

gurps — A LaTeX package for GURPS typesetting*

Nathanael Farley†

Released 1970/01/01

Abstract

GURPS is an excellent RPG system. \LaTeX is an excellent typesetting system. Together, they make excellent game aids. At least, they can with this package!

Contents

1	Motivation	1
2	Quickstart	1
3	In-depth usage	1
3.1	Character creation	2
3.2	Typographical features	3
3.3	Online policy macros	3
3.3.1	<code>\SJGamesOnlinePolicyDisclaimer text</code>	3
3.3.2	<code>\SJGamesOnlinePolicyNotice text</code>	3
3.3.3	<code>\SJGamesOnlinePolicyGameAid text</code>	3
4	Implementation	4
4.1	Character environment	5
4.2	A few toggles	7
4.3	Printing the character	7
4.3.1	Basic attributes	8
4.3.2	Secondary Characteristics	9
4.3.3	Stat block formatting	13

1 Motivation

I often like to make *GURPS* game aids for my own games. Over time, I started to collect various macros that made it easier. Eventually, I had enough game aids that I decided to write the macros up in a package, then I realised I could release on CTAN and share the *GURPS* goodness!

2 Quickstart

A brief overview of the commands in this package can be found in [Table 1](#).

3 In-depth usage

The macros in this package can be divided into roughly three sections:

1. character creation
2. common typographical features of *GURPS*
3. macros for abiding by SJGames online policy

*This file describes version v0.3, last revised 1970/01/01.

†E-mail: nasfarley88@gmail.com

Table 1: Basic commands in gurps package.

Macro	Example
<code>\dice{⟨no. of dice⟩}[⟨modifier⟩]</code>	3d+2
<code>\gurpsbook{⟨name of book⟩}[⟨page(s)⟩]</code>	<i>GURPS Zombies</i> p. 1
<code>\gurps</code>	GURPS
<code>\SJGamesOnlinePolicyDisclaimer</code>	The material presented here is my original creation, intended for use with the GURPS system from Steve Jackson Games . This material is not official and is not endorsed by Steve Jackson Games.
<code>\SJGamesOnlinePolicyNotice</code>	GURPS is a registered trademark of Steve Jackson Games , and is copyrighted by Steve Jackson Games. All rights are reserved by Steve Jackson Games. This material is used here in accordance with the SJ Games online policy .
<code>\SJGamesOnlinePolicyGameAid{⟨author name⟩}</code>	GURPS is a trademark of Steve Jackson Games , and its rules and art are copyrighted by Steve Jackson Games . All rights are reserved by Steve Jackson Games. This game aid is the original creation of AUTHOR NAME and is released for free distribution, and not for resale, under the permissions granted in the Steve Jackson Games Online Policy .
character environment, \ST, \advantage, etc.	See subsection 3.1

3.1 Character creation

character Character (i.e. NPCs) creation can be tricky and involve a *lot* of boilerplate code. This package provides most of the boilerplate code so you only specify what is *different* from assumed values (e.g. If your character has a ST 13 but DX 10, IQ 10, HT 10, you only need to specify ST)

It is not currently possible to (easily) change the look of the stats block for the character environment. This is planned for a future release.

Warning!

This feature is still in alpha. The external API (`\skill`, `\advantage`, etc.) should not change, but the internal \TeX and layout is *very* likely to change.

On to an example¹:

```
% \begin{character}
%   \ST{9}
%   \HP{12}
%   \advantage{Natural afro}[1]
%   \disadvantage{Big teeth}[-1]
%   \skill{Nunchuck}[DX/Easy]{16}
%   \meleeattack{
%     name=Punch,
%     level=18,
%     damage={5d cr},
%     reach={C,1},
%     notes={Believe it!}
%   }
% \end{character}
%
```

creates ...

Basic Attributes ST 9[-10]; DX 10[0]; IQ 10[0]; HT 10[0]

Secondary Characteristics HP 12[6]; Per 10[0]; Will 10[0]; FP 10[0]; Basic Speed 5[0]; Basic Move 5[0]

Other DR 0; Dodge 8; SM 1; sw 1d-1; thr 1d-2

¹Any resemblance to Napoleon Dynamite is completely ... intentional!

Advantages Natural afro[1]; T_EX-nical afro[1]

Disadvantages Big teeth[-1]

Skills Nunchuck 16[20]

Attacks

– *Punch* 18 *5d cr* Believe it! *Reach: C,1*

3.2 Typographical features

`\dice` Using the `\dice{⟨no. of dice⟩}[⟨modifier⟩]` provides dice notation consistent with **GURPS** materials. E.g.

`\dice{3}` → 3d
`\dice{4}[-1]` → 4d-1
`\dice{2}[1]` → 2d+1

`\gurpsbook` The command `\gurpsbook{⟨book name⟩}[⟨page number(s)⟩]` formats **GURPS** books like Steve Jackson Games. E.g.

