Roll Your Own… (7th Inning)

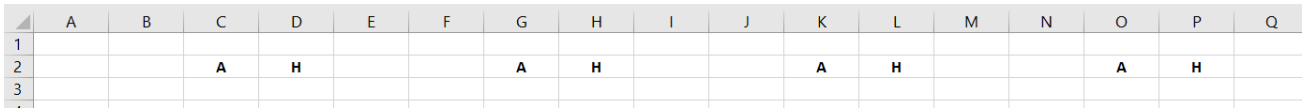
Top of the 7th – So, you wanna' make you own schedules, eh? Here is a method I developed and use to create my own series files using MS Excel and a text editor. You can create a FULL 2 league, 3 division, 5 teams per division schedule (30 teams total) or MORE with this method. In the example below, I will be creating a 4 team schedule (for demo purposes) that will play each team an equal number of HOME games and AWAY games with all other teams.

NOTE: Prior to doing this, you should work out how many teams you want in your league, and approximately how many games you want them to play, how many HOME games vs. each team, and how many AWAY games vs. each team. If you have more than 1 division, and also desire inter-league play, then a whole new set of math skills is needed. **REMEMBER:** Schedule playing within your division the most, then within your league, and then inter-league, all the while trying to even the amount of HOME and AWAY games against all teams! Yes, it's a tall order.

For Example: I want to play a 10 team league in a ≈160 game schedule. No divisions. No inter-league play. 160/(10-1)=17.77. Because the quotient is not a whole number (17.77), we're going to have to schedule more games against one or more teams, or lengthen or shorten our desired number of games. An easy way to do this is to use some "tricky" (brute force) math: 10-1 (-1 because I won't be playing against myself) teams x 18 games = 162. That's pretty close 😊. Figuring an even number of HOME and AWAY games against all the teams, I can see that I'll need 9 HOME games (18/2) and 9 AWAY games (18/2) to satisfy the 18 games required for each team. Now I know what I need, so let's begin to create the schedule:

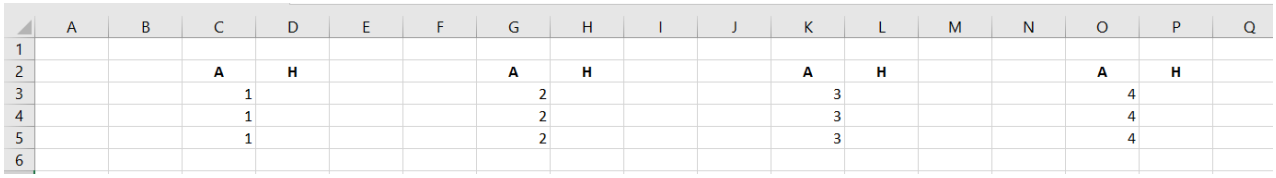
DON'T LET THIS LIST OF STEPS INTIMIDATE YOU! IT'S NOT DIFFICULT AT ALL, AND YOU'LL BE AN EXPERT AFTER GOING THRU THESE INSTRUCTIONS ONCE!

1) Open a blank MS Excel spreadsheet – place your cursor in cell C2. Add header markers to your sheet as shown… A for AWAY, and H for HOME.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1																	
2			**A**	**H**			**A**	**H**			**A**	**H**			**A**	**H**	
3																	

2) Create a column for each team (Team #1, Team #2, etc.) – Leave 3 blank columns in-between each team. You'll soon see why.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1																	
2			**A**	**H**			**A**	**H**			**A**	**H**			**A**	**H**	
3			1				2				3				4		
4			1				2				3				4		
5			1				2				3				4		
6																	

3) Place an opposing team number in the column next to each column that has a team number in it. DO NOT repeat the number of the column you're placing next to. You now have a basic schedule, where the AWAY team is the first team and the HOME team is the second team in the column next to it. If you look closely, each team plays an equal number of games HOME and AWAY, and play all other teams equally!

