering for .11ae needs to be adjusted. The header of the pink column shows the number of entries in the column. Non-blank entries are shaded an exquisite pink to highlight them.

Note, the column data typically includes a number of bugs in transcribing the numbers from a draft, as is shown above.

The most important data in the spreadsheet is the relative positioning of the rows, and the “numbering depth” (i.e. the number of dots) of a number.

#### Updating the spreadsheet

Updating the spreadsheet is a chore. I have developed some semi-automated tools to generate the pink numbering. Regenerating the pink numbering from scratch takes several hours, so editors are asked to maintain it incrementally once past the initial data-entry stage.

1. Initial data entry
   1. Add a column and declare your draft revision.
   2. Add entries for all subclauses, numbered tables, figures, numbered equations in your draft.
   3. Entries are one of two types:
      1. Where the baseline is being quoted, you will insert the number from your draft in the row corresponding to that item in the spreadsheet.
         1. Note, you should check the caption (where present) to check you’re inserting at the right location.
         2. The number you enter and the number of the baseline may differ if your baseline is a different version to the version on which the numbering spreadsheet is based. This does not matter.
      2. Where you are inserting a new numbered item, you insert a row in the spreadsheet at the appropriate location.
2. Initial pink number calculation
   1. This is probably best performed by the WG editor, during which he will iterate any questionable entries in your column with you.
3. Update revision
   1. When you update revision, cut and paste your previous revision column into a new column, then run down that column and correct numbering against the cited revision.
   2. Then check the new numbering agains the pink numbers. Your goal was probably to reduce the number of pink entries to zero. You may or may not have achieved that goal. Only remove pink entries that match your new declared draft number.
   3. If you inserted or deleted any rows, you need to propagate any changes to dependent drafts. These changes can “ripple” on through multiple dependent drafts, so don’t forget to check all drafts after you.

Consider the following example of a “ripple update”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TGa D5 |  | TGb D6 |  | TGc D4 |  |
| 6.3.84.8.4 |  |  |  |  |  |
|  |  | 6.3.85 |  |  |  |
|  |  | 6.3.85.1 |  |  |  |
|  |  | 6.3.85.2 |  |  |  |
|  |  |  |  | 6.3.86 |  |
|  |  |  |  | 6.3.86.1 |  |

TGa D6 inserts a new subclause 6.3.85 (in the highlighted row)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TGa D6 |  | TGb D6 |  | TGc D4 |  |
| 6.3.84.8.4 |  |  |  |  |  |
| 6.3.85 |  |  |  |  |  |
|  |  | 6.3.85 | 6.3.86 |  |  |
|  |  | 6.3.85.1 | 6.3.86.1 |  |  |
|  |  | 6.3.85.2 | 6.3.86.2 |  |  |
|  |  |  |  | 6.3.86 | 6.3.87 |
|  |  |  |  | 6.3.86.1 | 6.3.87.1 |

* 1. After propagating creating or updating any pink numbering, let the editors of the affected dependents know of the change – because this means they need to pay attention to the spreadsheet and update their numbering.

#### Source Control of the numbering spreadsheet

We will use an email lock via the editor’s reflector.

If you want the lock, email the editor’s reflector claiming the lock.

When you’re done, send the updated spreadsheet to the reflector. The WG editor may post it, but will certainly check it in to whatever source control he uses. Announce to the reflector that you have released the lock.

If you make changes without holding the lock it is possible you will lose that work and have to re-do it. You have been warned.

## Numbering of ANA-administered objects

The IEEE 802.11 Assigned Numbers Authority (ANA) is responsible for allocating certain resources related to the 802.11 standard.  These resources typically reflect bit positions in bitfields, or enumerated field values.  They are administered by the ANA when there is a possibility that multiple simultaneous amendments may try and modify the same resource to ensure that there are no collisions - i.e., attempted multiple uses of the same resource by different amendments.

Not all resources are administered by the ANA.

### ANA process and ANA database

The ANA process is described in the latest version of 11-07-0827. The "official" rules for operation of the ANA are contained in the latest version of 11-09-0002.

The ANA database is published as a spreadsheet, currently the latest version of document 11-11-0270.

Essentially the rules boil down to:

1. Identify which objects are administered
2. Initially use <ANA> as a flag for any objects inserted in administered namespaces
3. Ask the ANA to allocate values for these flags at an appropriate time.
4. Replace the <ANA> flags with the allocated values.

### Identifying which objects are administered

Look at the “TOC” tab of the ANA database, and you will see a list of administered resources.

### When should allocations be made

Allocations should be made prior to entry to sponsor ballot.