`\gurpsbook{High Tech}` → **GURPS High Tech**
`\gurpsbook{Zombies}[3]` → **GURPS Zombies** p. 3
`\gurpsbook{Warehouse~23}[1, 3--5]` → **GURPS Warehouse 23** pp. 1, 3–5

3.3 Online policy macros

`\SJGamesOnlinePolicyDisclaimer` The macros `\SJGamesOnlinePolicyDisclaimer`, `\SJGamesOnlinePolicyNotice` and `\SJGamesOnlinePolicyGameAid{⟨author name⟩}` provide common boilerplate statements taken directly from **SJGames online policy** with small modifications to make them more suitable to game aids created with Lua_{T_EX}.

`\gurps` The command `\gurps` is also included to provide bold and italicised **GURPS** as requested in the online policy

3.3.1 `\SJGamesOnlinePolicyDisclaimer` text

Using the code `\SJGamesOnlinePolicyDisclaimer` produces the text:

The material presented here is my original creation, intended for use with the **GURPS** system from **Steve Jackson Games**. This material is not official and is not endorsed by Steve Jackson Games.

3.3.2 `\SJGamesOnlinePolicyNotice` text

Using the code `\SJGamesOnlinePolicyOnlineNotice` produces the text:

GURPS is a registered trademark of **Steve Jackson Games**, and is copyrighted by Steve Jackson Games. All rights are reserved by Steve Jackson Games. This material is used here in accordance with the SJ Games **online policy**.

3.3.3 `\SJGamesOnlinePolicyGameAid` text

Using the code `\SJGamesOnlinePolicyGameAid{NAME}` produces the text:

GURPS is a trademark of **Steve Jackson Games**, and its rules and art are copyrighted by **Steve Jackson Games**. All rights are reserved by Steve Jackson Games. This game aid is the original creation of NAME and is released for free distribution, and not for resale, under the permissions granted in the **Steve Jackson Games Online Policy**.

4 Implementation

Loading required libraries and the lua logic for this class.

```

1 {\*package}
2 \RequirePackage{etoolbox}
3 \RequirePackage{xparse}
4 \RequirePackage{xspace}
5 \RequirePackage{luacode}
6 \RequirePackage{tikz}
7 \RequirePackage{calc}
8 \PassOptionsToPackage{inline}{enumitem}
9 \RequirePackage[inline]{enumitem}
10
11 \RequirePackage{hyperref}
12 \RequirePackage{xstring}
13 \RequirePackage{xkeyval}
14
15 \luadirect{require("gurps.lua")}
16 \luadirect{require("gurps_character.lua")}

```

`\gurps` Steve Jackson Games asks that the name ***GURPS*** is always in bold and italicised. This macro provides that.

```

17 \NewDocumentCommand{\gurps}{s}{%
18   \IfBooleanTF#1
19   % With star
20   {\href{http://www.sjgames.com/gurps/}{\textbf{\textit{GURPS}}\xspace}}
21   % Without star
22   {\textbf{\textit{GURPS}}\xspace}%
23 }

```

`\dice` Notates dice in the ***GURPS*** way (e.g. drops the 6 from d6) and puts it in an `\mbox`. First argument number of dice. Second argument is a modifier and optional. E.g. `\dice{3}[1]` creates 3d+1.

```

24 \NewDocumentCommand{\dice}{mO{0}}{%
25   \luadirect{print_dice([[#1]], [[#2]])}%
26 }

```

`\gurps@porpp` Helper macro to print pp. or p. when listing page numbers

```

27 \NewDocumentCommand{\gurps@porpp}{m}{\IfInteger{#1}{p.}{pp.}}

```

`\gurpsbook` Works like this: ***GURPS Horror*** p. 45 or ***GURPS Banestorm*** pp. 3, 45–48

```

28 \NewDocumentCommand{\gurpsbook}{mo}{%
29   \textbf{\textit{\gurps #1}}\IfValueT{#2}{ \gurps@porpp{#2}~#2}%
30 }

```

`\SJGames` Steve Jackson Games printed with link (starred) or without (unstarred).

```

31 \NewDocumentCommand{\SJGames}{s}{%
32   \IfBooleanTF#1%
33   % With star
34   {\href{http://www.sjgames.com/}{Steve~Jackson~Games}\xspace}%
35   % Without star
36   {Steve~Jackson~Games\xspace}%
37 }

```

`\SJGamesOnlinePolicyDisclaimer` Online policy disclaimer, basically straight from the Steve Jackson Games online policy.

```

38 \NewDocumentCommand{\SJGamesOnlinePolicyDisclaimer}{}{%
39   The material presented here is my original creation, intended for use with the
40   \gurps* system from \SJGames*. This material is not official and is
41   not endorsed by \SJGames.
42 }

```

`\SJGamesOnlinePolicyNotice` The disclaimer *almost* as it appears on the online policy. The words ‘the art’ have been removed since by default no art (including logos) are included in \TeX documents.

```

43 \NewDocumentCommand{\SJGamesOnlinePolicyNotice}{}{%
44 \gurps* is a registered trademark of \SJGames*, and is copyrighted by
45 \SJGames. All rights are reserved by \SJGames. This material is used
46 here in accordance with the SJ Games
47 \href{http://www.sjgames.com/general/online_policy.html}{online policy}.
48 }

