the gamedesigninitiative at cornell university

Lecture 14

Procedural Content Generation

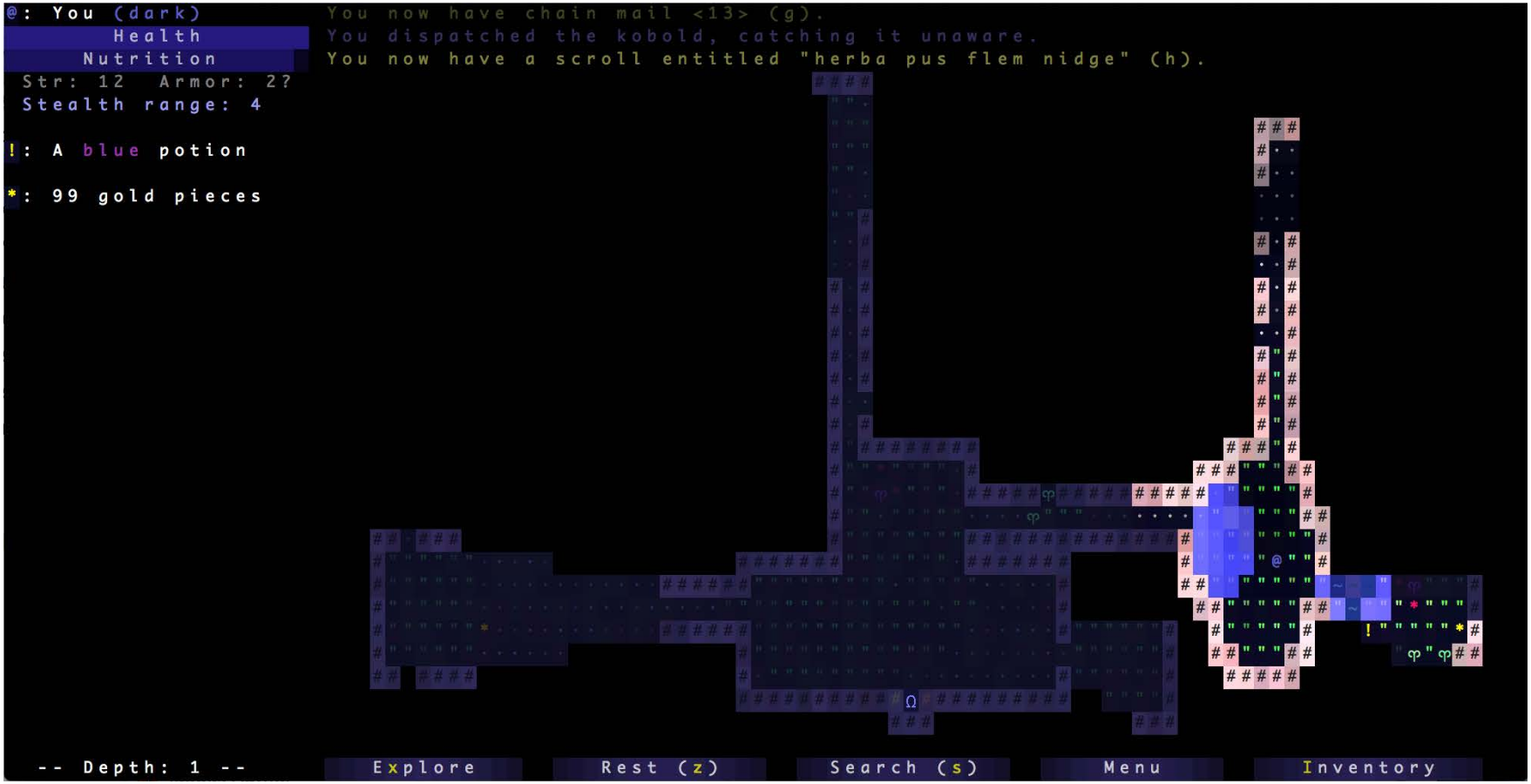
Lessons for Today

- Procedural content is **harder**, not easier
  - You must already know your *design patterns*
  - Controlling *difficulty* is a potential challenge
  - *Unwinnable levels* are also a challenge
- Many procedural approaches are **ad hoc**
  - Designed for specific games
  - Limited adaptability to other games
- Procedural generation is a **stretch goal**

Important Caveat

- This lecture is up to date as of **GDC 2025**
  - I attended the generative AI sessions in 2024
  - Have tried to keep up with the state-of-the-art
- But generative AI is rapidly advancing
  - New tools are produced every day
  - We know game studios have internal tech
- Have tried to be as current as I possibly can
  - Lecture includes news reports from **last week**
  - May not know more until a game uses the tech

In the Beginning, There Was *Rogue*



In the Beginning, There Was *Rogue*

Roguelike Genre

- Classic RPG style
- Procedural dungeons
- **Permadeath**

Modern Roguelikes: *Spelunky*

Modern Roguelikes: *FTL*

Modern Roguelikes: *FTL*

A Brief History of Roguelikes

- Precursors (1978)
  - *Beneath Apple Manor*
  - *Dungeon* (unfamous one)
  - Like *Rogue*, but less famous
  - Limited content generation
- *Rogue* (1980)
  - Multiplatform launch
- Immediate Copycats
  - *Hack* (‘82), *NetHack* (‘87)
  - *Moria* (‘83), *Angband* (‘90)
  - All very close in playstyle
  - Open source development
  - Middle Earth themed
- *Island of Kesmai* (1985)
  - *Legends of Kesmai* (1996)
  - Massively (~80) multiplayer
  - But content less procedural
- The Modern Revival
  - Relaxing RPG requirement

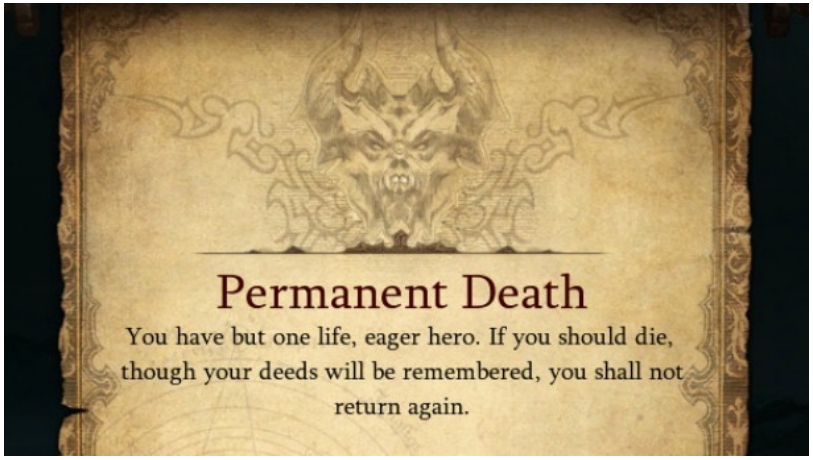
Changing Perspectives on Permadeath

Advantages

- Greater challenge
  - Used as a badge of honor
- Higher emotional stakes
  - Easy to instill fear & horror

Disadvantages

- Greater discouragement
  - Seen as a personal failure
- Missed game content
  - Cannot progress in story



Changing Perspectives on Permadeath

Advantages

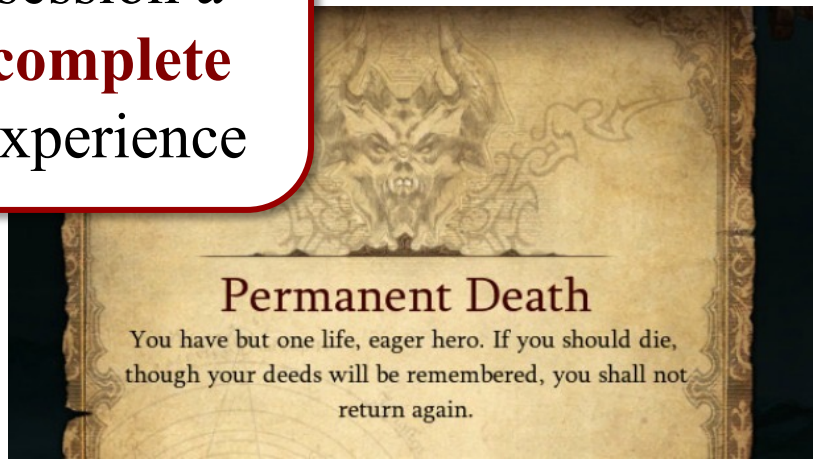
- Greater ch
  - Used as a
- Higher em
  - Easy to i

Disadvantages

- Greater discouragement
  - Seen as a personal failure
- Missed game content
  - Cannot progress in story