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1																	
2			**A**	**H**			**A**	**H**			**A**	**H**			**A**	**H**	
3			1	2			2	1			3	1			4	1	
4			1	3			2	3			3	2			4	2	
5			1	4			2	4			3	4			4	3	
6																	
7																	

4) Place your cursor on cell E3 and type in the following as shown:
=CONCAT("/v:",C3,".dat /h:",D3,".dat")
This creates text in column E3 that'll look like this:
/v:1.dat /h:2.dat
Look familiar? It is a line of the series schedule! Now copy and paste this line from E3 to all the remaining cells next to the 2 columns of numbers. It should end up looking like this:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1																		
2			**A**	**H**			**A**	**H**			**A**	**H**			**A**	**H**		
3			1	2	/v:1.dat /h:2.dat		2	1	/v:2.dat /h:1.dat		3	1	/v:3.dat /h:1.dat		4	1	/v:4.dat /h:1.dat	
4			1	3	/v:1.dat /h:3.dat		2	3	/v:2.dat /h:3.dat		3	2	/v:3.dat /h:2.dat		4	2	/v:4.dat /h:2.dat	
5			1	4	/v:1.dat /h:4.dat		2	4	/v:2.dat /h:4.dat		3	4	/v:3.dat /h:4.dat		4	3	/v:4.dat /h:3.dat	
6																		
7																		

5) Copy all cells from C3 to Q5 (C3:Q5)
6) Place your cursor in cell C6 and paste the copied cells (as values)
7) Copy all cells from C3 to Q8 (C3:Q8)
8) Place your cursor in cell C9 and paste the copied cells (as values). You should have 12 rows of schedule data for each team.

	A	B	C	D	E	F
1						
2			**A**	**H**		
3			1	2	/v:1.dat /h:2.dat	
4			1	3	/v:1.dat /h:3.dat	
5			1	4	/v:1.dat /h:4.dat	
6					/v:2.dat /h:1.dat	
7					/v:2.dat /h:3.dat	
8					/v:2.dat /h:4.dat	
9					/v:3.dat /h:1.dat	
10					/v:3.dat /h:2.dat	
11					/v:3.dat /h:4.dat	
12					/v:4.dat /h:1.dat	
13					/v:4.dat /h:2.dat	
14					/v:4.dat /h:3.dat	
15						

9) Copy and Paste all the cells containing data from E3:E14 to E15, and continue to Paste at cell E27 and E39. The last cell pasted should be at E50.
10) Congratulations! You have created a 4 round schedule (E column) for a league of 4 teams who will play each other an equal number of times (8) with 4 HOME games and 4 AWAY games against each other. This is a 24 game schedule for each team!

EXPANDING THIS CONCEPT

11) Place your cursor in cell E15. Edit the value in the cell by adding this to the end of the value (without the quotes): " /n:3" There is a space in front of the "/". Press return.
12) You have now made this line schedule a 3 games series with these 2 teams (/v:1.dat /h:2.dat /n:3)
13) If you edit the next 11 lines in addition to this line, you will make round 2 a 3 game series of all 12 scheduled games. You now have a 36 game series schedule for each team, each team plays 36 games with 18 HOME and 18 AWAY games, and each team plays the other teams an equal number of times (12 times each). Now try adding 4 games to another single round (the next 12 line of the spreadsheet), and you can see how quickly the schedule grows! (It'll grow to 54 games apiece, 27 HOME, 27 AWAY and play each other 18 times – 9 HOME and 9 AWAY!)
14) This schedule, staring the "not ready for prime time" players, is not ready for release! This is because it has a pattern. So, we can randomize this schedule (after we include the games series).
15) I use Notepad++ for all my text editing needs. It's free and is a BIG improvement on the Notepad that comes with MS Windows. You can get it here: **https://notepad-plus-plus.org/**
16) Copy Cells E3:E50 and paste into Notepad++.
17) From within Notepad++, make sure your cursor is on line 1 column 1. From the menu select: **Edit → Line Operations → Randomize Line Order**

Now, we're going to make the schedule random. I am using 2 rounds to show as a sample. One round has 3 games scheduled per line as shown below. The other ones are simply single games (see "ORIGINAL LIST")::

ORIGINAL LIST

```
/v:1.dat  /h:2.dat
/v:1.dat  /h:3.dat
/v:1.dat  /h:4.dat
/v:2.dat  /h:1.dat
/v:2.dat  /h:3.dat
/v:2.dat  /h:4.dat
/v:3.dat  /h:1.dat
/v:3.dat  /h:2.dat
/v:3.dat  /h:4.dat
/v:4.dat  /h:1.dat
/v:4.dat  /h:2.dat
/v:4.dat  /h:3.dat
/v:1.dat  /h:2.dat /n:3
/v:1.dat  /h:3.dat /n:3
/v:1.dat  /h:4.dat /n:3
/v:2.dat  /h:1.dat /n:3
/v:2.dat  /h:3.dat /n:3
/v:2.dat  /h:4.dat /n:3
/v:3.dat  /h:1.dat /n:3
/v:3.dat  /h:2.dat /n:3
/v:3.dat  /h:4.dat /n:3
/v:4.dat  /h:1.dat /n:3
/v:4.dat  /h:2.dat /n:3
/v:4.dat  /h:3.dat /n:3
```

