USING THE SDTTY "SESSION" FEATURE

Sdtty has extensive mechanisms for managing your sequences and the files that contain them. Depending on your individual tastes, there are several ways that you might choose to manage things. We will explain a number of different ways of doing things. Hopefully, one of them will be suitable for you.

THE OLD-FASHIONED WAY

The simplest way to use sdtty is without any session file at all. Just start the program with the command *sdtty* and the level at which you want to call, for example:

```
sdtty c1
```

Every sequence that you write will be written to a "default" file name, which is '`sequence.C1`', or '`sequence.A2`', or whatever the level is. When you write a sequence, sdtty checks to see whether that file exists. If it does not exist, sdtty creates it and then writes the sequence to it. If the file already exists, the new sequence is appended to the end. This way, a session of writing sequences will result in a file containing all of the sequences. If you print that file, the sequences will all be printed, one per page.

This is very simple, but it is dangerous. When you write another batch of sequences the next day, sdtty continues to append the sequences to that file. When you print the file, you will get all the old sequences as well as the new ones. Sdtty never deletes a file, or removes a sequence from an existing file, so the file '`sequence.C1`' will just keep getting bigger.

A very simple way to deal with this problem is to delete the file immediately after you print it. You might print it by saying

```
copy sequence.c1 lpt:
```

You could then delete it by saying

```
del sequence.c1
```

Be very careful with this. Don't delete the file until you are sure you have successfully printed it. Once you delete it, it is gone, and only the printed-on-paper copies of the sequences survive. You might want to copy the file first, before deleting it:

```
copy sequence.c1 lpt:
copy sequence.c1 21may95.c1
del sequence.c1
```

You must choose a new name (in the example above, it was '`21may95.C1`') each time. Then all those files will be an archive of everything that you have written.

You may find it more to your liking to change the name of the file, from '`sequence.C1`' to something else. The "change output file" command in sdtty lets you do this. As soon as you start sdtty, you can issue that command:

```
Version Sd 31.62 : db31.62 : ui1.7tty     C1
Output file is "sequence.C1"                <-- it tells us the default name
Enter startup command> change output file   <-- we desire to change it
Sequence output file is "sequence.C1".      <-- it tells us the old name
Enter new file name: 21may95.C1             <-- we give the new one
Output file changed to "21may95.C1"         <-- it tells us the new name
Enter startup command> sides start          <-- we're off and running
```

You can give the "change output file" command at any time, and it will take effect the next time a sequence is written.

As we will see later, we can make sdtty choose a file name for us, by giving the name '`+`' or '`*`'.

At the end of a sequence, sdtty always tells you what file it is writing to, and whether it is creating a new file or appending to an old one.

```
(waves)--> end this sequence               <-- we want to write the sequence
Type comment for this sequence, if desired.
Enter comment:                             <-- it asks for a comment;
                                                       we press ⟨ENTER⟩
File does not exist, creating it.          <-- it tells us what it is doing

Sat Mar  9 19:22:26 1996     Sd31.62:db31.62     C1

SIDES wheel thru
swing thru
right and left grand  (3/8 promenade)

Sequence written to "21may95.C1".          <-- and what file it wrote to
```

WARNING!!!!!! Sdtty can't always successfully append new sequences to a file that has been processed by a word processor. If you write a number of sequences, read the file into a word processor (e.g., Word Perfect), edit them to make them look nicer, and then print them, you must not let sdtty write to that file again. Delete the file (saving it first if you wish) before running sdtty on that file again. If you use the "change output file" command to make sdtty write to a different file each time, you won't have this problem. But if you write to '`sequence.C1`' each time, the following could happen:

You write 10 sequences, creating the file '`sequence.C1`'. You read that into Word Perfact and edit it, perhaps changing fonts, spacing, indentation, and so on. Then you leave Word Perfect and use sdtty to write 10 more sequences. You start Word Perfect again, intending to edit sequences 11 through 20 of the file. Sequences 11 through 20 will be lost. The problem is that word processors can convert files from sdtty into their own internal format, but they can't convert files that are partially in sdtty format and partially in their own internal format.

After editing sequences 1 through 10 with Word Perfect, you must print that file (preferably with the print command in Word Perfect.) After you print it, delete it before running sdtty again. Sdtty will re-create a fresh file, which Word Perfect will be able to read.

PRINTING SEQUENCES IMMEDIATELY

If you wish, you can tell sdtty to print each sequence directly, without storing it in a file. This may be a somewhat dangerous operation, in that the sequences are never saved in files.