Make dying expected & **inevitable**

Make each session a **complete** experience



Changing Perspectives on Permadeath

Advantages

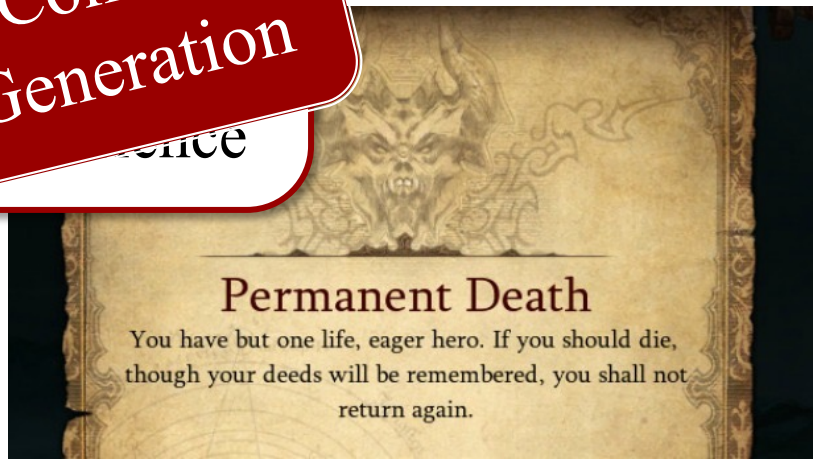
- Greater ch
  - Used as a
- Higher em
  - Easy to i

Make dying expected & **inevitable**

Content Generation

Disadvantages

- Greater discouragement
  - Seen as a personal failure
- Missed game content
  - Cannot progress in story



Aside: Rogue-lites

- Modern roguelikes do not completely reset
  - Some state carries over between play session
  - Way to reward player for simply trying
  - Call this variation a **Rogue-lite**
- Creates a sense of **meta-progression**
  - There is now the ability to "finish the game"
  - But there still replayability after the end
- **Examples**
  - Unlockable cards/items in *Slay the Spire*
  - Storyline and dialog in *Hades*

Issues with Roguelikes/lites

- Design is often **horizontal**
  - Many verbs, game elements
  - Little coupled behavior
- Each play is a **slice**
  - Access to limited elements
  - Work with what you get
- "Expensive" to create
  - Requires a lot of content
  - But historically just text
- Difficult to balance

WEAPON (Table 1)					
Dagger	COST	WGT	PROB	MATL	APPEARANCE
orcish dagger	$4	10	12	IRON	crude dagger
dagger	4	10	30	IRON	--
silver dagger	40	12	3	SILV	--
athame	4	10	0	IRON	--
elven dagger	4	10	10	WOOD	runed dagger
Knife	COST	WGT	PROB	MATL	APPEARANCE
worm tooth	2	20	0	NONE	--
knife (shito)	4	5	20	IRON	--
stiletto	4	5	5	IRON	--
scalpel	6	5	0	METL	--
crysknife	100	20	0	MINL	--
Axe	COST	WGT	PROB	MATL	APPEARANCE
axe	8	60	40	IRON	--
battle-axe	40	120*	10	IRON	double-headed axe
Pick-axe	COST	WGT	PROB	MATL	APPEARANCE
pick-axe	50	100	tool	IRON	--
dwarvish mattock	50	120*	13	IRON	broad pick
Short sword	COST	WGT	PROB	MATL	APPEARANCE
orcish short sword	10	30	3	IRON	crude short sword

Issues with Roguelikes/lites

- Design is often **horizontal**
  - Many verbs, game elements
  - Little coupled behavior
- Each play is a **slice**
  - A
  - W
- "Expensive" to create
  - Requires a lot of content
  - But historically just text
- Difficult to balance

Procedural Content for Modern Games?

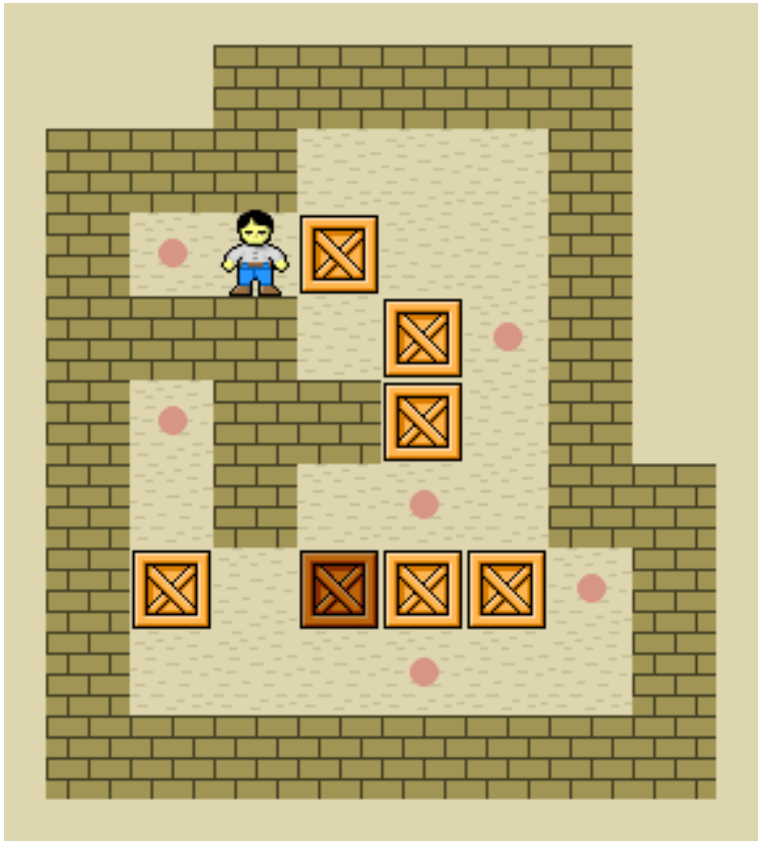
WEAPON (Table 1)					
Dagger	COST	WGT	PROB	MATL	APPEARANCE
orcish dagger	$4	10	12	IRON	crude dagger
dagger	4	10	30	IRON	--
silver dagger	40	12	3	SILV	--
athame	4			ON	--
					runed dagger
					APPEARANCE
					--
					--
			5	IRON	--
scalpel	6	5	0	METL	--
crysknife	100	20	0	MINL	--
Axe	COST	WGT	PROB	MATL	APPEARANCE
axe	8	60	40	IRON	--
battle-axe	40	120*	10	IRON	double-headed axe
Pick-axe	COST	WGT	PROB	MATL	APPEARANCE
pick-axe	50	100	tool	IRON	--
dwarvish mattock	50	120*	13	IRON	broad pick
Short sword	COST	WGT	PROB	MATL	APPEARANCE
orcish short sword	10	30	3	IRON	crude short sword

Can't We Use Generative AI?

- This is an application that designers **want**
  - Create custom levels for your game
  - Train an LLM on your designed levels
  - Use that to crank out even more levels
- But they **struggle** with this application
  - Related to the LLM counting problem
  - **E.g.** How many R's are in "strawberry"?
- Even if we solve it, level design is not obsolete
  - Still need good training data/example levels
  - What makes a good level is unique to your game

