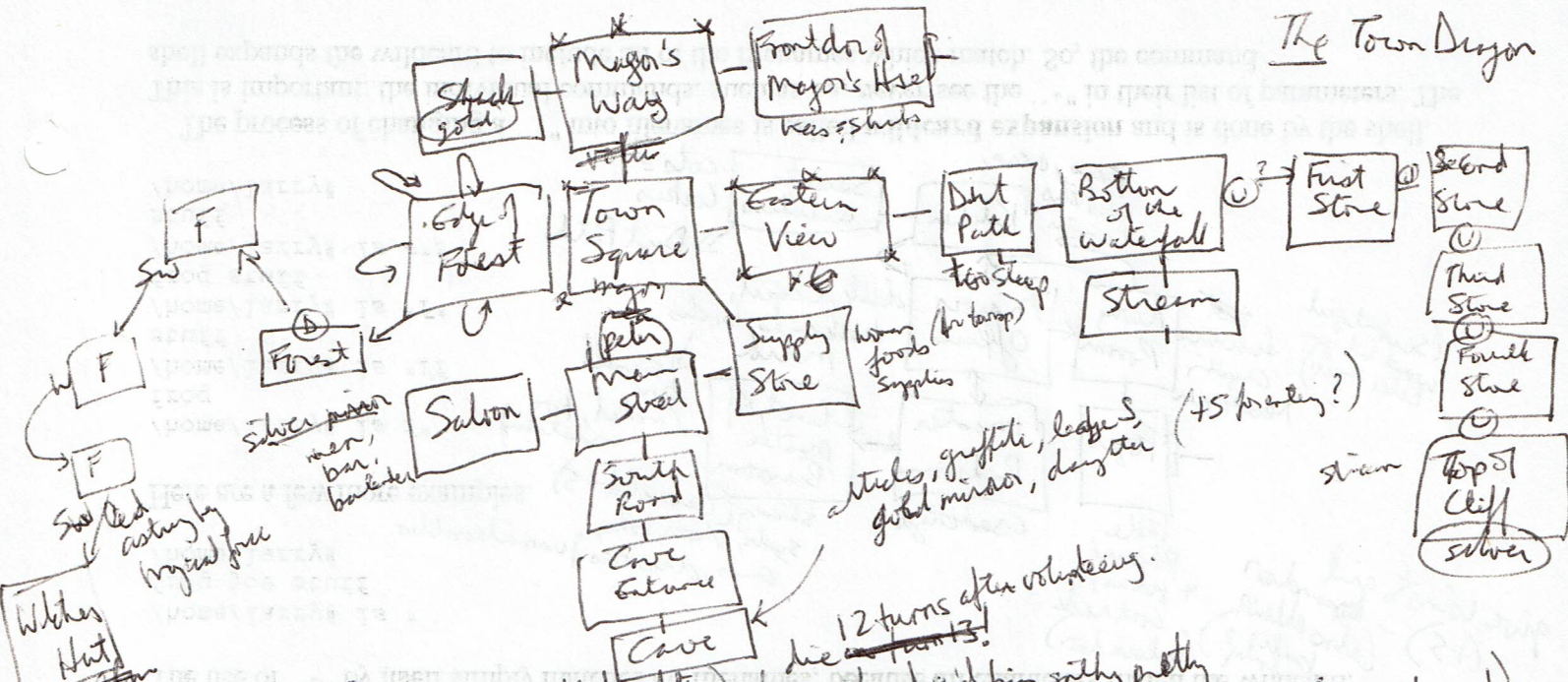


The Town Dragon



Witches Hut  
 Witches (short brown hair) slim +5 ask woman abt witch (in our woods) ask woman abt dragon (likely from mt) abt mayor (father) abt daughter (sweetheart) abt town (witch + dragon!)  
 volunteer [+5] (ask mayor abt daughter) dragon hoped g'd big sithy pretty  
 ask bartender abt saloon, dr, mayor (low drunk), dragon, daughter, mountains (to NW; major + hidden valley)  
 witch (lives in forest; article in town paper), town, paper (mayor took our copy).  
 take gold [+5] @ Shed give gold to dragon (buyt medicine) die at turn 23!  
 enter mayor's gold mine → Saloon. (cant go the reverse direction!)  
 knock on door @ Mayor's [+5; servant] ask servant abt paper, daughter (Irene, was gonna marry you) dragon, house, town, witch (dont believe to paper; she's sweet + lovely), saloon, forest, door, cave (which? I heard there's one behind the waterfall), waterfall  
 when with Peter, dragon doesnt kill y (or him?) ask daughter abt mayor (with paper passed away, did not stage), servant (Sally; we share secrets), peter, mirror (saloon has problem i know secrets), waterfall, saloon, town, forest (need map), witch (knows enchantress who lives i husband + dog; Peter enter at turn 77; dressed back to T.S.)  
 y looks almost but not quite like him.  
 Gary volunteers // ask woman abt waterfall, cave, forest, sally (tramp!) // ask bartender abt Sally (Niece!) (oh)  
 Thomas is 3rd volunteer // give mirror to mayor (outside Saloon → get copper wire)  
 ask mayor abt town, mirror, Irene = keyfles, volunteer, dragon, me, witch (by any)  
 (not: Saloon), Sally, forest, mirror, waterfall, stone, women, paper





Next: 3.8 UNIX Plumbing Up: 3 Linux Tutorial Previous: 3.6 Types of shells