```

\SJGamesOnlinePolicyGameAid This text is required for all game aid's produced for **GURPS** but without an official license. It takes one argument: author name.

```

49 \NewDocumentCommand{\SJGamesOnlinePolicyGameAid}{m}{%
50 \gurps* is a trademark of \SJGames*, and its rules and art are
51 copyrighted by \SJGames*. All rights are reserved by
52 \SJGames. This game aid is the original creation of #1 and
53 is released for free distribution, and not for resale, under the
54 permissions granted in the
55 \href{http://www.sjgames.com/general/online_policy.html}%
56 {\SJGames Online Policy}.
57 }

```

4.1 Character environment

This section lists the character environment and all the macros that go into it.

Future plans!

At the moment, there's a real mix of commands with and without @ that should probably be switched to be internal/external so that the user can configure them (e.g. \gurps@char@print@attr). In the future, this will happen. For now ... feel free to edit them and see what happens! :)

Also in the future, there will eventually be 'theme' support such that the user can configure different ways of printing the character (e.g. disable points for a monster manual, use a tcolorbox for the statblock, etc.). Can't say when yet, but pull requests are welcome!

character, character* This environment defines a 'character' i.e. anything with full stats. A character* will define a character but not print it.

It takes [*character key*] which provides a means to access the character later in the file. It is set with \SetCharacterKey and reset with \ResetCharacterKey. In order to access the character later, just use \SetCharacterKey{*character key*}, use the commands (e.g. \gurps@char@advantages) and then reset².

```

58 \NewDocumentEnvironment{character}{sO{}}{%
59 \SetCharacterKey{#2}%
60 \luadirect{new_character(_GCHARACTERKEY)}%
61 }{%
62 \gurps@char@checkandfixattrsandpoints%
63 \IfBooleanTF{#1}{\printcharacter}%
64 \ResetCharacterKey%
65 }

```

\SetCharacterKey Sets the key for the current character. This sets the lua string _GCHARACTER which acts as a key for the global Lua table _GCHARACTERS.

```

66 \NewDocumentCommand{\SetCharacterKey}{m}{%
67 \luadirect{_GCHARACTERKEY = \luastring{\unexpanded{#1}}}%
68 }

```

\ResetCharacterKey Resets the character key to _.

```

69 \NewDocumentCommand{\ResetCharacterKey}{}{%
70 \SetCharacterKey{_%
71 }

```

²Although resetting is not strictly necessary, global variables are a Bad Thing™ so we reset in order to emulate some kind of scope. Resetting just changes the character key to _.

`\gurps@char@insertattr` Macro for inserting an attribute to the character. Anything inserted with `:nolatemx:` in the name is not added to the character. (This is to aid importing directly from a `*.gcs` file.)

First, we define the `xkeval` keys for the function. These are set up to convert the values directly into the right Lua types.

```

72 \define@key{gurps@char@insertattr}{name}{%
73   \def\gurps@char@insertattr@name{\unexpanded{[[#1]]}} %TODO check this << has
74   %fixed it
75 \define@key{gurps@char@insertattr}{type}{%
76   \def\gurps@char@insertattr@type{\luastring{\unexpanded{#1}}}}
77 \define@key{gurps@char@insertattr}{level}{%
78   \def\gurps@char@insertattr@level{#1}}
79 \define@key{gurps@char@insertattr}{diceexpr}{%
80   \def\gurps@char@insertattr@diceexpr{\luastring{#1}}}
81 \define@key{gurps@char@insertattr}{points}{%
82   \def\gurps@char@insertattr@points{#1}}
83 \define@key{gurps@char@insertattr}{basedon}{%
84   \def\gurps@char@insertattr@basedon{\luastring{\unexpanded{#1}}}}
85 \define@key{gurps@char@insertattr}{difficulty}{%
86   \def\gurps@char@insertattr@difficulty{\luastring{\unexpanded{#1}}}}

```

Then we define `\gurps@char@insertattr{<key-value pairs>}`. The macro first checks we are in a character environment (throwing an error if we are not), defines the keys, then does the Lua stuff.

