SGP/28.8

Starlink Project
Starlink General Paper 28.8

M D Lawden
A C Charles

25 September 1997

How to write good documents for Starlink

Abstract

This document gives advice on how to write good Starlink documents. It lists Starlink documents which users said they liked in the 1994 Starlink Software Survey, and identifies common characteristics of style and organisation which can be recommended to authors.

Various features of documentation which the authors believe to be harmful are specified, including common problems with pages produced for the World Wide Web. Many of these will inevitably reflect the taste of the authors, and these may not correspond to your own. However, they derive from our belief that it is right for an author to spend time and effort on a document in order to save the reader time and effort in reading it.

Contents

1 Introduction **1**

2 Starlink Software Survey **2**

2.1 Best documents 2

3 Planned improvements to Starlink documents **3**

3.1 Cookbooks 3

3.2 Hypertext 3

4 World Wide Web **4**

5 Good style and organisation **4**

6 Document diseases **5**

6.1 Web diseases 7

7 References **7**

A Standard elements of a Starlink document **9**

A.1 Code, date, filename 9

A.2 Standard headers, type size, page size, layout 9

A.3 Title 9

B Producing a document **10**

B.1 Starting or revising a document 10

B.2 Using LaTeX 10

B.3 Using Star2HTML 10

B.4 Submitting a document for distribution 10

1 Introduction

This note gives advice on how to prepare good documents for Starlink. It applies particularly to Starlink User Notes (SUN), Starlink Cookbooks (SC), and Starlink Guides (SG), as these are Starlink's main ways of telling its users how to use its software. Some extra advice is given for World Wide Web pages.

Any judgement about what constitues a good document is bound to be partly a question of taste, and our taste may differ from yours. Naturally, this document reflects our taste. However, a lot of our advice is based on our analysis of the common characteristics of documents which have been judged good by Starlink users.

We believe the following general principles apply to good documents:

- Basic literacy, including a respect for conventional spelling, punctuation, and grammar, is a fundamental requirement.
- The layout and organisation of a document should help the reader to understand its structure and find information.
- The text of a document should be physically easy to read.
- A user document should have more readers than writers. It is, therefore, cost-effective for a writer to spend his own time in order to save his readers' time. It is wrong for a writer to excuse a shoddy or difficult-to-understand document by claiming that it saved him time doing it that way.
- Avoid irritating your readers. (Not an easy task, as no doubt this document shows).

The Starlink Document Librarian (Mike Lawden) is responsible for ensuring that Starlink documents reach an acceptable standard. If Starlink published poor quality documents it could suffer political damage which, in turn, could damage our service to users. Because of this, the Librarian has the power to refuse to accept poor documents for distribution within Starlink because of the bad impression they would give of the Project. Alternatively, he may modify them in order to make them acceptable.

Three recent developments have influenced our views on what constitutes a good Starlink document:

Starlink Software Survey, 1994 – this provided valuable feedback from users about what sort of documents they wanted, and what they thought of current documents. (see SGP/43)

Starlink Software Strategy – this is a response by Starlink to the wishes expressed by its users in the Starlink Software Survey. Among other things, it contains plans for improving Starlink documents. (see SGP/42)

World Wide Web – this is an important technique for publishing information which makes it easier and faster to use, update, and distribute.

The impact of these developments on Starlink's document production is discussed in the next three sections. This is followed by a section containing advice on how to achieve good style and organisation in your documents, and a section describing practices which we think are bad and should be avoided. Various technical matters concerning Starlink documents and how to produce them are contained in two appendices.

2 Starlink Software Survey

A demand for better documentation was a marked feature of the Starlink Software Survey of 1994. This is shown by the answers given to questions 3 and 4.

> Question 3: *How could Starlink software be improved?*

Out of 18 possible answers, the 2nd and 3rd most popular (totalling 17% of the vote) were:

- Better on-line documentation.
- Better printed documentation.

> Question 4: *How could Starlink Documentation be improved?*

Out of 23 possible answers, the top four (getting 45% of the vote) were:

- Cookbooks for specific types of work.
- A more tutorial style.
- More practical examples.
- Better examples for programmers.

The message is loud and clear: users want more help with how to *use* the facilities provided by a piece of software, rather than just plain descriptions of the facilities provided.