### 3.7 Wildcards

A key feature of most Unix shells is the ability to reference more than one filename using special characters. These so-called **wildcards** allow you to refer to, say, all filenames which contain the character "n".

The wildcard "\*" refers to any character or string of characters in a filename. For example, when you use the character "\*" in a filename, the shell replaces it with all possible substitutions from filenames in the directory which you're referencing.

Here's a quick example. Let's suppose that Larry has the files frog, joe, and stuff in his current directory.

```
/home/larry# ls
frog joe stuff
```

```
/home/larry# ls *o*
frog joe
```

```
/home/larry# ls *o*
frog joe stuff
```

```
/home/larry# ls *o*
frog joe stuff
```

```
/home/larry# ls *
```

Here are a few more examples.

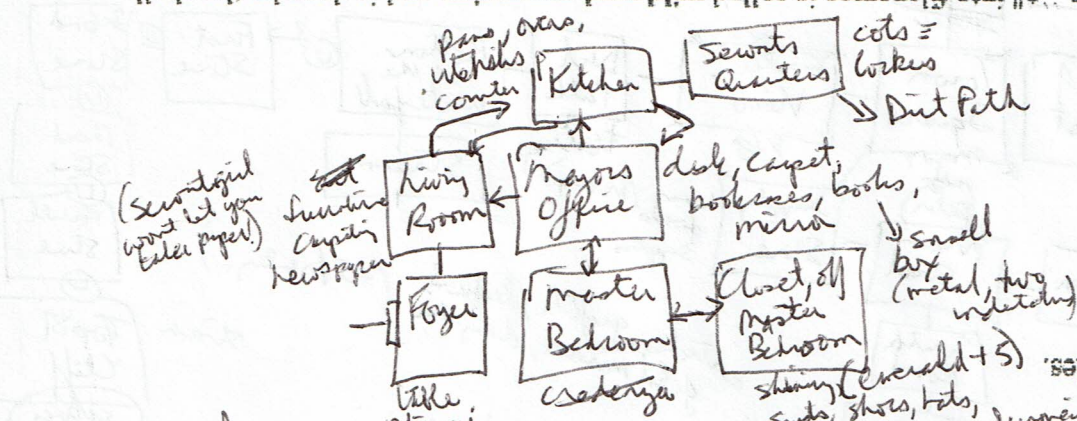
```
/home/larry# ls *
frog
stuff
/home/larry# ls *f*
frog stuff
/home/larry# ls *f*
frog stuff
/home/larry# ls *f*
frog stuff
/home/larry# ls *f*
frog stuff
```

look up mayor,

To access all files with the letter "o" in the filename, we can use the command `ls *o*`. Shortly after mayor's daughter (08-Aug-1997, a little bit later) was born, a little bit later.

ask sally abt paper (just a grade, but leave it behind)  
- ask sally abt paper (just a grade, but leave it behind)  
mayor's fault dupon has Irene.  
Kissgirl @ Sewants Quarters  
give kissgirl (45) - (also allows)

The use of "\*" by itself simply matches all filenames, because all characters match the wildcard. As you can see, the use of the "\*" wildcard was replaced with all substitutions which matched the wildcard from filenames in the current directory.



The process of changing a "\*" into filenames is called **wildcard expansion** and is done by the shell. This is important: the individual commands, such as `ls`, never see the "\*" in their list of parameters. The shell expands the wildcard to include all of the filenames which match. So, the command