Allocations should not be made until the draft is sufficiently stable that there is not a lot of churn in these allocations (i.e. releasing a previous reservation).

### Relationship to the MDR

The MDR includes a review for ANA items. This review will check that all allocated values match the values in the database, that there are no <ANA> flags, and that no other administered objects are allocated without going through the ANA.

# Posting a draft

## Roles and responsibilities

The WG chair usually nominates a WG vice chair to be responsible for posting drafts.

The TG chair is responsible for ensuring that the draft to be posted meets certain minimal quality requirements (e.g., correct date and version number).

The TG editor is responsible for producing the clean (i.e. non-redline) and redline drafts.

## Review panel

Editing errors will happen when you incorporate approved changes.

Don’t expect your voters to want to correct editorial defects.

It is recommended that an editorial review panel is formed – within the TG - to review and allow you to correct any substantive changes prior to ballot.

## Format for posted drafts

Posted drafts:

* Must be in .pdf format
* Should have bookmarks for navigation
* Internal Cross references should be “hot” links that take you to the target
* During a recirculation ballot a redline .pdf must be provided

Other subsidiary documents (e.g., Word Conversion) can be provided at the discretion of the TG editor.

## Naming convention for drafts and redlines

### Labelling drafts

A draft must be labelled “IEEE <project>/<draft>, <month><year>” on the header line.

Example: IEEE P802.11ab/D3.0, July 2099.

The <project> designation is determined by the IEEE-SA.

### Document file-name for draft

Syntax:

Draft <project>\_<draft>.pdf

Examples:

* Draft P802.11n\_D8.0.pdf

A draft sent to ballot must have the number D<x>.0 – i.e., it must be a whole number.

A draft not intended to be sent to ballot must have a non-integer number: D<x>.<y>, where y != 0.

### Document file-name for redline

Syntax:

Draft <project>\_<draft> Redline [Compared to <project>\_<draft>].pdf

Examples:

* Draft P802.11n\_D8.0 Redline.pdf
* Draft P802.11n\_D7.04 Redline Compared to D7.03.pdf

The short form must be used for a redline that accompanies a balloted draft. The short form implies comparison with draft (<draft> - 1).

## Checklist for posted drafts

### Interim (i.e., not for balloting)

* Labeled and named correctly
  + Labeled according to style guide on the top of each page
  + Document name assigned according to 802.11 practice (see above)
* Updated copyright notice per style guide and current year in copyright notice
* “This is an unapproved IEEE Standards Draft, subject to change” on the bottom of all pages

### For balloting

* Must meet all requirements for posting as an interim draft, as well as:
* Substantially Complete
  + No Avoidable TBDs!
  + No Incomplete Text
  + No Outstanding Motions To Be Done
* No publication flaws of these types:
  + “reference not found” errors
  + “figure not found” errors
  + TOC updated
* Pages numbered and lines numbered
* Clean version and, if recirculation, a redline
* If sponsor ballot:
  + 11MC process complete
  + IEEE-SA MEC process complete

### Posting process

1. The TG editor:
   1. determines it’s time to post the draft
   2. produces clean and, if required, redline drafts
   3. sends it to the TG chair
2. The TG chair
   1. Performs the interim/balloting checklist (as appropriate) and determines the suitability of the draft for posting
   2. Indicates in an email to the WG vice chair that the draft is ready for posting
3. The WG vice chair
   1. posts the draft
   2. Asks the TG chair to check and announce the posted draft
4. The TG chair
   1. Checks that the draft can be opened from the website, that it is the correct document.
   2. Announces in an email, to one or more 802.11 reflectors, the availability of the draft.

### Access to Drafts

Drafts are accessible from the 802.11 website members’ area. These drafts are accessible to anybody who is an active member of 802.11 (see Operations Manual for definition of “active”). The drafts are removed once the Standards Board approves publication of a project.

Drafts accessible from the 802.11 website are also accessible to all attendees at an 802.11 session, using a local file-server.

As the editor of a draft, you may be asked to provide copies of the draft. Providing copies of the draft, except for the purpose of standards development is not allowed. If in doubt, route any such requests to the WG vice chair.

Do not make drafts accessible to the public – e.g., by placing on an unsecured web server. Using a secured web server to share documents within your editorial team is permitted. Please limit those with access to those contributing to your editorial process.

# Source Control

## Need for source control

Source control is essential for the orderly development of the draft.

The following are recommended requirements of any source control tool:

* The ability to reproduce exactly any posted draft from sources
* The ability to easily create checkpoint changes during the development of the draft so that, on corruption or mangling of the file, a minimal amount of work is lost.