2.1 Best documents

The survey also asked users to mark the quality of the documents provided with each software item they used. The ones which got the highest average marks (starting with the best) were:

- SUN/40: *CHR – Character handling routines.*
- SUN/39: *PRIMDAT – Processing of primitive numerical data.*
- SUN/100: *TPOINT – Telescope pointing analysis system.*
- SUN/33: *NDF – Accessing extensible n-D data.*

- SUN/105: *STARLSE – Starlink extensions to language sensitive editor.*
- SUN/110: *SST – A simple software tools package.*
- SUN/31: *REF – Handling references to HDS objects.*
- SUN/87: *JPL – Solar system ephemeris.*
- SUN/90: *SNX – Starlink extensions to NCAR graphics utilities.*

These are the documents the users said they liked best, so take a look at them and consider modelling your documents on their organisation and style. In section 5 we list what we think are common characteristics of these documents. These characteristics can be used as guides for your own writing.

3 Planned improvements to Starlink documents

Major improvements to Starlink's documents are underway (see SGP/42). Two key ones are:

- Cookbooks.
- Hypertext.

3.1 Cookbooks

These provide recipes for common types of data reduction. Users will, therefore, not need to understand a package in depth before being able to do useful work with it. They are issued as *Starlink Cookbooks* (SC).

3.2 Hypertext

This publishes Starlink information on the World Wide Web in the form of HTML files, and enables Starlink documents to be linked together as an integrated whole, using a hypertext linker called HTX (see SUN/188). Starlink documents are written mostly in LaTeX. New documents are produced in both hypertext and paper form. Old documents which only exist in paper form can be converted to HTML form by translators such as `Star2HTML` (see SUN/199).

The hypertext infrastructure software and organisation required is now operational (see SUN/188). The work of transforming the current document set to hypertext form is proceeding rapidly.

A beginner's guide to HTML has been distributed as MUD/149. An updated version of this is available on the web at:

```
http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html
```

4 World Wide Web

As with paper documents, an effective way to write good pages for the web is to model them on existing good pages. You can easily examine the source text of pages that interest you, using standard facilities of browsers such as Netscape and Mosaic. Have a look at the Starlink pages. There are lots of them, and a good place to start is the central Starlink page at:

```
http://www.starlink.ac.uk/
```

From here you can go to most Starlink sites from the *Sites* link. There are also lots of Style Guides available to help you write good web pages. An example is:

```
http://www.w3.org/hypertext/WWW/Provider/Style/Overview.html
```

5 Good style and organisation

In Section 2.1 we listed the Starlink documents which the users voted the best available. Do these have any common features which can be used as guides for ones own writing? We took a close look at them, and all or most of them have the following characteristics:

- Well organised and formatted, so the structure of the document is clear and it is easy to find things.
- An abstract or description on the front page, summarising what the document is about.
- A contents list (at the front).
- An introduction and overview.
- A summary of the types of functions provided, and a classified list of commands or routines (by function; not just alphabetical).
- Clear instructions on how to start up the package or use the routines.
- Plenty of examples.
- Detailed descriptions of commands or routines, placed in a separate section or appendix.
- Descriptions of new features or changes introduced by the latest version of the software, and possibly a description of limitations or restrictions.

Any non-trivial document describing a software package or routine library should have this sort of sectioning. Some (such as SUN/1 which surveys the whole of Starlink software) can also benefit from having an index (much easier to produce with the new version of LaTeX).

We also recommend the following further characteristics for your Starlink user documents:

- Organise your paper into sections using sectioning commands (see SC/9, Section2).
- Use the *list* facilities to highlight options or features (see SC/9, Section 4).
- Indent example commands and show them as they appear to the user, including any prompt symbol. Leave a space between the prompt symbol and the command – 'what you read is what you see'. Thus:

  ```
  ICL> CAR_HELP SEARCH
  ```

  is clearer than

  ```
  ICL>CAR_HELP SEARCH
  ```

- Try to keep your formatting techniques as simple and obvious as possible. We find the basic techniques shown in SC/9 to be adequate for all normal purposes.
- Documents are often revised, merged, or restructured many times, so you should make it as easy as possible for this to be done on a computer. In your source document we recommend that you start each sentence on a new line; this makes future identification, insertion, selection, and movement of text very easy.

Starlink is trying to achieve a common style in all its documents, particularly in their format, so as to present a professional and uniform image. To achieve this, we request that you include in your documents the standard elements specified in appendix A.

