

# RPG relationship mapping with Graphviz

FRIDAY, JANUARY 7TH, 2022

Interrupting my train of  
thought  
Lines of longitude and  
latitude  
Define and refine my altitude

---

Wire, Map Ref. 41řN 93řW

## Introduction

I recently started running a Burning Wheel<sup>1</sup> game after about two decades away from GMing role-playing games. We are calling our game *Burning Locusts*.

<sup>1</sup> Burning Wheel HQ (<https://burningwheel.com>)

The Situation revolves around a spooky mercenary company who returns every 17 years to wreak havoc (there is a joke in there if you look for it). It is set in a city called Scintallante that is modelled after Italian city states during the Renaissance period. The city resembles Milan geographically, with an ancient dwarven stronghold in the center called Hartheld. In reality it is two cities: the dwarven city of Hartheld and the human city of Scintallante.

There are three player characters in the game:

- Ansidora: A dwarven runecaster
- Antonius di Mari: A human former mercenary captain turned thinker
- Frederico Pascolini: An ambitious human soldier, spy, thief

For more background, one of the players has written excellent session notes. They can be found [here](#)<sup>2</sup>, [here](#)<sup>3</sup>, and [here](#)<sup>4</sup>.

As detailed in the *Character creation* post, the players created characters with a number of interesting relationships and Beliefs right off the bat. To make the most out of this, I wanted to find some way to visualize the connections and observe how they change over time.

<sup>2</sup> Burning Locusts: Character Creation (<https://takeonrules.com/2021/12/10/burning-locusts-character-creation/>)

<sup>3</sup> Burning Locusts: Session 1 (<https://takeonrules.com/2021/12/16/burning-locusts-session-1/>)

<sup>4</sup> Burning Locusts: Session 2 (<https://takeonrules.com/2021/12/30/burning-locusts-session-2/>)

## Mapping the relationships

I am not a visual thinker so I find that I benefit from forcing myself into that mindset. I also wanted to adopt better organizational

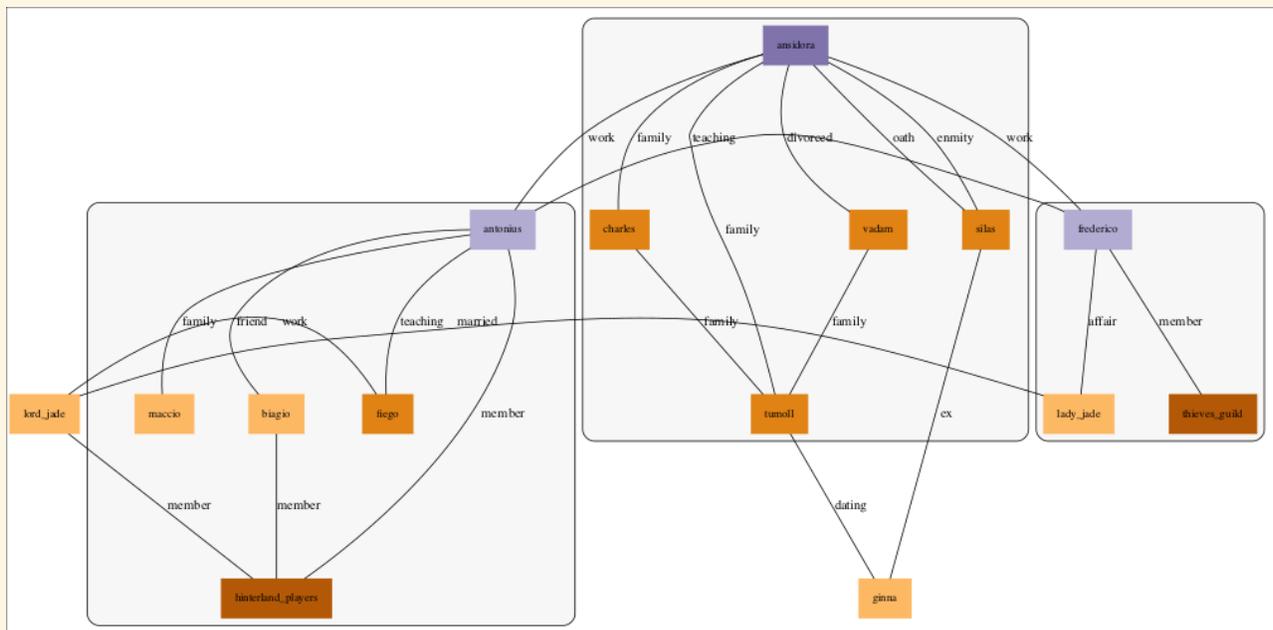
habits as a GM. My previous organizational habits from 20 years ago involved writing things down on napkins, so there is certainly room for improvement. So I created a relationship map<sup>5</sup> to better understand all of the interesting things happening in our game.

To create the map, I evaluated a few options but I didn't want to use a graphical or a web-based application to do it. I also wanted something relatively easy to use. I settled on Graphviz<sup>6</sup>. Graphviz includes a programming language called *DOT* that can be used to describe a graph in plaintext using any text editor and turn it into an image.

<sup>5</sup> AFAIK, Sorcerer by Ron Edwards was the first game to introduce relationship maps as a tool for role-playing games in the *The Sorcerers Soul* supplement ([https://en.wikipedia.org/wiki/Sorcerer\\_\(role-playing\\_game\)](https://en.wikipedia.org/wiki/Sorcerer_(role-playing_game)))

<sup>6</sup> Graphviz, open-source graph visualization software (<https://graphviz.org/>)

### The graph



### Concepts

A minimal amount of background knowledge is needed to start using the basic features of Graphviz. The lines on the graph are called *edges*. The lines connect *nodes* or *vertices*. That's all we need to know to begin.