Case Study: *Sokoban*

Graphical Interface

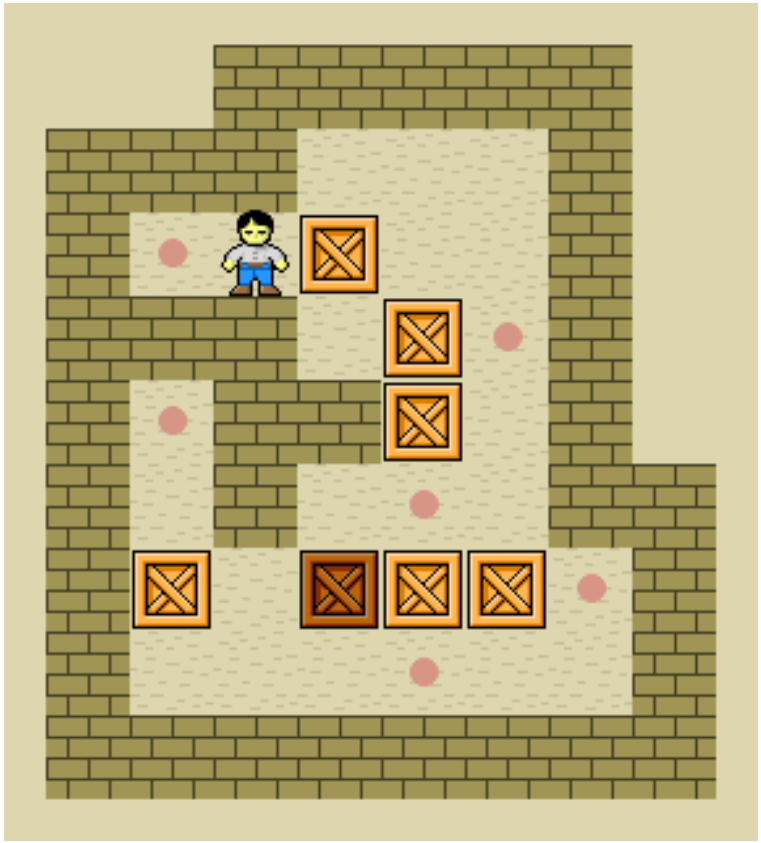


Level Representation

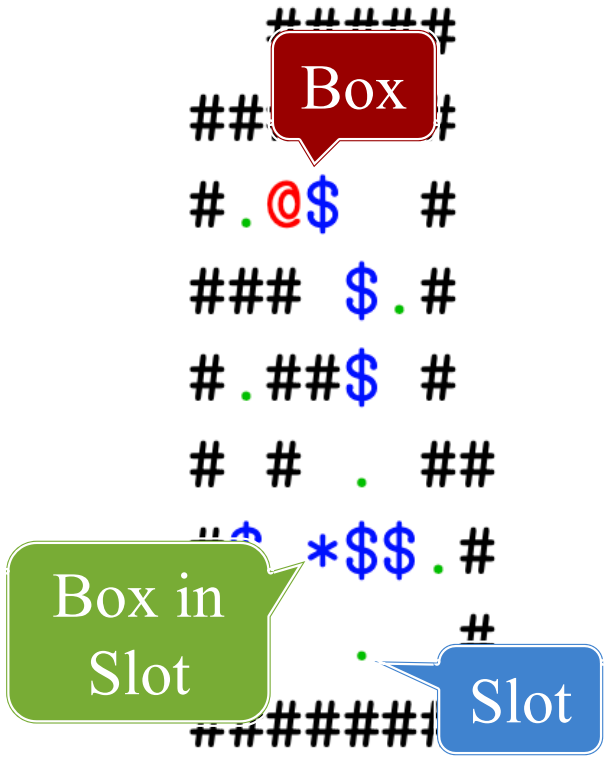
```
    #####
  ###   #
  #.@$  #
  ### $.#
  #.##$ #
  # #  . ##
  #$ *$$.#
  #   .  #
  ########
```

Case Study: *Sokoban*

Graphical Interface



Level Representation



Case Study: *Sokoban*

Graphical Interface

Level Representation

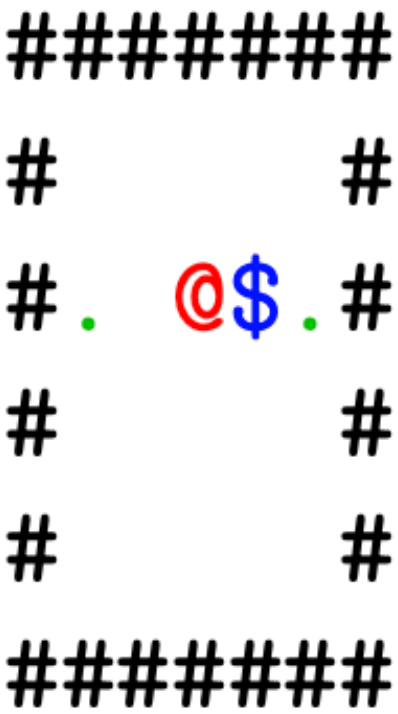
```
  #####
###   #
#.@$  #
### $.#
#.##$ #
# #  . ##
#$ *$$.#
#   .  #
########
```

The Solvability Problem

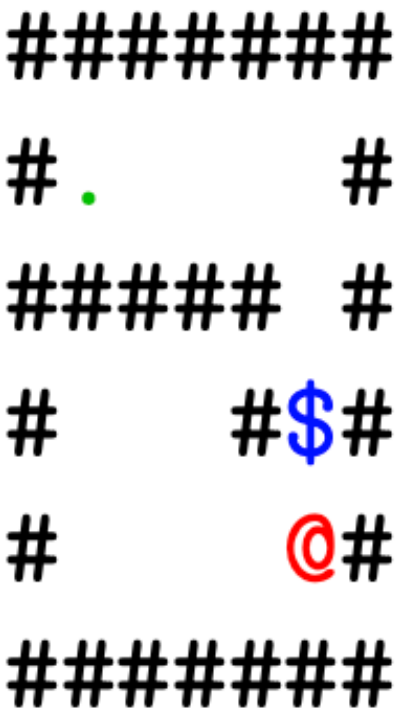
- Levels need to be **solvable**
  - Must be possible to reach the objective
  - Unsolvable levels are treated as bugs
- PCG does not always guarantee solvability
  - Possible to generate map with no route to exit
  - Possible to generate enemy too powerful to beat
- True PCG must include **constraints**
  - Rules to limit the content generated
  - Often *algorithmic* or mathematical in nature

Sokoban Constraints

Parity



Reachability



The Unfortunate Result

- Study by Julian Togelius et al (NYU Poly)
  - Generic Chat-GPT can recognize Sokoban text
  - But it **cannot generate a solvable level**
  - Vast majority of the time, even parity fails
- Can train on GPT with existing levels
  - A few **hundred levels** seems(?) to address parity
  - Reachability still an issue for larger levels
  - No way to tune difficulty/solution length

The Unfortunate Result

- Study by Julian Togelius et al (NYU Poly)
  - Generic Chat-GPT can recognize Sokoban text
  - But it **cannot generate a solvable level**
  - Vast
- Can tr
  - A fe parity
  - Reachability still an issue for larger levels
  - No way to tune difficulty/solution length

LLMs will solve **programming** long before they replace this.

Verdict: Only for “Cosmetic” Content

Verdict: Only for “Cosmetic” Content

Ensuring Solvability

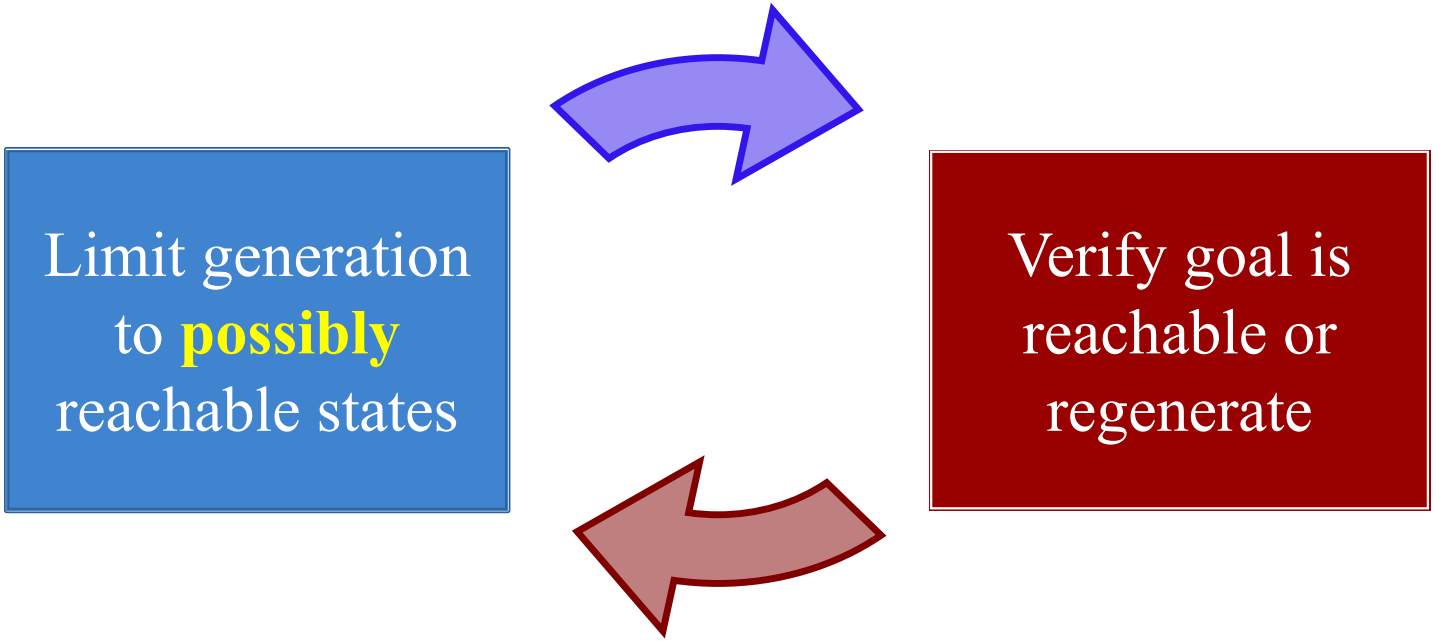
Two Options:

Limit generation to reachable game states

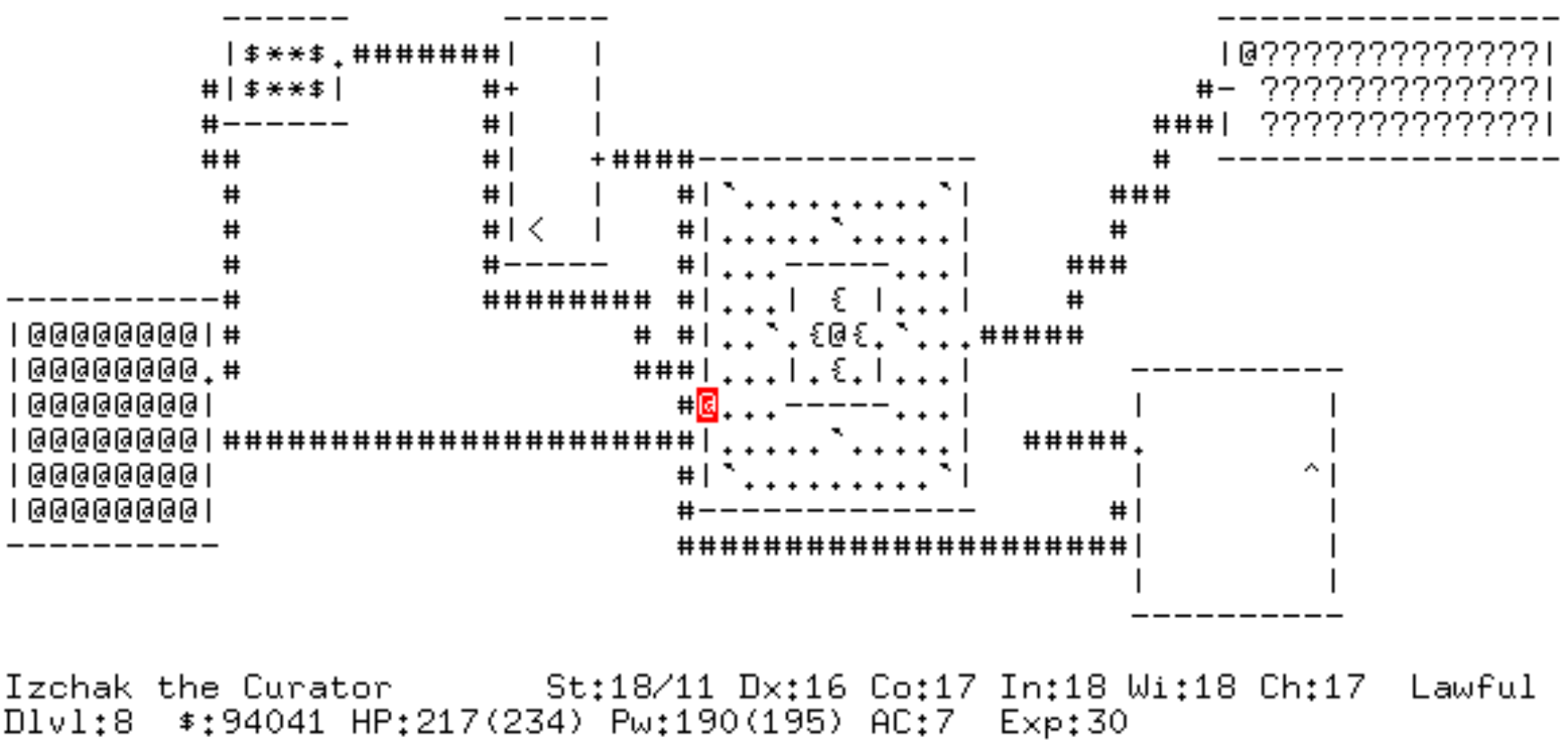
Verify goal is reachable or regenerate

Ensuring Solvability

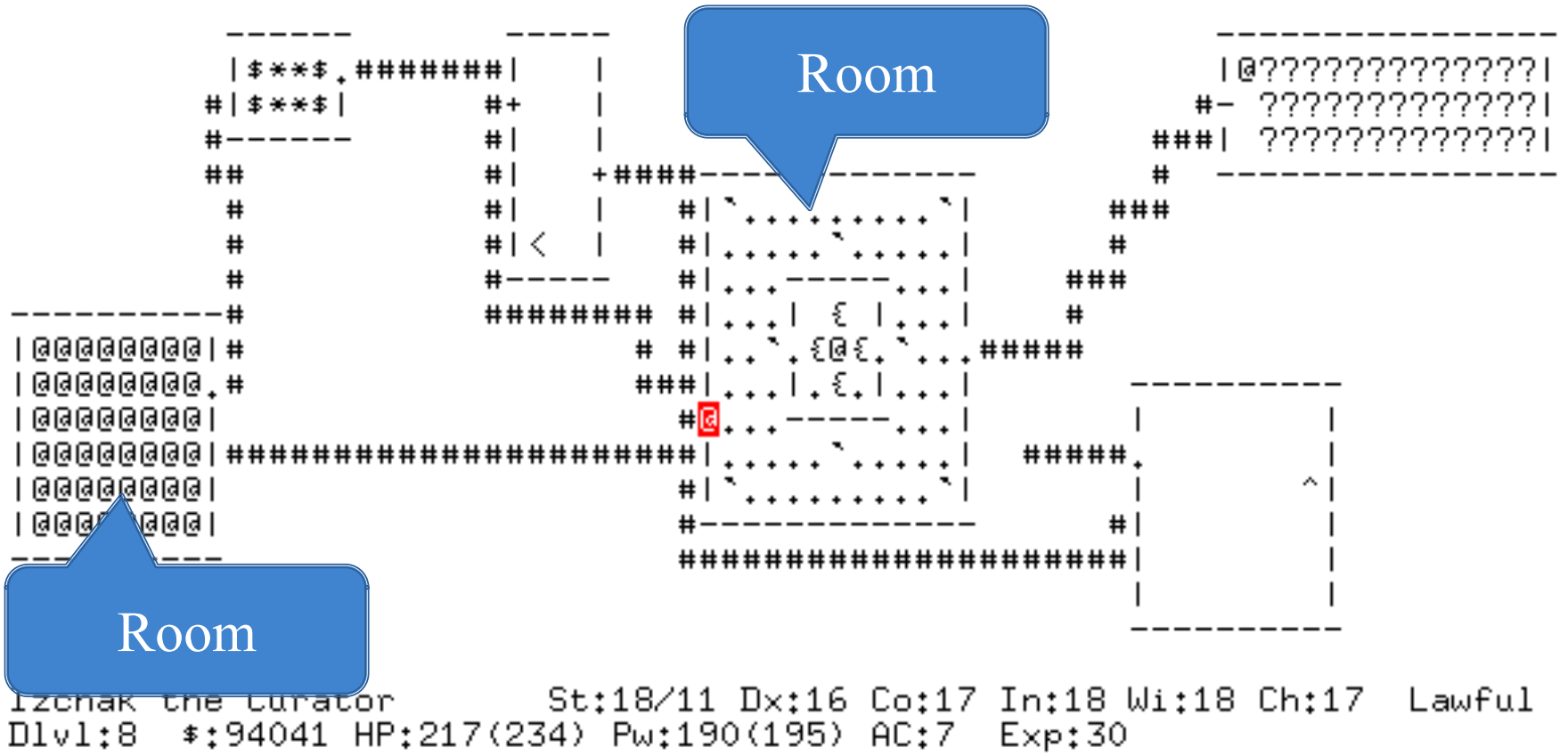
Two Options:



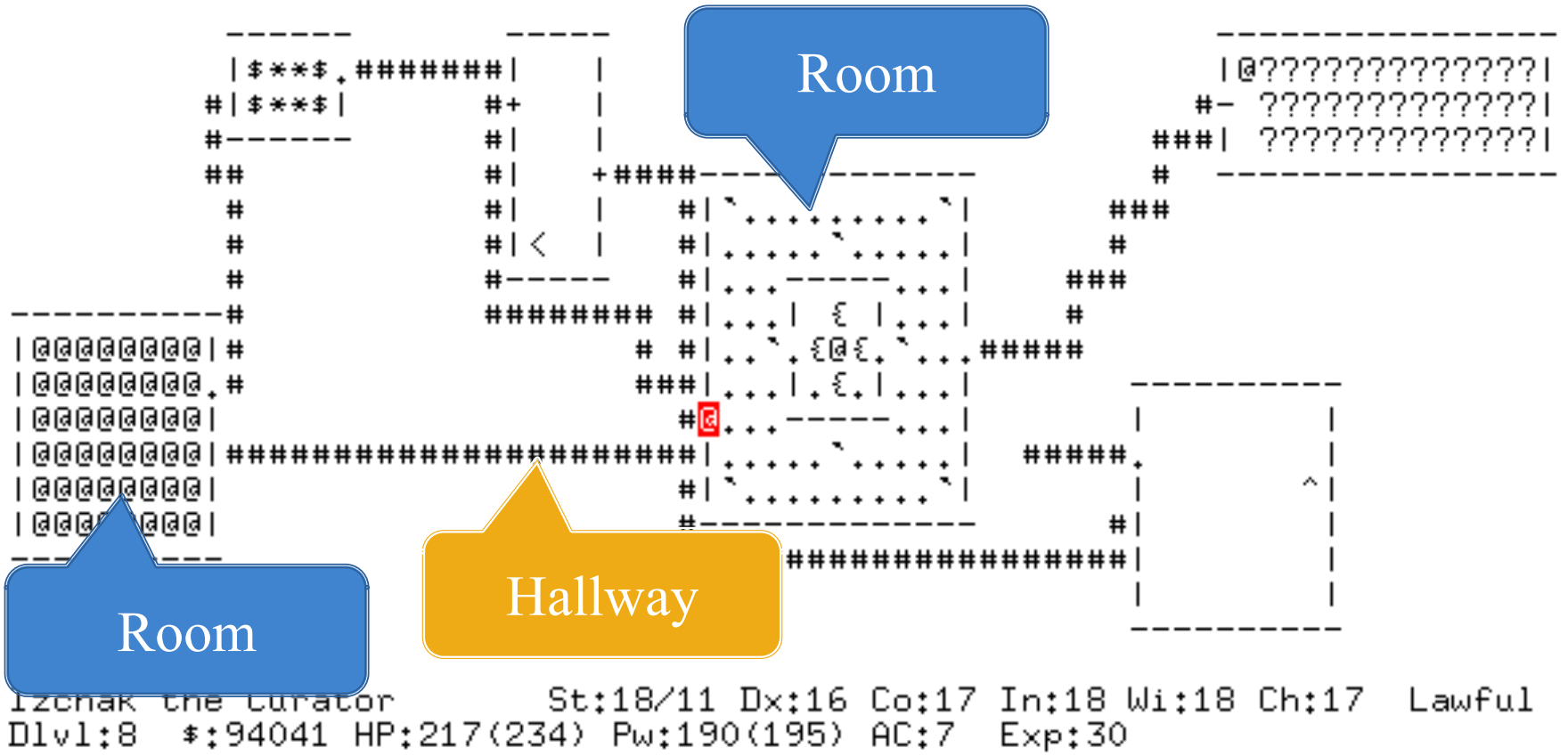
Example: NetHack



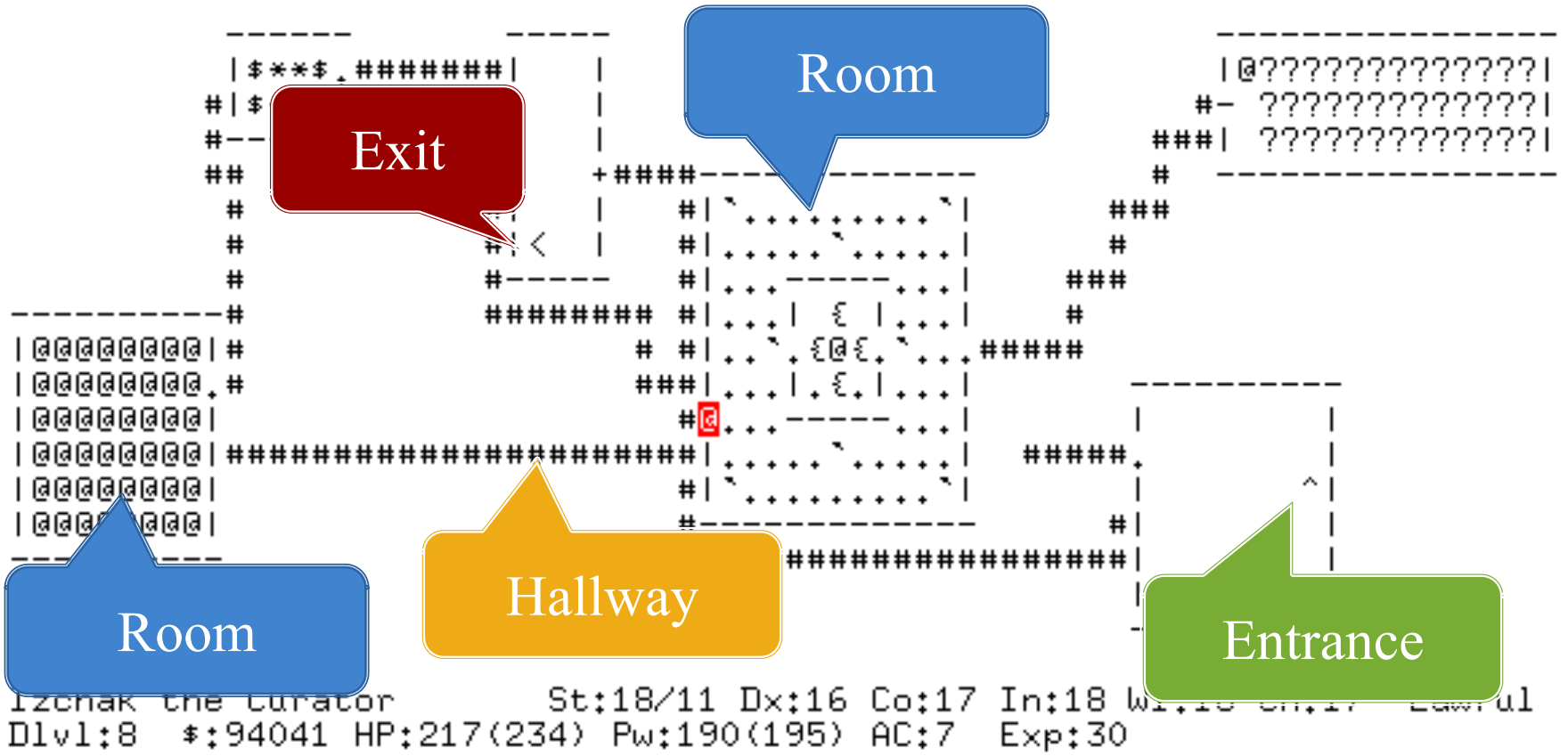
Example: NetHack



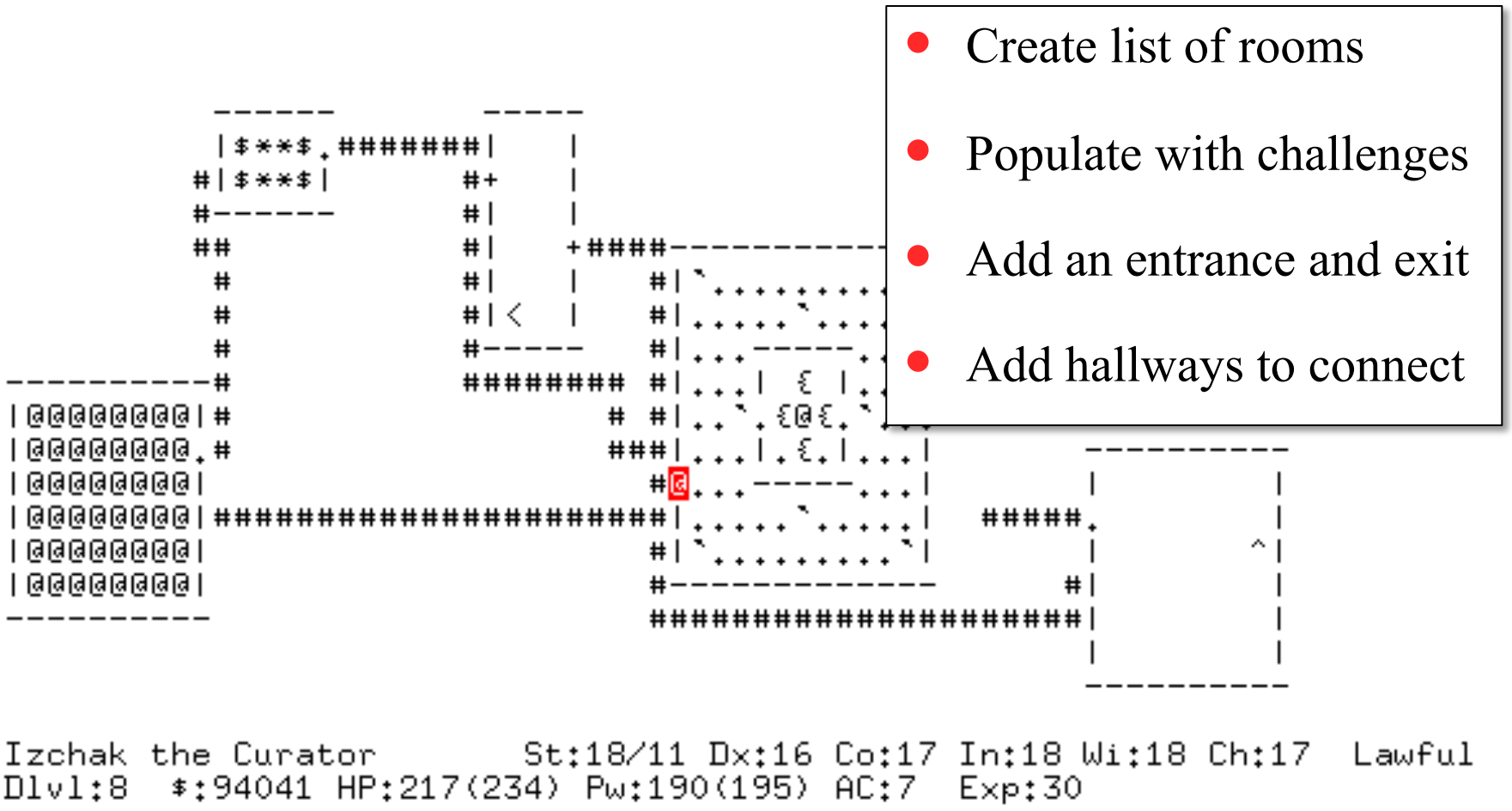
Example: NetHack



Example: NetHack

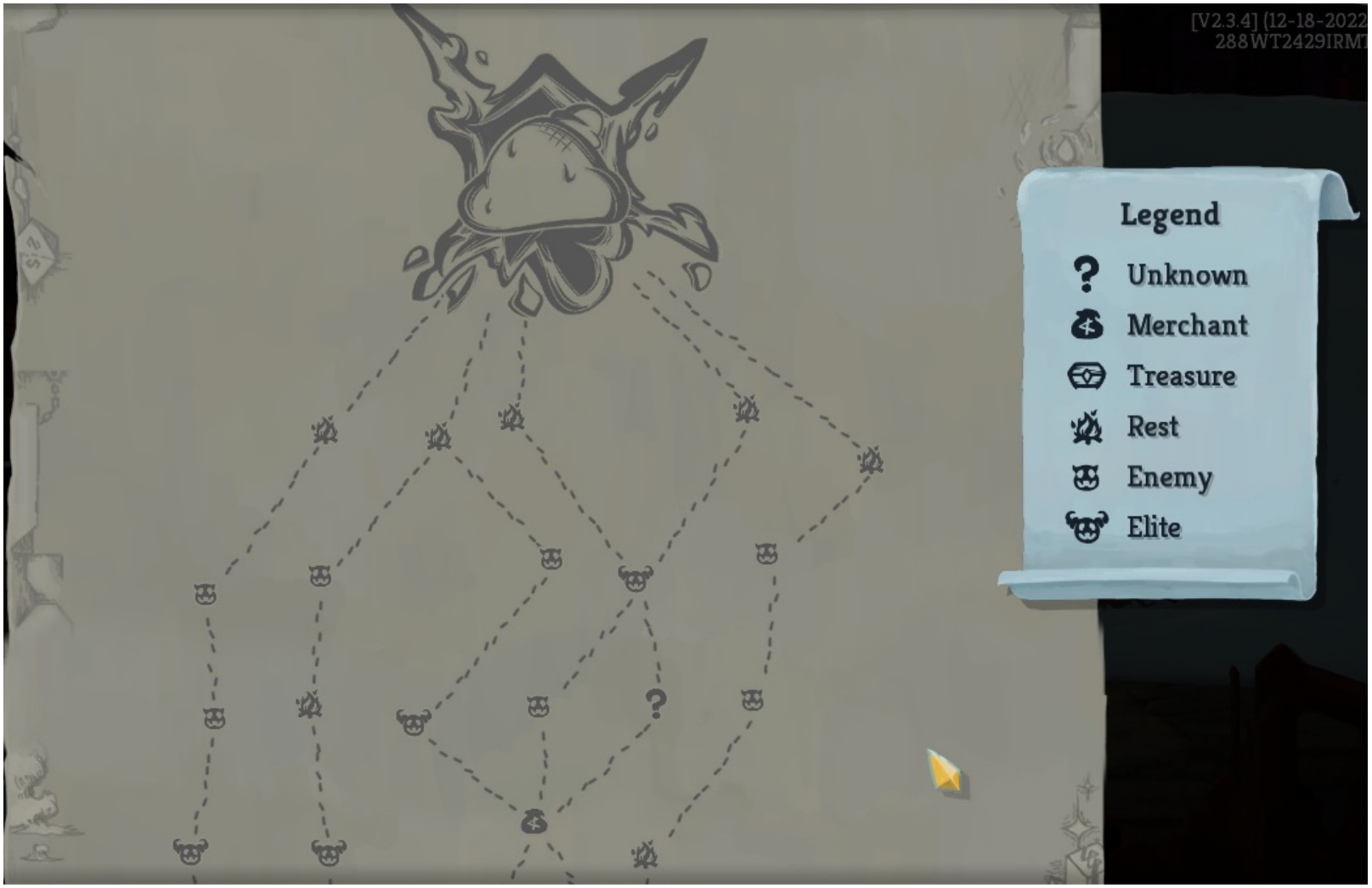


Example: NetHack

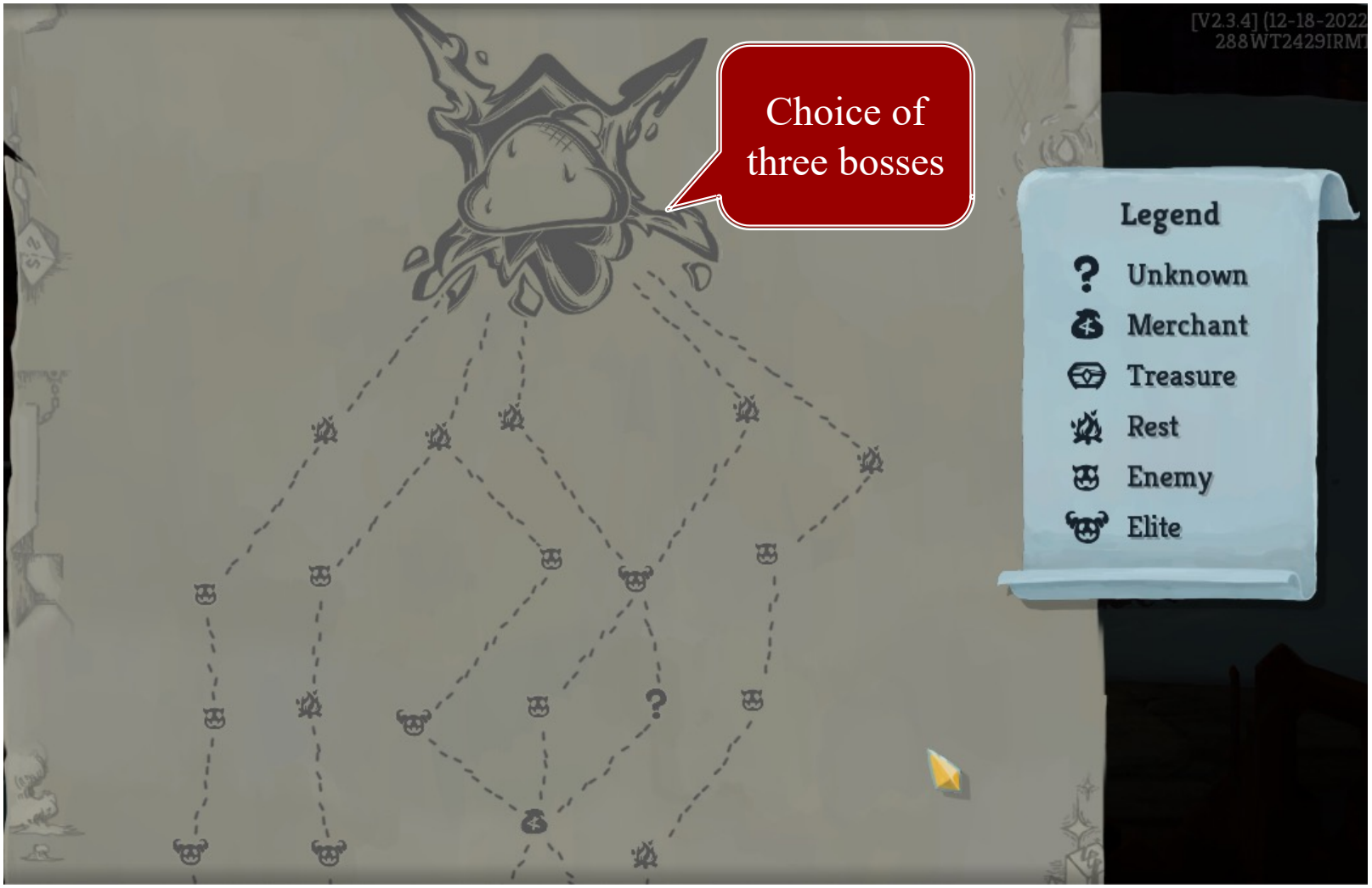


Case Study: *Slay the Spire*

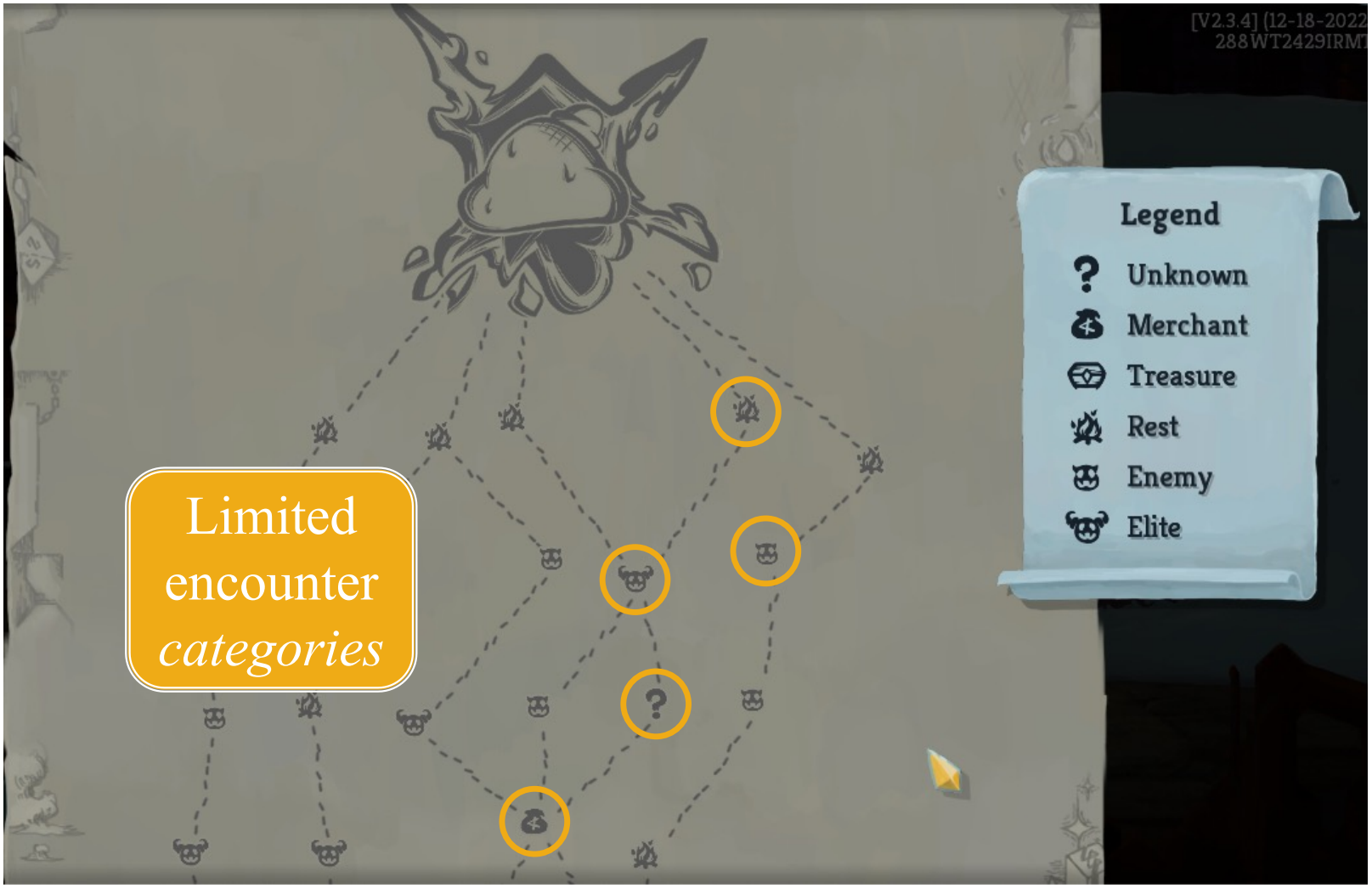
Case Study: *Slay the Spire*



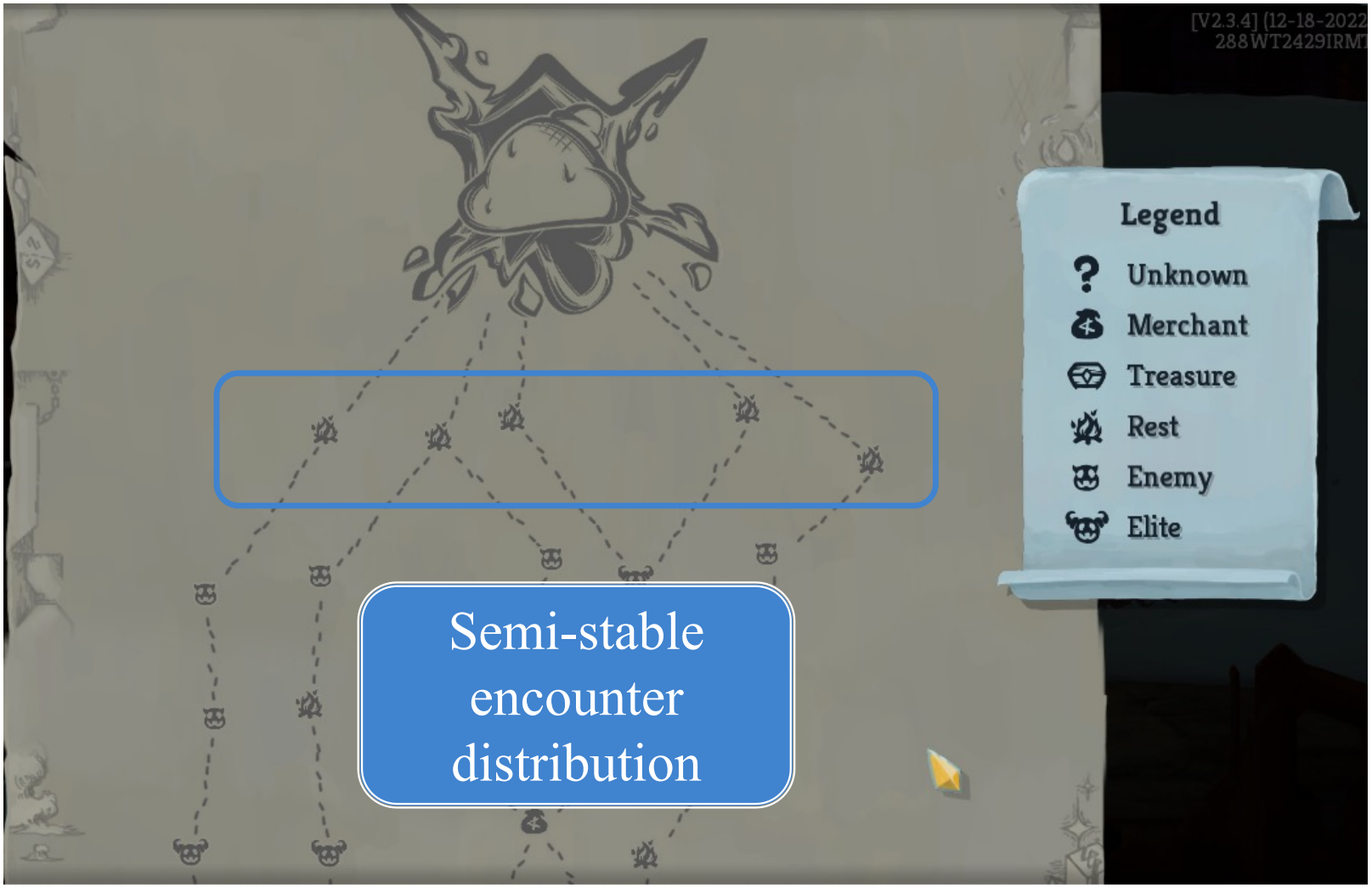
Case Study: *Slay the Spire*



Case Study: *Slay the Spire*



Case Study: *Slay the Spire*



Case Study: *Slay the Spire*

Other Considerations in *Slay the Spire*

- PCG includes card, relic and potion rewards
  - Like loot boxes, but rarity is tightly controlled
  - **Goal:** Ensure that a viable build is possible
- **Example:** Card Rewards
  - Each fight has a base rarity for each card