This can be managed by appropriate use of directories and copy/paste. However, because there are tools out there that provide this and additional features like logging changes, it is recommended to use a source control tool.

An example of a free tool is Subversion. It also handles binary files well.

## Frame source layout

This subclause is specific to drafts in Frame format.

There are two structures to consider: source files and the contents of the book file.

The book file lists components that together make up the draft.

For example, the book file for 802.11REVmb D9 looks like this:

|  |  |
| --- | --- |
|  | Generally you will have a separate file for the front-matter, one each for the generated tables of contents, and one or more files for your body.  The case of REVmb is somewhat extreme, with some clauses being split into multiple files.  It is recommended, for practical reasons[[6]](#footnote-6), that each file contain no more than 100 pages of text. |

For the source directory, it is recommended to use a simple layout with all the sources plus the book file in one directory. All the figure binaries in a subdirectory, and all the figure sources in another subdirectory.

The following is an example file layout for my REVmb editing work:

The root directory

A directory containing one subdirectory per balloted draft, and one for recent posted drafts.

BTW, the icon overlays are part of the tortoise SVN tool, indicating which of the directories need committing.

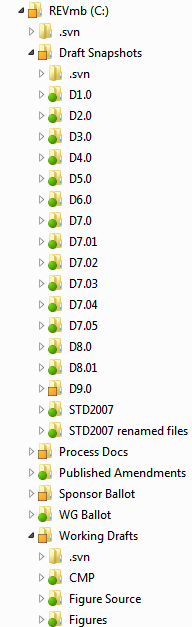
Where ballot data is kept

Where the working drafts are kept

The redline generation work area

Source of most figures

Binaries of all non-embedded figures



# Comment Resolution

## WG Comment Resolution Guide

Refer to document 11-11/1625 (r2 at the time of writing) for general comments on the comment resolution process and how to write comment resolutions. This document does not repeat guidance given in that document.

## Editor’s role in comment resolution

The management of comment resolution is not strictly an editorial task.  However it often falls to the editor to manage this process. A comment resolution committee (CRC) – the group responsible for writing comment resolutions, may also be created by TG.

Formally, it is the job of the TG chair to nominate whoever does this role.

## Treatment of editorial comments

### Recommended process

With one exception below, there is little distinction between the treatment of editorial and technical comments.

The commenter has two degrees of freedom with which he can label the comment:

* Technical / Editorial / General (at sponsor ballot only)
* Must be satisfied / need not be satisfied (i.e. part of a no vote / not part of a no vote)

It is recommended that editorial comments are handled as follows:

1. The editor reviews all comments and determines which are editorial
   1. Usually the commenter’s marking is accurate.
   2. However, the commenter may ask for changes that the editor believes have a technical impact. In this case the editor should deem the comment technical.
   3. The reverse is also true, a commenter may label a comment technical that is clearly and obviously editorial (i.e., cannot have any impact on the operation of any device).
      1. The editor may deem these comments editorial.
      2. However, the editor needs to make the TG aware of this, e.g. – by sending an email “I believe the following comments should be deemed editorial” to the appropriate reflector. This allows this marking to be checked and challenged.
2. The editor prepares comment resolutions for all comments deemed editorial.
3. The editor publishes these comment resolutions to allow review by the task group.
   1. The editor may also prepare a “speculative” draft showing the effect of the proposed changes. It is speculative in the sense that it reflects the opinion of the editor, and has not been approved by the TG.
   2. Generation of a speculative draft shortens the time taken to start the next ballot after an 802.11 session, because this is editing work that can be done off the critical path.
4. The editor presents whatever comment resolutions the TG wishes to discuss. Usually the group will be interested in only a small subset of the comments.
5. After discussion and any necessary changes, the comments are approved by motion in the TG.

Each group has its own needs, and may find an alternate route preferable.

### Which comments are editorial?

Comments sit in one of three buckets:

* Obviously editorial (e.g. add missing space), change font size, correct spelling
* Borderline (e.g. correct ambiguity in description, use of normative verbs)
* Obviously technical (e.g. add new frame format)

The editor should only deem editorial those comments that are obviously editorial.

If the editor is asked to perform resolution of any borderline comments, it is recommended that these resolutions be presented and labelled separately from editorial comments, and be considered one by one by the CRC. The point is that any change with a possible technical impact must be given individual attention by the CRC.

### Can the WG delegate resolution of editorial comments?

This is not something we do much of, because there is not much reason not to use the process described above. The overhead of approving editorial comments in the TG is small, given that the resolutions have been circulated in sufficient time for those interested to determine they have no issue with the resolution.