6 Document diseases

The prime purpose of technical documents is to convey correct information efficiently and clearly. However, documents written by non-specialist authors often display features which, we believe, decrease this efficiency. Also, most authors type their own text, so various filters that cleaned raw text in the past (like fussy typists and obsessive editors) have been bypassed. The result has been an epidemic of document diseases which makes life harder for the reader. Here are some that we come across:

Verbosity: The most common fault in writing. It is an insidious disease which afflicts us all, but it can be controlled if you look for it. There is far too much to read without having to wade through redundant waffle. Good documents should be *short* and *concise*. Get to the point! A reader does not want to waste his time ploughing through:

> *The purpose of this document is to introduce the reader to the CHART software system which has been developed to facilitate the production of finding charts which, hopefully, will be of some assistance to the user's astronomical investigations.*

when he would be quite satisfied with:

> *This program produces star finding charts.*

Members only: Authors sometimes get so wrapped up in their work that they cannot conceive of anyone not being *au fait* with what they are writing about. Don't just write for members of your clique, but give outsiders some background and context to your work. Also, a brief introduction is very useful for me when I am adding notes on new software to SUN/1.

Where's the ON switch?: Believe it or not, people sometimes omit to describe how to start up their software, or they may bury this information in the middle of a big paragraph. Put this information near the front and make it clear, e.g.:

> *To start the program, type the command:*
>
> ```
> $ CHART
> ```
>
> *You can then get an introductory summary of the principal commands by typing:*
>
> ```
> $ PROMPT
> ```

Smothered punctuation marks: Most people accept the convention of leaving a space between individual words as they recognise that it helps the reader. However, the convention of leaving a space between sentences and after punctuation marks is under attack. One now sees text written like this:

> *I can save my precious time,and energy,by missing out unnecessary spaces;anyway,I'm impatient.I don't care if this makes my text irritating to read,and infuriating to edit,as my time is more important than yours,and this extra speed gives me a competitive edge.*

This style of punctuation might be tolerable in a Fortran FORMAT statement (where, perhaps, it originated), but it is barbaric and illiterate in text meant for the human eye.

Lonely parentheses: This is the inverse of the *Smothered punctuation marks* disease – too much space rather than too little. It has the following symptom:

> *Some authors (not many) do this.*

No doubt, such authors think this is clearer than:

> *Most (thank goodness) do this.*

The problem with this disease is that it makes the syntactic structure of a sentence less obvious – you are writing for the human eye, not a parser. Moreover, unless you go to special trouble, odd parentheses will stray onto the previous or next line.

Tabs: The trouble with tabs is that they may not be interpreted as you expect, since this depends on the device or software your reader is using. What you see on your terminal may not be what he sees on his. The safest thing is not to use them.

Missing files: A common disease of TEX virtuosi is to submit document files which refer to other files which only exist in their author's local environment. Make sure you submit all the files required for someone else to produce hardcopy.

Clever dates: Many text processors have a facility to generate today's date automatically. This can be very convenient in things like letter headings which are used to produce a single document at a specific moment, but it is dangerous in long-life source documents that may generate hardcopy several times. A date should identify such a document uniquely and should not be vulnerable to arbitrary change. The use of clever dates can cause a number of versions of a document to go into circulation which are identical, apart from the date in the heading. This can cause great confusion.

6.1 Web diseases

The World Wide Web is a popular tool for publishing information, and Starlink uses it extensively. The enormous amount of material on the web and the arrival of commercial pages has led to a competitive striving for attention by its information providers. This has not always been to the advantage of the reader, as there is sometimes a conflict between attracting a reader to your text, and making that text easy to read. Three forms of attention-seeking have spread across the web like cane toads in Queensland. We believe they compromise legibility:

Background images: Text is often presented against a background image. The intention is to make the page look attractive and encourage people to read it. This technique may work when used with discretion in high-quality publications printed on glossy paper, but we feel it doesn't work successfully on computer terminals. The reason is that most computer displays have much lower resolution and dynamic range than glossy paper, so a background image seriously degrades the legibility of the foreground text. Another problem is that some browsing software just can't handle background images properly. If you use background images your text may become illegible (and your writing efforts futile).

Fancy colours and fonts: You don't have to use background images to make your reader abandon an attempt to read what you have written. This can also be achieved by extrovert choices of colour for text and background, and also by using fancy fonts.