    - **Normal:** 3% Rare, 37% Uncommon, 60% Common
    - **Elite:** 10% Rare, 40% Uncommon, 50% Common
  - Numbers are offset by special percentage
  - Rares more likely each time a Common is rolled
  - Rolling a Rare resets the special percentage

Case Study: *Hades*

Case Study: *Hades*

Case Study: *Hades*

Not Just Map Generation

- Tempting to think of all PCG like NetHack
  - Generate paths from entrance to exit
  - Randomly generate content in-between
  - Just make sure the content gets "harder"
- Good roguelites understand full progression
  - **Entrance** = player abilities at start of run
  - **Exit** = fight with final boss at end of run
  - Constrain the **ludic space**, not just geography
- **Recall:** Card rarity in *Slay the Spire*

Skyrim's Radiant Quest System

- Geography includes NPCs
  - Mobile, removable location
  - Dialogue is also a space
- System "randomly" choses
  - Quest giver
  - Quest location
  - Location's challenges
  - Quest redeemer
- Randomness is limited
  - Lists appropriate to quest
  - Depends on earlier actions

- **Goals:**
  - Send to unexplored areas
  - Provide some predictability
  - Adjust challenges to level
  - Can never be missed

Skyrim's Radiant Quest System

- Geography includes NPCs
  - Mobile, removable location
  - Dialogue is also a space
- System "randomly" choses
  - Quest giver
  - Quest location
  - Location's challenges
  - Quest redeemer
- Randomness is limited
  - Lists appropriate to quest
  - Depends on earlier actions

Guarantees reachability

- unexplored areas
- Provide some predictability
- Adjust challenges to level
- Can never be missed