To do this, use the "change output file" command to set the file name to '`lpt:`' or whatever the name of your printer is. For example, if you normally would print a file by typing

```
copy sequence.c1 lpt:
```

your printer is called 'lpt:'. you can make sdtty send the file directly to the printer.

```
Enter startup command> change output file
Sequence output file is "sequence.C1".
Enter new file name: lpt:                   <-- we give the name lpt:
Output file changed to "lpt:"
Enter startup command>
```

Doing it this way makes the management problem extremely simple – perhaps too simple. If your printer jams, you lose the sequence.

USING THE SESSION FILE

For more sophisticated file management, you can use the "session file". This is the file 'SD.INI'. Normally, no such file exists when sdtty is shipped, so the session feature is turned off. You can turn it on by creating that file. The simplest way is by copying the file 'SAMPLE1.INI', which is shipped with the program.

Just do this:

```
copy sample1.ini sd.ini
```

The file 'sd.ini' is now ready to go. Sdtty will now use it to enable the session feature.

What if, after reading the rest of this memo and trying out the session feature, you decide that you don't want it? Simple. Just delete the file:

```
del sd.ini
```

and sdtty will go back to the old way of doing things.

Assuming that you have copied 'sample1.ini' to 'sd.ini', you can try the session feature. What will be set up is just a demonstration. After you learn how to use it, you can change the configuration to suit your own style of usage.

Start sdtty. You don't need to specify a level. That is, just type:

sdtty

You should see this:

```
Do you want to use one of the following sessions?

  0     (no session)
  1  sequence.C3        C3              12      NACC, June 1995
  2  sequence.C4        C4              31      NACC, June 1995
  3  workshop           C1              75      My wednesday group
  4  a2stuff            A2               9      NESRDC/1995
  5  a1stuff            A1               2      Lake Shore Farm Weekend
  6     (create a new session)

Enter the number of the desired session:
```

There are five sessions shown in the demonstration, numbered 1 through 5. Type the number of the desired session, and press ⟨ENTER⟩.

Incidentally, any time you want to use sdtty without the session feature, you can type *0* and press ⟨ENTER⟩ at this point, or just press ⟨ENTER⟩. Sdtty will act as though the session file had not existed, and will operate the old-fashioned way. The session file will still be left intact, so it can be used the next time. Of course, if you decline the session feature, you need to give the level.

The sample session file presumes that there are 5 groups, or dances, or festivals, or workshops, or whatever, that we are writing material for. One of them is our Wednesday C1 workshop. Choose that one by typing 3 and pressing ⟨ENTER⟩.

```
Enter the number of the desired session:  3
Sd: reading database....................................................done

Version Sd 31.62 : db31.62 : ui1.7tty     C1
Output file is "workshop"                   <-- it tells us the output file

Enter startup command> heads start          <-- we're off and running
```

When we specified session 3, sdtty got 4 pieces of information out of the session file:

- The file name. In this case, it is '`workshop`', instead of the '`sequence.C1`' that would normally be used.
- The level. C1 in this case.
- The starting sequence number. Every card will be serialized with a number.

- The title. In this case, it is "My wednesday group". This title will be written at the top of every card.

So we are now running with the Wednesday C1 workshop session, about to write sequence number 75. Let's write a short opening "biggie" sequence, like heads wheel fan thru, pass the ocean, relay the shadow, extend, right and left grand. When we type "end this sequence", sdtty prompts us for a comment, in the usual way. This is an optional comment that is specific to the sequence. It will be printed on the card after the title. It is intended for things like "hard", or "awkward for side women", or whatever might be useful to know when calling the card. Let's type the comment "opening biggie":

```
(waves)--> end this sequence
Type comment for this sequence, if desired.
Enter comment: opening biggie
File does not exist, creating it.

Sat Mar  9 20:06:54 1996     Sd31.62:db31.62     C1
             My wednesday group   #75       opening biggie

HEADS wheel fan thru
pass the ocean
relay the shadow
extend
right and left grand  (3/4 promenade)

Sequence #75 written to "workshop".
Enter startup command>
```

Sdtty displays the entire sequence in the usual way. Note that both the title "My wednesday group" and the individual comment "opening biggie" appear on the card, along with the sequence number. Sdtty also tells us the usual things, such as the fact that the file did not exist and hence was created, and that the file name was '`workshop`'.