Flashing lights: You can draw attention to the latest or most important information on your web page by repeatedly flashing it on and off. The effect on at least some readers is similar to that produced by those car alarms that go off in the car park next to your office window. The problem is the same: you can't turn the message off once you've got the point. It then becomes distracting and merely adds to the ambient noise.

No doubt when we all have multi-media terminals, information will be accompanied by exciting background music and stimulating drum beats. We are not looking forward to it.

7 References

(1) MUD/149: *HTML – A beginner's guide.*

(2) SC/9: *LaTeX Cook-book.*

(3) SGP/42: *Starlink software strategy.*

(4) SGP/43: *Starlink software survey results, 1994.*

(5) SUN/9: *LaTeX – A document preparation system.*

(6) SUN/188: *HTX – Hypertext cross-reference utilities.*

(7) SUN/199: *Star2HTML – Converting Starlink documents to hypertext.*

(8) SUN/201: *Latex2HTML – LaTeX to HTML conversion.*

(9) Leslie Lamport: *A Document Preparation System. LaTeX User's Guide & Reference Manual*, 2nd ed, Addison-Wesley, 1994, ISBN 0-201-52983-1

A Standard elements of a Starlink document

A.1 Code, date, filename

Every Starlink document has an identifier and a date of issue. Identifiers have a format like `SGP/28.7` where `SGP` is the class code, `28` is the number within that class, and `7` is the version number starting at 1. Identifiers are allocated by the Document Librarian (`mdl@star.rl.ac.uk`). When a note is revised, its version number should be incremented and the date of issue updated.

Files which store Starlink documents should have a name like `sgp28.tex`, where `sgp` shows the document class, `28` tells you the number, and `tex` tells you it is a TEX or LATEX source file. Files containing Starlink classified documents are stored in `/star/docs`.

A.2 Standard headers, type size, page size, layout

Use the LATEX template files like `sun.tex` which are stored in `/star/docs`. These have properly formatted headers and correct type size, page size, and layout. They also have the definitions required for you to produce hypertext, as described in SUN/199. Avoid using a type size smaller than 11 point.

A.3 Title

A user looking for information usually selects a note on the basis of its title, and this should, therefore, be concise and informative. It should contain the acronym used to refer to the software, and indicate its function. Remember, your note may have a long life, so phrases such as “A New…", which will quickly become either obsolete or positively misleading, should be avoided. An example title is:

ASPIC – A set of image processing programs

Don’t assume a reader already knows what your software does. For example, don’t have a title like:

MYPROG – An introduction

Say what MYPROG does:

MYPROG – An HTML editor: Introduction

B Producing a document

B.1 Starting or revising a document

When *starting a new document*, copy the appropriate template file from `/star/docs` into your own directory and change the variable information; then add your own text. This procedure is explained in some detail in SUN/199. Produce your original file in the form of a `.tex` file ready for processing by LaTeX and Star2HTML.

When *updating a document*, copy the existing file from `/star/docs` and edit that; do *not* start from a private copy as your submitted file may have been edited by a Librarian before final issue, and the changes (such as spelling corrections) will be lost and have to be done again.

B.2 Using LaTeX

For all but the simplest of notes you will probably wish to use a text processing program. Starlink recommends LaTeX. Don't use TeX unless you enjoy making life difficult for yourself and others; LaTeX is much easier. In our experience, the output from TeX source is inferior to that of standard Starlink LaTeX documents, and invariably has idiosyncratic style and format.

SC/9 contains many examples of how to use LaTeX and shows the style preferred by Starlink. SUN/9 describes how to use LaTeX. There are lots of books about LaTeX to help you use it properly. The basic reference is the book by Leslie Lamport listed in the References.

B.3 Using Star2HTML

If possible, you should add hypertext links to your document and use Star2HTML to produce the hypertext version. This procedure is explained in SUN/199. The basic procedure for checking the hypertext version of your document from within your working directory is as follows:

```
% star2html sgp28
% hlink
% showme ./sgp28
```

This will display your hypertext document (in this case, sgp28.tex) in a window running a web browser.

B.4 Submitting a document for distribution

When you have finished your document, tell the Software Librarian (`starlink@jiscmail.ac.uk`) where your file is. If necessary, send paper copies of any additional material, such as diagrams, which is required in the final document, but which doesn't appear in hardcopy derived from your file. The Librarians may edit your file to make it conform to Starlink standards, and will then produce a paper master and get copies made. They will also distribute the file in a software release, and send several copies to every Starlink Site Manager for users.