The Graphviz *DOT* language provides a simple shorthand for expressing connections:

```
1 graph G {
2   ansidora -- charles [label = "family"]
3 }
```

I have chosen to use an undirected graph for our relationship map because it is easier, ie: the edges do not point in any direction, they are just connections between nodes. In the code example above, this choice is denoted by the keyword `graph`. A directed graph would use the keyword `digraph` instead.

If this were directed graph, we would use `->` instead of  to express these connections. The `label` tag should be obvious but to further elaborate we are labelling this connection as a *family* connection. *Family* is not a special concept in Graphviz, it is just an arbitrary label for our relationship map.

Please refer to the documentation<sup>7</sup> for information on how to use the language.

<sup>7</sup> DOT Language documentation (<https://graphviz.org/doc/info/lang.html>)

### Subgraphs

When I made the graph, I decided to put each player character in a subgraph to represent who is inside or outside of that character's circle. This also had the benefit of showing that the marriage between Lord Jade and Lady Jade crossed different character subgraphs via the affair between Lady Jade and Frederico.

### Coloring

I wanted colors to represent the following:

- Human player characters: `color=8, #B2ABD2`
- Dwarven player characters: `color=9, #8073AC`
- Human non-player characters: `color=4, #FBD863`
- Dwarven non-player characters: `color=3, #E08214`
- Factions: `color=2, #B35806`

I chose a set of Brewer colors<sup>8</sup> to have a range of colors that will stand out from each other. The particular set I chose is *puor11*.

<sup>8</sup> Graphviz Brewer colorschemes (<https://graphviz.org/doc/info/colors.html#brewer>)

The following is a minimal example to demonstrate coloring and styling nodes in a graph:

```
1 graph G {
2   node [colorscheme=puor11 color=4 margin=0.2 fontsize=12 width=0.5 shape=box
3     style=filled];
4   lord_jade -- lady_jade [label = "married"]
5 }
```

There are the attributes from the code example:

- colorscheme: A pre-determined set of colors
- color: A color from this set
- margin: The margin around the node name
- fontsize: The size of the node name
- width: The width of the node
- shape: The shape of the node (ex: box, circle, etc.)
- style: The style of the node (ex: filled, rounded, etc.)

### *Generating the graph*

Graphviz can be installed on Windows, Mac OS X, and Linux. Instructions can be found on the website that is linked above.

Any text editor can be used to create a graph file in *DOT* and some code editors provide extensions to automatically preview the source code.

There are also a few online Graphviz editors such as this one<sup>9</sup>.

<sup>9</sup> GraphvizOnline (<https://dreampuf.github.io/GraphvizOnline/>)

This is the complete *dot* source code for the Burning Locusts relationship map (`burning-locusts-1.dot`):

```

1 graph G {
2   size = 10.5;
3   colorscheme=puor11;
4   node [colorscheme=puor11 color=4 margin=0.2 fontsize=12 width=0.5 shape=box
5     style=filled];
6   nodesep=0.8;
7   ranksep=1.9;
8
9   subgraph cluster_1 {
10    style=rounded;
11    bgcolor=6;
12    antonius [color=8]
13    antonius -- maccio [label = "family"]
14    antonius -- biagio [label = "friend"]
15    antonius -- fiego [label = "teaching"]
16    antonius -- hinterland_players [label = "member"]
17  }
18
19  hinterland_players [color=2]
20  biagio -- hinterland_players [label = "member"]
21  lord_jade -- hinterland_players [label = "member"]
22  lord_jade -- lady_jade [label = "married"]
23  fiego [color=3]
24  fiego -- lord_jade [label = "work"]
25  thieves_guild [color=2]
26
27  subgraph cluster_2 {
28    style=rounded;
29    bgcolor=6;
30    frederico [color=8]
31    frederico -- antonius [label = "teaching"]
32    frederico -- lady_jade [label = "affair"]
33    frederico -- thieves_guild [label = "member"]
34  }
35
36  subgraph cluster_3 {
37    style=rounded;
38    bgcolor=6;
39    ansidora [color=9]
40    ansidora -- charles [label = "family"]
41    ansidora -- tumoll [label = "family"]
42    ansidora -- vadam [label = "divorced"]
43    ansidora -- frederico [label = "work"]
44    ansidora -- antonius [label = "work"]
45    ansidora -- silas [label = "oath"]
46    ansidora -- silas [label = "enmity"]
47  }
48
49  vadam [color=3]
50  silas [color=3]
51  tumoll [color=3]
52  charles [color=3]
53  vadam -- tumoll [label = "family"]
54  charles -- tumoll [label = "family"]
55  tumoll -- ginna [label = "dating"]
56  ginna -- silas [label = "ex"]
57 }

```

To generate the relationship map, run the following command:

```
$ dot -Tsvg burning-locusts-1.dot -o burning-locust-1.svg
```

### *Other work*

PlantUML<sup>10</sup> can also be used to the same effect. Jeremy at the Take on Rules blog<sup>11</sup> has an example for another Burning Wheel game that I am playing in here<sup>12</sup>.

<sup>10</sup> PlantUML (<https://plantuml.com/>)

<sup>11</sup> Take on Rules blog (<https://takeonrules.com/>)

<sup>12</sup> Using PlantUML to Model RPG Relationship Maps (<https://takeonrules.com/2022/01/09/using-plant-uml-to-model-rpg-relationship-maps/>)