Side note: anything with `:nolatemx:` in the name is discarded and not put into the character.

```

87 \NewDocumentCommand{\gurps@char@insertattr}{m}{
88   \def\myenvname{character}
89   \ifx\@currenvir\myenvname
90     % pass
91   \else
92     \PackageError{gurps}{Not in character environment! Currently in \@currenvir{} environment.}{}
93   \fi
94
95   \setkeys{gurps@char@insertattr}{
96     name=NotSet,
97     type=NotSet,
98     points=nil,
99     level=nil,
100    basedon=DX,
101    difficulty=NotSet,
102    diceexpr=NotSet,
103  }
104
105   \setkeys{gurps@char@insertattr}{#1}
106
107   % If the name has the tag :nolatemx:, don't add it! Otherwise, add it.
108   \luadirect{
109     if not string.find(\luastring{\gurps@char@insertattr@name}, \luastring{:nolatemx:}) then
110       % If arg is blank (e.g. identity()), then this returns nil.
111       function identity(x)
112         return x
113       end
114
115       x = {
116         name=\gurps@char@insertattr@name,
117         type=\gurps@char@insertattr@type,
118         points=\gurps@char@insertattr@points,
119         level=identity(\gurps@char@insertattr@level),
120         basedon=\gurps@char@insertattr@basedon,
121         difficulty=\gurps@char@insertattr@difficulty,
122         diceexpr=\gurps@char@insertattr@diceexpr
123       }
124
125       if _else_packageerror(
126         function() return is_valid_type(x.type) end,

```

```

127     "Type '" .. x.type .. "' is not a valid attribute type!"
128     .. " Did you remember to set the type?"
129 )
130
131 if_else_packageerror(
132     function() return is_valid_points(x.points) end,
133     "Points value '" .. tostring(x.points) .. "' is not valid!"
134 )
135
136 if x.type == "skill" or x.type == "spell" then
137     if_else_packageerror(
138         function() return is_valid_difficulty(x.difficulty) end,
139         "Difficulty value '" .. x.difficulty .. "' is not valid!"
140     )
141     end
142     insert_attr(_GCHARACTERKEY, x)
143 end
144 }%
145 }

```

4.2 A few toggles

These should be set by lua stuff.

```

146 \newtoggle{hasadvantages}
147 \newtoggle{hasdisadvantages}
148 \newtoggle{hastraits}
149 \newtoggle{hasskills}
150 \newtoggle{hasspells}
151 \newtoggle{hasattacks}

```

\gurps@char@checkandfixtoggles This command fixes all the toggles used in printing the character.

```

152 \NewDocumentCommand{\gurps@char@checkandfixtoggles}{}{%
153 \luadirect{check_and_fix_toggles(_GCHARACTERKEY)}%
154 }

```

4.3 Printing the character

\printcharacter (This function is mostly for debug at the moment)

```

155 \NewDocumentCommand{\printcharacter}{}{%
156 \IfValueTF{#1}{\SetCharacterKey{#1}}{}%
157 \gurps@char@checkandfixtoggles%
158 \charactersection*{Basic Attributes}%
159 \gurps@char@basicattributes%
160 \charactersection*{Secondary Characteristics}%
161 \gurps@char@secondarycharacteristics%
162 \charactersection*{Other}%
163 \gurps@char@properties%
164 \iftoggle{hasadvantages}{%
165     \charactersection*{Advantages}%
166     \gurps@char@advantages%
167 }{}%
168 \iftoggle{hasdisadvantages}{%
169     \charactersection*{Disadvantages}%
170     \gurps@char@disadvantages%
171 }{}%
172 \iftoggle{hasskills}{%
173     \charactersection*{Skills}%
174     \gurps@char@skills%
175 }{}%
176 \iftoggle{hasspells}{%
177     \charactersection*{Spells}%
178     \gurps@char@spells%

```

```

179 }{}%
180 \iftoggle{hasattacks}{%
181   \charactersection*{\strut Attacks}\ \linebreak%
182   \gurps@char@attacks%
183 }{}%
184 \IfValueTF{#1}{\ResetCharacterKey}{}%
185 }

```

\gurps@char@getattrlevel

```

186 \NewDocumentCommand{\gurps@char@getattrlevel}{m}{%
187   \luadirect{
188     x = filter(({name=\luastring{#1}}, get_character(_GCHARACTERKEY))
189     if x then
190       tex.sprint(x.level)
191     else
192       tex.sprint([[ \PackageError{gurps}{Stat '#1' not found!}]]))
193     end
194   }%
195 }

```

4.3.1 Basic attributes

\ST Sets the strength of the current character

```

196 \NewDocumentCommand{\ST}{mO{nil}}{%
197   \gurps@char@insertattr{
198     name=ST,
199     type=basic_attribute,
200     level=#1,
201     points=#2
202   }%
203 }

```

\DX Sets the dexterity of the current character

```

204 \NewDocumentCommand{\DX}{mO{nil}}{%
205   \gurps@char@insertattr{
206     name=DX,
207     type=basic_attribute,
208     level=#1,
209     points=#2
210   }%
211 }

```

\IQ Sets the IQ of the current character

```

212 \NewDocumentCommand{\IQ}{mO{nil}}{%
213   \gurps@char@insertattr{
214     name=IQ,
215     type=basic_attribute,
216     level=#1,
217     points=#2
218   }%
219 }

```

\HT Sets the health of the current character

```

220 \NewDocumentCommand{\HT}{mO{nil}}{%
221   \gurps@char@insertattr{
222     name=HT,
223     type=basic_attribute,
224     level=#1,
225     points=#2
226   }%
227 }

```


4.3.2 Secondary Characteristics

This section defines secondary characteristics (as *GURPS Basic Set*). Points are set in a later function `\gurps@char@checkandfixattrsandpoints` if they are not set here.