RANDOM LIST

```
/v:3.dat  /h:1.dat /n:3
/v:3.dat  /h:1.dat
/v:3.dat  /h:2.dat
/v:2.dat  /h:1.dat /n:3
/v:4.dat  /h:3.dat /n:3
/v:2.dat  /h:4.dat
/v:1.dat  /h:4.dat /n:3
/v:4.dat  /h:3.dat
/v:2.dat  /h:4.dat /n:3
/v:3.dat  /h:2.dat /n:3
/v:1.dat  /h:3.dat /n:3
/v:4.dat  /h:1.dat /n:3
/v:4.dat  /h:1.dat
/v:1.dat  /h:3.dat
/v:1.dat  /h:2.dat
/v:2.dat  /h:3.dat /n:3
/v:3.dat  /h:4.dat /n:3
/v:3.dat  /h:4.dat
/v:1.dat  /h:2.dat /n:3
/v:4.dat  /h:2.dat /n:3
/v:1.dat  /h:4.dat
/v:4.dat  /h:2.dat
/v:2.dat  /h:3.dat
/v:2.dat  /h:1.dat
```

RANDOMIZED 3x

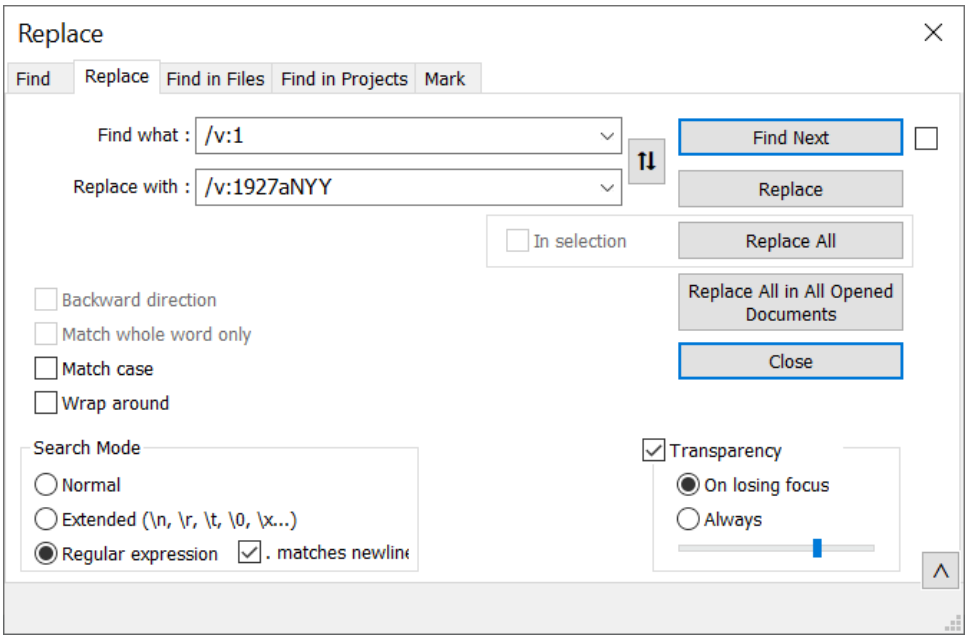
```
/v:2.dat  /h:4.dat
/v:2.dat  /h:1.dat
/v:3.dat  /h:4.dat
/v:4.dat  /h:3.dat /n:3
/v:2.dat  /h:1.dat /n:3
/v:1.dat  /h:2.dat
/v:3.dat  /h:2.dat /n:3
/v:4.dat  /h:2.dat /n:3
/v:4.dat  /h:1.dat /n:3
/v:4.dat  /h:3.dat
/v:3.dat  /h:1.dat
/v:4.dat  /h:1.dat
/v:1.dat  /h:4.dat /n:3
/v:3.dat  /h:4.dat /n:3
/v:4.dat  /h:2.dat
/v:1.dat  /h:3.dat /n:3
/v:1.dat  /h:3.dat
/v:2.dat  /h:4.dat /n:3
/v:3.dat  /h:2.dat
/v:2.dat  /h:3.dat /n:3
/v:2.dat  /h:3.dat
/v:1.dat  /h:2.dat /n:3
/v:1.dat  /h:4.dat
/v:3.dat  /h:1.dat /n:3
```

Notice that the first 2 lines in the "RANDOM LIST" have the same teams playing against each other! You could delete one of these lines and make the other line a 4 game series (/n:4) or re-shuffle the list several times to get the effect as shown in the "RANDOMIZED x3" list. Not only did the number of series games get shuffled, but also AWAY and HOME teams are randomized. If I do this randomization 3 or 4 times, It will keep shuffling the list, making it a completely random schedule!