When we write the next sequence, its number will be 76, and it will be appended to the '`workshop`' file.

After writing 10 sequences, they will be numbered 75 through 84. We can then exit from sdtty, print the file '`workshop`', and then delete it. Or we could just leave it around. The next time we start sdtty with session 3, it will continue appending sequences to the same file, and will start numbering with 85. It always keeps track of the sequence number for each session.

CHANGING THE OUTPUT FILE NAME

The "change output file" command still works when the session feature is used, with one difference – the new output file name will be remembered permanently. The next time sdtty is started, it will continue to use the new file name. If your file management style involves changing names daily, you could give the "change output file" command during the first session each day. (But we will presently discuss a way of making sdtty do that for you automatically.)

CHANGING THE TITLE

The "change title" command may be used to change the title for a session:

```
(facing lines)--> change title
Current title is "My wednesday group".
Enter new title: My thursday group
Header comment changed to "My thursday group"
```

The new title will be remembered permanently. Sdtty remembers the output file name and the title from one session to the next by rewriting them into the file '`sd.ini`'. It keeps track of sequence numbers the same way.

You can't change the level of an existing session – you must create a new session. You also can't change the sequence number. You must create a new session, which will start with sequence number 1.

CREATING A NEW SESSION

Now that you know how to use the session feature, you can create your own. Suppose that you are writing material for a Monday C2 group. Start sdtty with the level c2:

```
sdtty c2
Do you want to use one of the following sessions?

  0    (no session)
  1  sequence.C3        C3              12      NACC, June 1995
  2  sequence.C4        C4              31      NACC, June 1995
  3  workshop           C1              75      My wednesday group
  4  a2stuff            A2               9      NESRDC/1995
  5  a1stuff            A1               2      Lake Shore Farm Weekend
  6    (create a new session)

Enter the number of the desired session:
```

Type the number that is indicated for creating a new session. In this case, 6:

```
Enter the number of the desired session:  6
Sd: reading database....................................................done

Version Sd 31.62 : db31.62 : ui1.7tty     C2
Output file is "sequence.C2"
Enter new title:
```

It will create the new session, which will henceforth be session number 6. The level will be C2, and the sequence numbers will start at 1. The output file will initially be the default '`sequence.C2`', though you can change that. Sdtty doesn't yet know what title to use, so it prompts you for it:

```
Enter new title: Monday group
Enter startup command>                      <-- we're off and running
```

If we want to use a special filename, like '`monday.C2`', we can use the "change output file" command at this point:

```
Enter startup command> change output file
Sequence output file is "sequence.C2".
Enter new file name: monday.C2
Output file changed to "monday.C2"
```

We then write a sequence, ending with "end this sequence" in the usual way:

```
(left waves)--> end this sequence
Type comment for this sequence, if desired.
Enter comment: biggie
File does not exist, creating it.

Sat Mar  9 20:33:37 1996     Sd31.62:db31.62     C2
             Monday group   #1       biggie

SIDES star thru
double pass thru
stack the line
mix
left allemande  (at home)

Sequence #1 written to "monday.C2".
Enter startup command>
```

The session is now established, with its title, level, output file, and numbering. If we exit sdtty at this point and restart it, we will see:

```
Do you want to use one of the following sessions?

  0     (no session)
  1  sequence.C3         C3              12      NACC, June 1995
  2  sequence.C4         C4              31      NACC, June 1995
  3  workshop            C1              75      My wednesday group
  4  a2stuff             A2               9      NESRDC/1995
  5  a1stuff             A1               2      Lake Shore Farm Weekend
  6  monday.C2           C2               2      Monday group
  7     (create a new session)

Enter the number of the desired session:
```

By typing *6*, we will continue with this session. It will start with sequence number 2.

DELETING A SESSION

There are two reasons we might want to delete a session:

- We don't need it any more.
- It was part of the demonstration file, and we want to use our own sessions.

For example, we have now learned how to create and use sessions. The demonstration file that came with '`sample1.ini`' is no longer needed. We want to delete sessions 1 through 5, and just use the one we have created for our Monday C2 group.