`\HP` Sets the HP of the current character

```
228 \NewDocumentCommand{\HP}{mO{nil}}{%
229   \gurps@char@insertattr{
230     name=HP,
231     type=secondary_characteristic,
232     level=#1,
233     points=#2
234   }%
235 }
```

`\Per` Sets the perception of the current character

```
236 \NewDocumentCommand{\Per}{mO{nil}}{%
237   \gurps@char@insertattr{
238     name=Per,
239     type=secondary_characteristic,
240     level=#1,
241     points=#2
242   }%
243 }
```

`\Will` Sets the will of the current character

```
244 \NewDocumentCommand{\Will}{mO{nil}}{%
245   \gurps@char@insertattr{
246     name=Will,
247     type=secondary_characteristic,
248     level=#1,
249     points=#2
250   }%
251 }
```

`\FP` Sets the FP of the current character

```
252 \NewDocumentCommand{\FP}{mO{nil}}{%
253   \gurps@char@insertattr{
254     name=FP,
255     type=secondary_characteristic,
256     level=#1,
257     points=#2
258   }%
259 }
```

`\SM` Sets the SM of the current character

```
260 \NewDocumentCommand{\SM}{m}{%
261   \gurps@char@insertattr{
262     name=SM,
263     type=property,
264     level=#1
265   }%
266 }
```

`\DR` Sets the DR of the current character. Takes `\DR[<specialisation>][<level>]`.

```
267 \NewDocumentCommand{\DR}{om}{
268   \IfValueTF{#1}{%
269     \gurps@char@insertattr{%
270       name={DR (#1)},%
271       type=property,%
272       level=#2%
273     }%%
274   }{%
```

```

275 \gurps@char@insertattr{%
276     name=DR,%
277     type=property,%
278     level=#2%
279 }%
280 }%
281 }

\thr Sets the thr of the current character
282 \NewDocumentCommand{\thr}{m}{
283 \gurps@char@insertattr{
284     name=thr,
285     type=property,
286     diceexpr=#1
287 }%
288 }

\sw Sets the sw of the current character
289 \NewDocumentCommand{\sw}{m}{
290 \gurps@char@insertattr{
291     name=sw,
292     type=property,
293     diceexpr=#1
294 }%
295 }

\basicspeed Sets the Basic Speed of the current character
296 \NewDocumentCommand{\basicspeed}{mO{nil}}{
297 \gurps@char@insertattr{
298     name={Basic Speed},
299     type=secondary_characteristic,
300     level=#1,
301     points=#2
302 }%
303 }

\basicmove Sets the Basic Move of the current character
304 \NewDocumentCommand{\basicmove}{mO{nil}}{
305 \gurps@char@insertattr{
306     name={Basic Move},
307     type=secondary_characteristic,
308     level=#1,
309     points=#2
310 }%
311 }

\gurps@char@insertattack

312 \define@key{gurps@char@insertattack}{name}{
313 \def\gurps@char@insertattack@name{\luastring{\unexpanded{#1}}}}
314 \define@key{gurps@char@insertattack}{type}{
315 \def\gurps@char@insertattack@type{\luastring{\unexpanded{#1}}}}
316 \define@key{gurps@char@insertattack}{damage}{
317 \def\gurps@char@insertattack@damage{\luastring{\unexpanded{#1}}}}
318 \define@key{gurps@char@insertattack}{level}{\def\gurps@char@insertattack@level{#1}}
319 \define@key{gurps@char@insertattack}{reach}{
320 \def\gurps@char@insertattack@reach{\luastring{\unexpanded{#1}}}}
321 \define@key{gurps@char@insertattack}{range}{
322 \def\gurps@char@insertattack@range{\luastring{\unexpanded{#1}}}}
323 \define@key{gurps@char@insertattack}{notes}{
324 \def\gurps@char@insertattack@notes{\luastring{\unexpanded{#1}}}}
325
326 % Set default values
327 \setkeys{gurps@char@insertattack}{