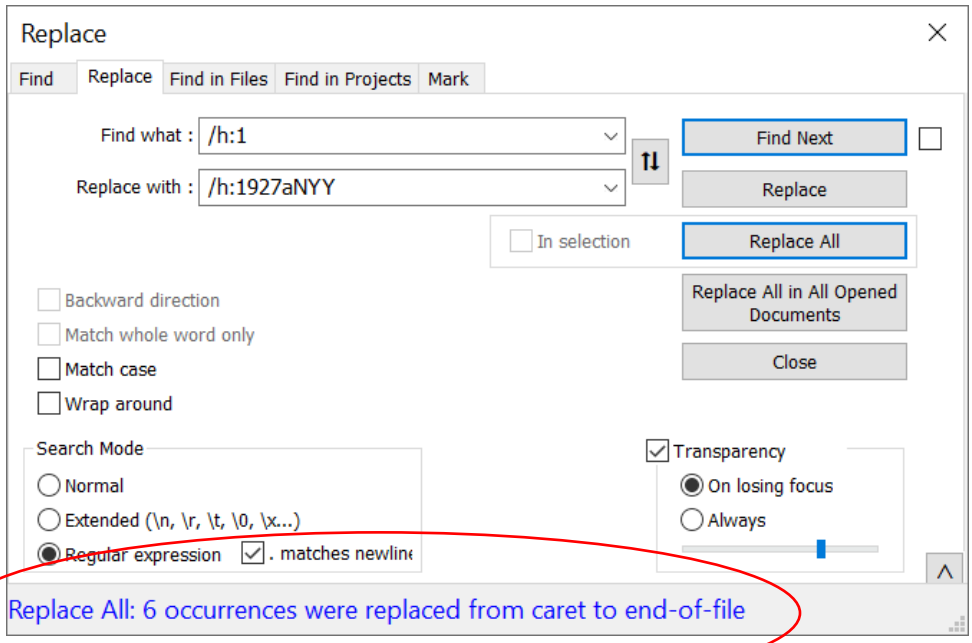
Now, to **finalize** the schedule, you have a choice:

- Either keep the schedule file as is, and rename your teams to 1.dat, 2.dat, 3.dat, etc., OR
- Using the power of Notepad++ replace all the roster team names from numbers to actual team names as shown below:

a. First, insure your cursor is placed at the beginning of line 1 column 1 in order for the replace feature to correctly change all occurrences. Select Search → Replace from the menu and change the parameters in the "Replace" dialog box as shown in the figure below:



b. Press the "Replace All" button. This will change every occurrence from /v:1 (all visiting teams named '1' to visiting team named '1927aNYY
Do the same for all home teams named '1' to home team named '1927aNYY'



c. Continue to make all team name changes using the patterns described above. The bottom left portion of the window will tell you how many replacements the editor made (shown in BLUE text). Remember to Save the file when completed. **NOTE:** There should be an equal number of occurrences for every replacement you made! My example only had 6 changes made per Replace, but it's only an example. **CAUTION:** Be careful not to change the number of games (/n:3, etc.) when Replacing team names.
d. Make any other changes to the schedule file manually (additional parameters to certain games, etc.)
e. Save, then Exit the editor and rename your file if needed using the .ser file extension. **REMEMBER –** Because SBS is an MS-DOS program, MS-DOS file names must be 8 characters long or less. 1927Al.ser or 1927MLB.ser would identify a series schedule file for the 1927 season.

That's it! You have created a series schedule using MS Excel and Notepad++. It is best to perform a "run thru" or "trial run" using the SBS program to insure that your newly created .series file works as advertised without problems.

The Seventh Inning Stretch (my namesake) – Modifying a schedule (.sch) file with Notepad++

CAUTION: There is a BIG "gottcha" when editing .sch files. The new filenames you're using, must be the exact same length as the filenames you're going to replace! This is because each record in this file is a strict 430 characters.

I use Notepad++ as my default text editor, as it is FREE and has so many more features than does the default Notepad that comes with MS Windows. However, this time, because of the formatting style used by the .sch file, Notepad++ displays this file in its entirety on one very long line! To see this entire line, select "View → Word Wrap" to enable the display to view the entire line as a scrolling line of text!