Sdtty will delete a line from the session file '`sd.ini`' if you type the negative of that session's number. We want to get rid of session 5, the "Lake Shore Farm Weekend" session that came as part of the demonstration. Type -5:

```
sdtty
Do you want to use one of the following sessions?

  0     (no session)
  1  sequence.C3         C3              12      NACC, June 1995
  2  sequence.C4         C4              31      NACC, June 1995
  3  workshop            C1              75      My wednesday group
  4  a2stuff             A2               9      NESRDC/1995
  5  a1stuff             A1               2      Lake Shore Farm Weekend
  6  monday.C2           C2               2      Monday group
  7     (create a new session)

Enter the number of the desired session:  -5
```

Sdtty will immediately exit. It will not let you write any sequences. Do not be alarmed. Just start it again to delete the next one. You will see that the old session 5 is gone, and our new C2 session has been renumbered. Delete session 4:

```
sdtty
Do you want to use one of the following sessions?

  0     (no session)
  1  sequence.C3         C3              12      NACC, June 1995
  2  sequence.C4         C4              31      NACC, June 1995
  3  workshop            C1              75      My wednesday group
  4  a2stuff             A2               9      NESRDC/1995
  5  monday.C2           C2               2      Monday group
  6     (create a new session)

Enter the number of the desired session:  -4
```

The next time it is started, the old session 4 will be gone, and it will look like this:

```
sdtty
Do you want to use one of the following sessions?

  0     (no session)
  1  sequence.C3         C3              12      NACC, June 1995
  2  sequence.C4         C4              31      NACC, June 1995
  3  workshop            C1              75      My wednesday group
  4  monday.C2           C2               2      Monday group
  5     (create a new session)

Enter the number of the desired session:
```

You can keep deleting the demonstration sessions until you have only the ones you want.

USING CREATED FILE NAMES

You can let sdtty choose the file name for you, that will be unique for each day. This might be suitable for your style of file management. For example, you might like to print all sequences at the end of the day. Each day's sequences will go into their own file. You don't need to delete or copy any files, and sequences should never get lost.

To do this, use the special file name '`+`'. That's right, just a "plus" sign. You can put this in your session file, or you can use it without the session feature, by giving the "change output file" command.

As an example, let's create a session for a C3 weekend:

```
sdtty c3
Do you want to use one of the following sessions?

  0     (no session)
  1  monday.C2           C2              2      Monday group
  2     (create a new session)

Enter the number of the desired session:  2 <-- new session
Sd: reading database....................................................done

Version Sd 31.62 : db31.62 : ui1.7tty     C3
Output file is "sequence.C3"
Enter new title: Fall Festival              <-- new title
Enter startup command> change output file
Sequence output file is "sequence.C3".
Enter new file name: +                      <-- new file name
Output file changed to "9mar96.C3"          <-- it tells us the real file name
Enter startup command> heads start          <-- we're off and running
```

All our sequences will be written to the file '`9mar96.C3`' this time around. If we exit the program and restart it, we will see:

```
sdtty
Do you want to use one of the following sessions?

  0     (no session)
  1  monday.C2           C2              2      Monday group
  2  +                   C3              2      Fall Festival
  3     (create a new session)

Enter the number of the desired session:  2
Sd: reading database....................................................done

Version Sd 31.62 : db31.62 : ui1.7tty     C3
Output file is "9mar96.C3"
```

Note that the session file shows a file name of '`+`', so that it will always generate a file name based on today's date. When we run sdtty the next day, the file shown in the session file will still be '`+`', and the actual file will be '`10mar96.C3`'. We never need to delete files. Everything will remain on the computer, listed by the day that it was written. We just need to remember to print out each day's file. Listing the directory (or using the Windows file manager) may be useful for keeping track of things.

USING UNIQUE CREATED FILE NAMES

Creating a new file name each day will help you keep track of your files, and should greatly reduce the risk of lost sequences. However, if you use a word processor, there is still a vulnerability. If sdtty is used after Word Perfect is used on the same day, it will append its sequences to the same file, and, as noted previously, those sequences will be lost.

You can make sure that this will never happen by using the file name '`*`' instead of '`+`'. That is, just an asterisk. The behavior is the same as '`+`' except that SDTTY ALWAYS CREATES A NEW FILE, EVERY TIME IT IS STARTED.

For example, when sdtty is started for the second time on March 9, it will create the file '`9mar96a.C3`'. Note the '`a`' after the year. When started, it will see that the file '`9mar96.C3`' already exists. Since it doesn't know whether that file had been processed by a word processor, it can't trust it, so it creates a new one by adding the letter '`a`' after the year. Next time around it will add '`b`', and so on.

This style of use can lead to a large number of files to print, particularly if your style involves multiple writing sessions per day, but it will avoid having files corrupted if sdtty attempts to append to a file that a word processor has processed.