```

```

328   name=NotSet,
329   type=NotSet,
330   damage=0d,
331   level=nil,
332   reach=NotSet,
333   range=NotSet,
334   notes={},
335 }
336 \NewDocumentCommand{\gurps@char@insertattack}{sm}{
337   \def\@myenvname{character}
338   \ifx\@currentvir\@myenvname
339     % pass
340   \else
341     \PackageError{gurps}{Not in character environment! Currently in \@currentvir{ } environment.}{}
342   \fi
343
344   \IfBooleanTF{#1}{}{
345     \setkeys{gurps@char@insertattack}{
346       name=NotSet,
347       type=NotSet,
348       damage=0d,
349       level=nil,
350       reach=NotSet,
351       range=NotSet,
352       notes={},
353     }
354
355     \setkeys{gurps@char@insertattack}{#2}
356   }
357   % \show\gurps@char@insertattack@name
358   % \show\gurps@char@insertattack@type
359   % \show\gurps@char@insertattack@level
360   % \show\gurps@char@insertattack@damage
361   % \show\gurps@char@insertattack@reach
362   % \show\gurps@char@insertattack@range
363   % \show\gurps@char@insertattack@notes
364   \luadirect{x = {
365     name=\gurps@char@insertattack@name,
366     type=\gurps@char@insertattack@type,
367     level=\gurps@char@insertattack@level,
368     damage=\gurps@char@insertattack@damage,
369     reach=\gurps@char@insertattack@reach,
370     range=\gurps@char@insertattack@range,
371     notes=\gurps@char@insertattack@notes,
372   }
373   if_else_packageerror(
374     function() return is_valid_type(x.type) end,
375     "Type '" .. x.type .. "' is not a valid attribute type!"
376     .. " Did you remember to set the type?"
377   )
378   insert_attr(_GCHARACTERKEY, x)
379 }
380 }
381
\ rangedattack

382 \NewDocumentCommand{\rangedattack}{m}{%
383   \setkeys{gurps@char@insertattack}{#1}%
384   \setkeys{gurps@char@insertattack}{type=ranged_attack}
385
386   \gurps@char@insertattack*{ }
387 }

```

\gurps@char@print@attack

```

388 \NewDocumentCommand{\gurps@char@print@attack}{mmmO{RangeOrReach}mm}{%
389 \textbf{\textit{\#1 \fbox{\#2} \#3}} \#6 \textit{\#4: \#5}%
390 }

```

\meleeattack

```

391 \NewDocumentCommand{\meleeattack}{m}{%
392 \setkeys{gurps@char@insertattack}{\#1}%
393 \setkeys{gurps@char@insertattack}{type=melee_attack}
394
395 \gurps@char@insertattack*{}
396 }

```

\advantage Adds an advantage to a character. NOTE: this only works in a character.

```

397 \NewDocumentCommand{\advantage}{mO{nil}}{%
398 \gurps@char@insertattr{%
399 name={\#1},%
400 type=advantage,%
401 points=\#2%
402 }%
403 }

```

\disadvantage Adds a disadvantage to a character. NOTE: this only works in a character or lens environment.

```

404 \NewDocumentCommand{\disadvantage}{mO{nil}}{%
405 \gurps@char@insertattr{%
406 name={\#1},%
407 type=disadvantage,%
408 points=\#2%
409 }%
410 }

```

\levelledadvantage Adds an advantage to a character. NOTE: this only works in a character or lens environment.

```

411 % TODO finish this!
412 \NewDocumentCommand{\levelledadvantage}{mmO{nil}}{%
413 \gurps@char@insertattr{%
414 name={\#1},%
415 type=advantage,%
416 level=\#2,
417 points=\#3%
418 }%
419 }

```

\levelleddisadvantage Adds a disadvantage to a character. NOTE: this only works in a character or lens environment.

```

420 \NewDocumentCommand{\levelleddisadvantage}{mmO{nil}}{%
421 \gurps@char@insertattr{%
422 name={\#1},%
423 type=disadvantage,%
424 level=\#2,
425 points=\#3%
426 }%
427 }

```

\gurps@char@SpellOrSkill Macro for skills or spells (since functionally they are the same in *GURPS*). The [*points*] can be a specifier e.g. DX/Easy or points e.g. 48.

```

428 \NewDocumentCommand{\gurps@char@SpellOrSkill}{mm}{
429 \NewDocumentCommand{\#1}{mom}{%
430 \IfValueTF{\#2}{%
431 \makeatletter%
432 \luadirect{
433 tmpone = \luastring{\unexpanded{\#1}}
434 tmptwo = \luastring{\unexpanded{\#2}}
435 tmpthree = \luastring{\unexpanded{\#3}}
436 }%
437 \IfInteger{\#2}{%

```

```

438     \luadirect{
439         tex.sprint(
440             [[\gurps@char@insertattr{]]
441             .. [[name={]] .. tmpone .. [[],]]
442             .. [[type=#2,]]
443             .. [[points=##2,]]
444             .. [[level=]] .. tmpthree
445             .. "]"
446         )
447     }
448 }{%
449     \luadirect{
450         basedon = tmptwo:gsub('/.*', '')
451         difficulty = tmptwo:gsub('.*/', '')
452         tex.sprint(
453             [[\gurps@char@insertattr{]]
454             .. [[name={]] .. tmpone .. [[],]]
455             .. [[type=#2,]]
456             .. [[basedon=]] .. basedon .. ', '
457             .. [[difficulty=]] .. difficulty .. ', '
458             .. [[level=]] .. tmpthree
459             .. "]"
460         )
461     }%
462 }%
463 \makeatother%
464 }{%
465     \gurps@char@insertattr{%
466         name={##1},%
467         type=#2,%
468         level=##3
469     }%
470 }%
471 }
472 }

\skill
473 \gurps@char@SpellOrSkill{\skill}{skill}

\spell
474 \gurps@char@SpellOrSkill{\spell}{spell}