OR… You can open the file using Notepad (the default text editor that comes with MS Windows). This editor auto-wraps a long line.

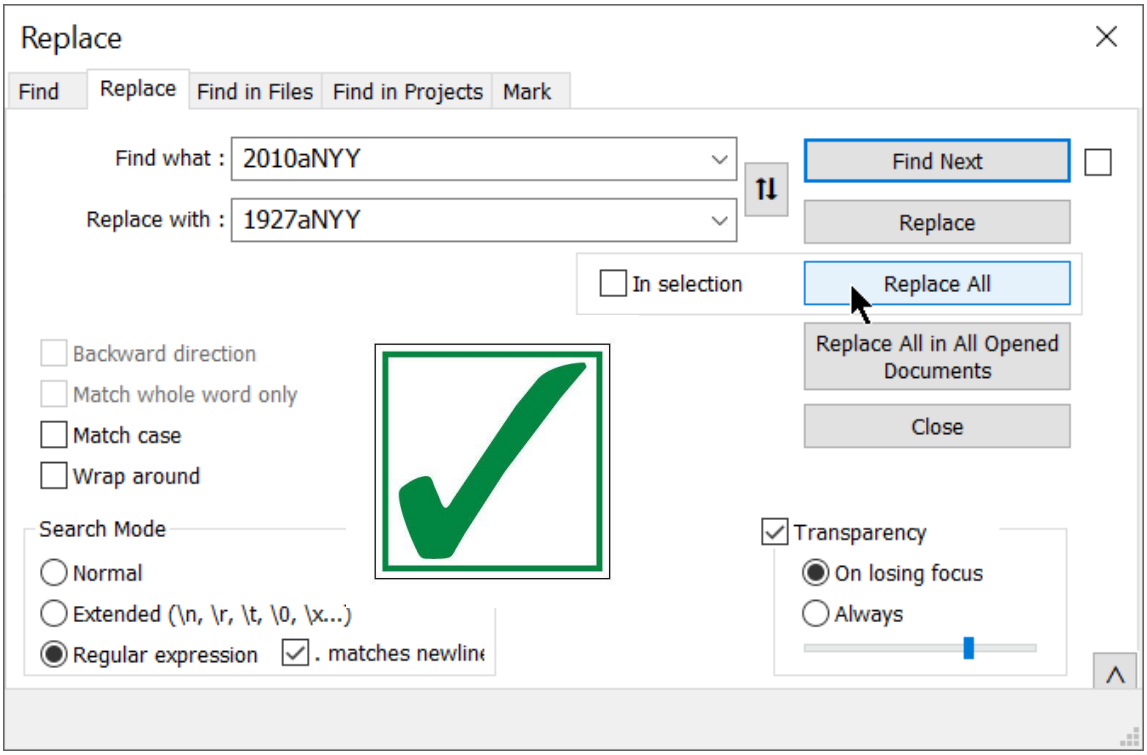
Either way, here is the secret to editing this file:

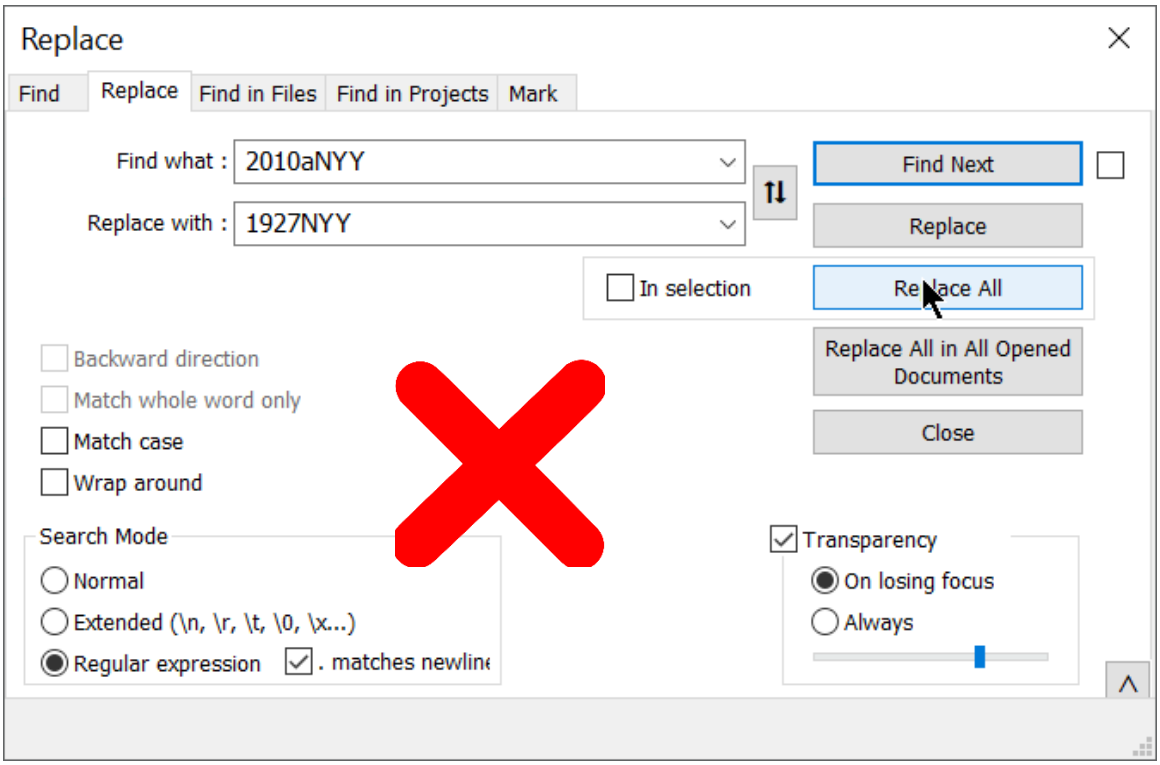
ALWAYS make a copy of any file you're about to edit. First, make sure you use a known good schedule file and make a copy of it. If you somehow create a new file that doesn't work, you can always "roll back" to your saved original. The #1 thing to remember is that you must maintain the original format of the file… no extra blanks or shortened records (maintain a record length of 430 characters).

Now, all you need to do is to simply **Replace All** old <filename> with new <filename> in the schedule file. Do this for all team names you want to replace. **WARNING:** This effort is pretty time consuming!

If you want to also edit the dates in the .sch file, the date format is **MM/DD/YYYY**

CAUTION: old <filename> and new <filename> must have the same number of characters, or your new schedule won't work, or will produce strange results at the least!. This following example will work fine, because both filenames are of the same length:





The above method will NOT work, if old <filename> and new <filename> are not the same length! While the text editor thinks the change is fine, the SBS program will not when you run it, because of the filename length differences.

Continue your edits, replacing old team names with new team names, and make sure to save your file when you're done! When complete try a trial run thru to see if the simulator plays all the games in your new .sch file without problems. THAT'S IT! If you goof it all up, no worries, as you have an original copy to try again with.

Alternatives - Other ways to edit the .sch file might be one of the following:

Convert the file to a more common format: You can try converting the SBS .sch file to a more common file format such as CSV or Excel. Once converted, you can modify the schedule in the new format and then reconvert it back to the SBS .sch format.

OR…

Use a third-party tool such as the "schedule editor" tool found in the stand-alone program "Run SBS". It is available here:

https://runsbs.tripod.com/

There are instructions on their website on how to use this tool.

Roll Your Own - The SBS User Manual is a great place to start to show you how a .ser or .sch file is laid out. For you programmers, Appendix B explains the file formats not only of the .ser and .sch files, but ALL the files used by SBS. Feel adventurous? To get you started, here's the beginnings of a Python program that reads the .sch file (2010.MLB.sch):

```
# Define a list to store the dictionaries of parameters
params_list = []

# Open the SBS .sch file in read mode
with open("c:/sbs/2010mlb.sch", "r") as f:
    # Read the contents of the file
    contents = f.read()
    # Split the contents of the file by newline character
    lines = contents.split("\n")
    # Loop through each line in the file
    for line in lines:
        # Skip empty lines
        if not line.strip():
            continue
        # Split the line into individual parameters
        params = line.split(",")
        # Check if the line contains at least 6 parameters
        if len(params) < 6:
            print(f"Error: Line '{line}' contains fewer than 6 parameters.")
            continue
        # Create a dictionary to store the parameters
        param_dict = {
            "Team1": params[0].strip(),
            "Team2": params[1].strip(),
            "Location": params[2].strip(),
            "Date": params[3].strip(),
            "Time": params[4].strip(),
            "GameType": params[5].strip()
        }
        # Add the dictionary of parameters to the list
        params_list.append(param_dict)

# Print the list of parameters
print(params_list)
```

Next Inning: Adapting Your Schedules For Use With SBS