In the case that an editorial comment is not marked as “must be satisfied”, the following “blanket” comment resolution may also be applied:

*“This comment is deemed editorial and delegated to the task group editor for consideration in developing future drafts. Please note that the IEEE standards are edited professionally prior to publication”*

### Can we pass editorial comments to the publication editor?

It is possible also to pass strictly editorial comments received during sponsor ballot to the publications editor. This should be reserved for the “last ballot but one” during sponsor ballot, where comment resolutions are being written that do not require a change in the draft. During this process, if any comment resolution does require a change in the draft, the editorial resolutions should also be handled by the editor.

In this case the resolution should be a “revised”:

*“The WG will request that IEEE-SA Editorial Staff review this comment prior to publication and implement changes as required by their editorial guidelines”*

# The publication process

At the end of the project, the project technical editor works with IEEE-SA editorial staff to prepare the draft for publication.   This section describes that process.

1. The project technical editor supplies sources and graphics files to the IEEE-SA, matching the draft approved (or pending approval) by the standards board.
2. The first feedback from the SA editor is a marked up copyedit pdf.
   1. This document will highlight/show all proposed changes
   2. This document will highlight any issues where a response is requested
3. The project technical editor will review the copy edit and provide responses to all issues highlighted,  and any changes that we have a concern with.
   1. The project technical editor should maintain a spreadsheet showing location of change,  the response to it,  and an agreed outcome
   2. This spreadsheet will bounce to and fro with the SA editor until there is a consensus on all changes
   3. The SA editor may need to issue updates to the copyedit.  These will usually be distinguished visually (i.e. different comment colour) so that the project technical editor doesn't have to re-review old material.
4. Near the end of this iterative process, the SA editor will provide a clean draft that the project technical editor can check against the copyedit/spreadsheet
5. The WG technical editor will support and respond to any queries from the project technical editor during the process.
6. Towards the end, the WG technical editor will check with the project technical editor that he or she is satisfied with the resolution of any issues encountered.
7. The WG technical editor should do sanity checks (i.e., random sampling, and review the implementation of all contentious issues) on these, and how they’ve been handled in the draft
8. When the WG technical editor is satisfied, he or she passes on a recommendation, to approve publication, to the WG chair.
9. The WG chair has final WG approval of the draft, and passes this approval on to the IEEE-SA.

During this process, the SA editor will ensure consistency of style and numbering between an amendment and its baseline.  The editor will adjust for compliance to the IEEE-SA style guide, for internal consistency, for syntax, grammar and spelling.   This typically involves multiple corrections per page.   You may be surprised at how big an improvement the editor will make to the readability of the document.

The project technical editor should expect to spend 1 week of effort on this process per 100-200 pages of document.

# References

<http://standards.ieee.org/develop/>

[IEEE-SA Standards Style Manual](https://development.standards.ieee.org/myproject/Public/mytools/draft/styleman.pdf)

11-07-0827 - The ANA process

11-08-0644 – The Numbering spreadsheet

11-09-0002 – The [802.11 OM](https://mentor.ieee.org/802.11/documents?is_dcn=0002&is_year=2009)

11-09-0615 – The 11MC process

11-09-1034 - [WG11 style guide](https://mentor.ieee.org/802.11/documents?is_dcn=1034&is_year=2009)

11-11-1625 – The WG11 Comment Resolution Guide

1. Henry Ptasinski comments: “*I believe the reference is complete, but I have not worked with conditional text beyond a simple test of the instructions. The steps required to define the actual conditional text do not seem to lend themselves to being used for more than a handful of times. I definitely would not want to go through the process for thousands of comments.*” [↑](#footnote-ref-1)
2. Bash scripts are available if you are running Linux. If you are running Windows, you can install Cygwin and get the same result – which is how I do it. [↑](#footnote-ref-2)
3. We have no experience in the group of using this method, so cannot comment on its convenience. [↑](#footnote-ref-3)
4. This is what the IEEE-SA rules say. However, IEEE-SA has accepted Frame with embedded objects (including Visio) in the past. Whether the IEEE-SA editors can handle any particular flavour of embedded object is unknown. Whether the 802.11 editor can handle any particular flavour of embedded object when incorporating the amendment is somewhat easier to determine, but doesn’t really help in this case. [↑](#footnote-ref-4)
5. The IEEE-SA won’t be interested in these as they generally make no attempt to modify equations/drawing. [↑](#footnote-ref-5)
6. Redline generation works best with small files. The redline generation process appears to take polynomial time on the size of the file. Before Annex C was split up, the last comparison hadn’t completed even after an overnight run. After it was split into 4 files, the worst case redline time was 1 hour. [↑](#footnote-ref-6)