```

4.3.3 Stat block formatting

`\characteritle` As a title for the character stats block, we define `\characteritle`

```

475 \newcounter{characteritle} % to keep LaTeX happy
476 \newcommand{\characteritlemark}[1]{} % ditto
477 \newcommand\characteritle{%
478     \@startsection{characteritle}%
479     {100}%          level for secnumdepth and tocdepth
480     {\z@}%          indentation
481     {\topsep}%      space before
482     {\topsep}%      space below
483     {\raggedright\Large\bfseries}% format of the text
484 }

```

`\charactersection` For sections in the the character stats block, we define `\charactersection`

```

485 \newcounter{charactersection} % to keep LaTeX happy
486 \newcommand{\charactersectionmark}[1]{} % ditto
487 \newcommand\charactersection{%
488     \@startsection{charactersection}%
489     {101}%          level for secnumdepth and tocdepth
490     {\z@}%          indentation

```

```

491 {\topsep}% space before
492 {-1em}% space below
493 {\raggedright\bfseries}% format of the text
494 }
495 % \NewDocumentCommand{\charactersection}{m}{%
496 % \noindent\par\noindent\textbf{\large #1}\noindent%
497 % }

```

who's code is taken almost without change from <https://tex.stackexchange.com/questions/97565/flexible-vspace>

`charactertraitlist` We define the `charactertraitlist` to list the traits. By default, this is an `itemize*` from the `enumitem` package with some customisations

```

498 \NewDocumentEnvironment{charactertraitlist}{}
499 {\noindent\begin{itemize*}[itemjoin={{; }},label=,afterlabel={}]}
500 {\end{itemize*}}

```

`attacklist` We define the `attacktraitlist` to list the traits. By default, this is an `itemize*` from the `enumitem` package with some customisations

```

501 \NewDocumentEnvironment{attacklist}{}
502 {%
503 \noindent%
504 \begin{itemize}[%
505 % itemjoin={\},% TODO figure out why this causes a problem!
506 label={--},%
507 wide,
508 nosep
509 % afterlabel={},% Default of ~ is good
510 ]%
511 }\end{itemize}}

```

`\gurps@char@print@attr` It's like this: `\gurps@char@print@attr[<level/diceexpr>]{<name>}[<points>]`. This macro is usually called by the lua function `attr_to_tex` (which returns a string)

```

512 \NewDocumentCommand{\gurps@char@print@attr}{omo}{%
513 \IfValueTF{#3}{%
514 \def\tmp@points{[#3]}%
515 }{%
516 \def\tmp@points{}%
517 }%
518 \IfValueTF{#1}{%
519 #2~#1\tmp@points%
520 }{%
521 #2\tmp@points%
522 }%
523 }

```

`\gurps@char@attacks`

```

524 \NewDocumentCommand{\gurps@char@attacks}{}{%
525 \luadirect{attacklist(_GCHARACTERKEY)}
526 }

```

`\gurps@char@TraitListMaker`

```

527 \NewDocumentCommand{\gurps@char@TraitListMaker}{mmO{nil}}{
528 \NewDocumentCommand{#1}{}{%
529 \makeatletter%
530 \luadirect{traitlistmaker(#2, _GCHARACTERKEY, #3)}%
531 \makeatother%
532 }%
533 }

```

`\gurps@char@totalpoints`

```

534 \NewDocumentCommand{\gurps@char@totalpoints}{O{}}{%
535 \luadirect{tex.sprint(sum_points(luastring{\unexpanded{#1}}))}~pt%
536 }

```

```

\gurps@char@traits
537 \gurps@char@TraitListMaker{\gurps@char@traits}{is_trait}

\gurps@char@advantages
538 \gurps@char@TraitListMaker{\gurps@char@advantages}{is_advantage}

\gurps@char@disadvantages
539 \gurps@char@TraitListMaker{\gurps@char@disadvantages}{is_disadvantage}

\gurps@char@skills
540 \gurps@char@TraitListMaker{\gurps@char@skills}{is_skill}

\gurps@char@spells
541 \gurps@char@TraitListMaker{\gurps@char@spells}{is_spell}

\gurps@char@basicattributes
542 \gurps@char@TraitListMaker{\gurps@char@basicattributes}{is_basic_attribute}[basic_attributes_sorter]

\gurps@char@secondarycharacteristics
543 \gurps@char@TraitListMaker{\gurps@char@secondarycharacteristics}{is_secondary_characteris-
tic}[secondary_characteristics_sorter]

\gurps@char@properties
544 \gurps@char@TraitListMaker{\gurps@char@properties}{is_property}

\gurps@char@checkandfixattrsandpoints
545 \NewDocumentCommand{\gurps@char@checkandfixattrsandpoints}{\}{\%
546 \luadirect{check_and_fix_attrs_and_points(_GCHARACTERKEY)}\%
547 }

548 \endinput
549 \endpackage

```

Change History

v0.1		\gurpsbook command added. 1
General: First public release 1	v0.3	
v0.1.1		General: Completely changed underlying
General: Minor changes to CTAN upload 1		character lua structure. Removed lens
v0.2		environment (to be put in at a later date).
General: Better character environment;		TODO 1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	A	C
\@currentvir .. 89, 92, 338, 341	\advantage 397	character (environment) 2
\@myenvname .. 88, 89, 337, 338	attacklist (environment) . 501	character, character* (en- vironment) 58
\@startsection 478, 488		\charactersection
\\ 496, 505	B 158, 160, 162,
	\basicmove 304	165, 169, 173, 177, 181, 485
	\basicspeed 296	\charactersectionmark .. 486
	\begin 499, 504	\characterstitle 475
_ 181	\bfseries 483, 493	

\characteritlemark . . .	476	297, 305, 398, 405,	\makeatother	463, 531
charactertraitlist (envi- ronment)	498	413, 421, 440, 453, 465	\meleeattack	391
D				
\def	73, 76, 78, 80, 82, 84, 86, 88, 313, 315, 317, 318, 320, 322, 324, 337, 514, 516	\gurps@char@insertattr@basedon	N	
\define@key	72, 75, 77, 79, 81, 83, 85, 312, 314, 316, 318, 319, 321, 323	\gurps@char@insertattr@diceexpr	\newcounter	475, 485
\dice	3, 24	\gurps@char@insertattr@difficulty	\newtoggle	
\disadvantage	404	\gurps@char@insertattr@level	\noindent	496, 499, 503
\DR	267	\gurps@char@insertattr@name	P	
\DX	204	\gurps@char@insertattr@points	\PackageError . . .	92, 192, 341
E				
\else	91, 340	\gurps@char@insertattr@type	\par	496
\end	500, 511	\gurps@char@print@attack	\PassOptionsToPackage	8
\endinput	548	\gurps@char@print@attr .	\Per	236
environments:		\gurps@char@properties .	\printcharacter	63, 155
attacklist	501	R	
character	2	\gurps@char@secondarycharacteristics	\raggedright	483, 493
character, character* .	58	\rangedattack	382
charactertraitlist .	498	\gurps@char@skills .	\RequirePackage	2,
F				
\fbox	389	\gurps@char@spellorskill	\ResetCharacterKey .	64, 69, 184
\fi	93, 342	\gurps@char@spells . .	S	
\FP	252	\gurps@char@totalpoints	\SetCharacterKey .	59, 66, 70, 156
G				
\gurps . . .	3, 17, 29, 40, 44, 50	\gurps@char@traits	\setkeys	95, 105, 327, 345, 355, 383, 384, 392, 393
\gurps@char@advantages .	166, 538	\gurps@porpp	\show	357, 358,
\gurps@char@attacks .	182, 524	\gurps@book	359, 360, 361, 362, 363
\gurps@char@basicattributes	159, 542	H		
\gurps@char@checkandfixattrsandpoints	62, 545	\HP	\SJGames	31, 40, 41, 44, 45, 50, 51, 52, 56
\gurps@char@checkandfix toggles	152, 157	\href	\SJGamesOnlinePolicyDisclaimer	3, 38
\gurps@char@disadvantages	170, 539	\HT	\SJGamesOnlinePolicyGameAid	3, 49
\gurps@char@getattrlevel .	186	I		
\gurps@char@insertattack	312, 386, 395	\IfBooleanTF . . .	\SJGamesOnlinePolicyNotice	3, 43
\gurps@char@insertattack@damage	317, 360, 368	\IfInteger	\skill	473
\gurps@char@insertattack@level	318, 359, 367	\iftoggle	\SM	260
\gurps@char@insertattack@name	313, 357, 365	\IfValueT	\spell	474
\gurps@char@insertattack@notes	324, 363, 371	\IfValueTF	\ST	196
\gurps@char@insertattack@range	322, 362, 370	\ifx	\strut	181
\gurps@char@insertattack@reach	320, 361, 369	\IQ	\sw	289
\gurps@char@insertattack@type	315, 358, 366	L		
\gurps@char@insertattr .	72, 197, 205, 213, 221, 229, 237, 245, 253, 261, 269, 275, 283, 290,	\Large	T	
		\large	\textbf	20, 22, 29, 389, 496
		\levelledadvantage	\textit	20, 22, 29, 389
		\levelleddisadvantage . .	\thr	282
		\linebreak	\tmp@points	514, 516, 519, 521
		\lua direct	\topsep	481, 482, 491
		U	
		\luastring	\unexpanded	
		67, 73, 76, 84, 86, 313, 315, 317, 320, 322, 324, 433, 434, 435, 535
		\makeatletter	W	
		431, 529	\Will	244
			X	
			\xspace	20, 22, 34, 36
			Z	
			\z@	480, 490