But Sometimes a Problem

But Sometimes a Problem

Example: *Card Crawl*

Example: *Card Crawl*

Grammars: A Formal Approach

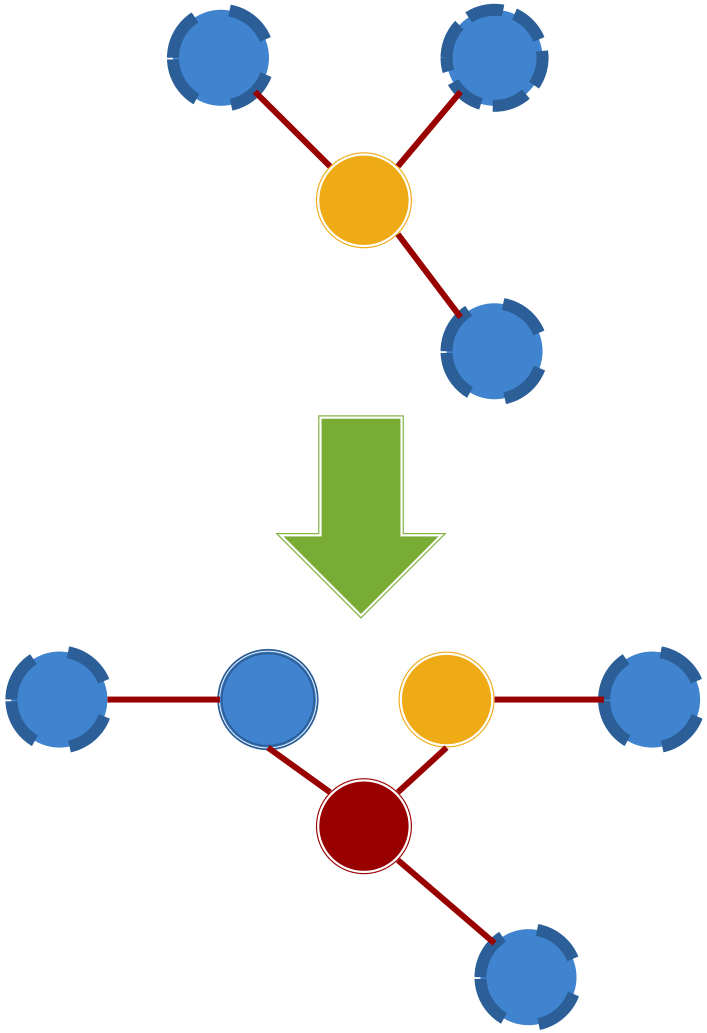
- **Notation**
  - Set $\mathcal{N}$ of nonterminals
  - Set Σ of terminal symbols
  - Set $\mathcal{P}$ of production rules
    - Have the form $A => B$
    - A, B are **words** of symbols
- To generate a value
  - Start with word XAY
  - Pick any rule $A => B$
  - Replace with XBY
  - Repeat until only terminals

Example

- $\mathcal{N}$ = { S, B}
- Σ = {a, b, c}
- $\mathcal{P}$ is the list of rules
  - S => aBSc
  - S => abc
  - Ba => aB
  - Bb => bb
- Possible **outputs**
  - abc, aabbcc, aaabbbccc, …

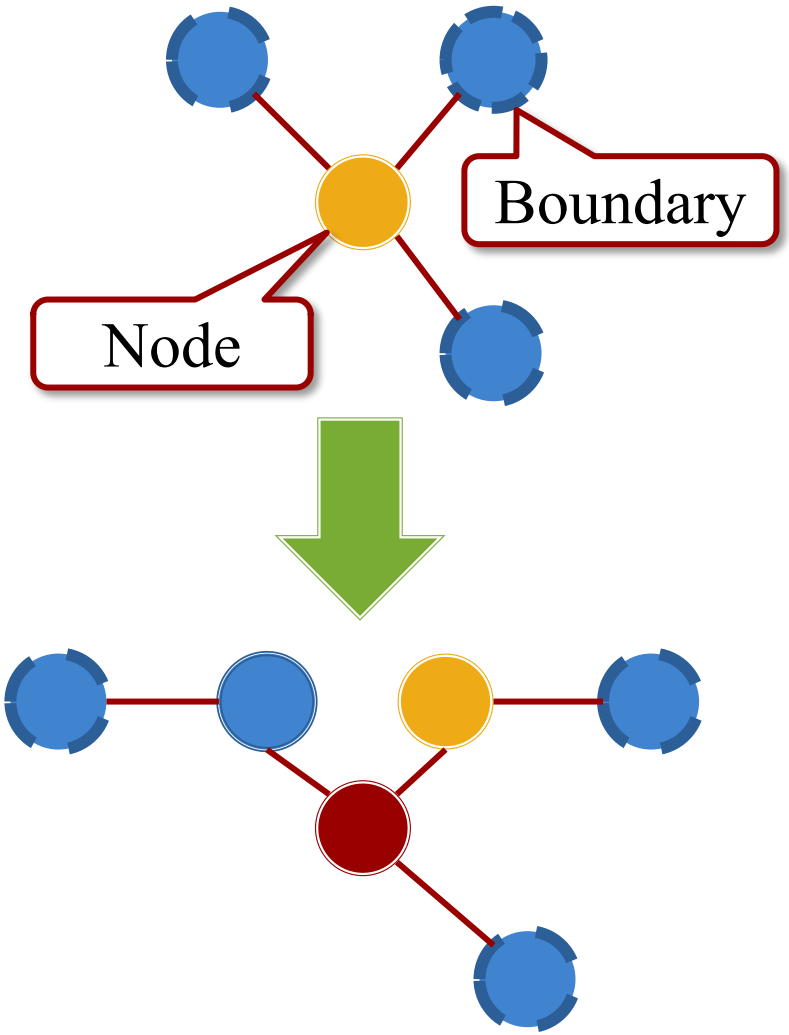
Grammars on Graphs

- Symbols are colored nodes
  - Either terminal or not
  - Edges replace word order
- Words are now graphs
  - Productions on subgraphs
  - LHS is node+boundary
  - RHS alters the node
- Output built as before
  - But rule matching harder
  - Graph equivalency



Grammars on Graphs

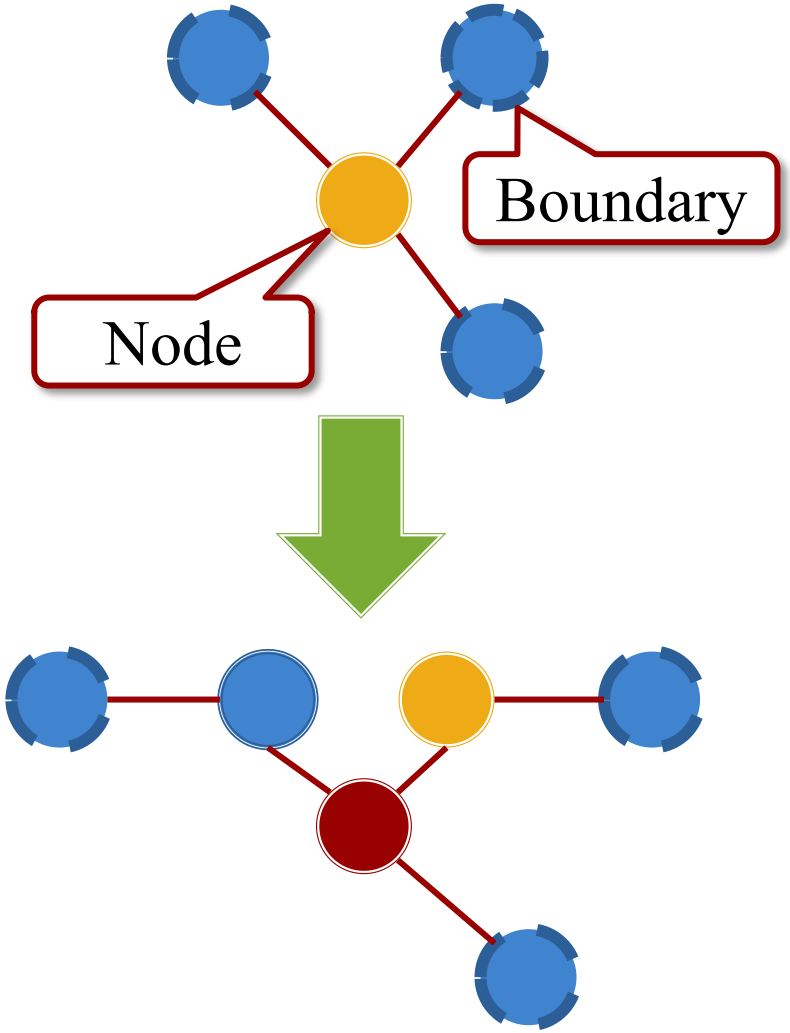
- Symbols are colored nodes
  - Either terminal or not
  - Edges replace word order
- Words are now graphs
  - Productions on subgraphs
  - LHS is node+boundary
  - RHS alters the node
- Output built as before
  - But rule matching harder
  - Graph equivalency



Grammars on Graphs

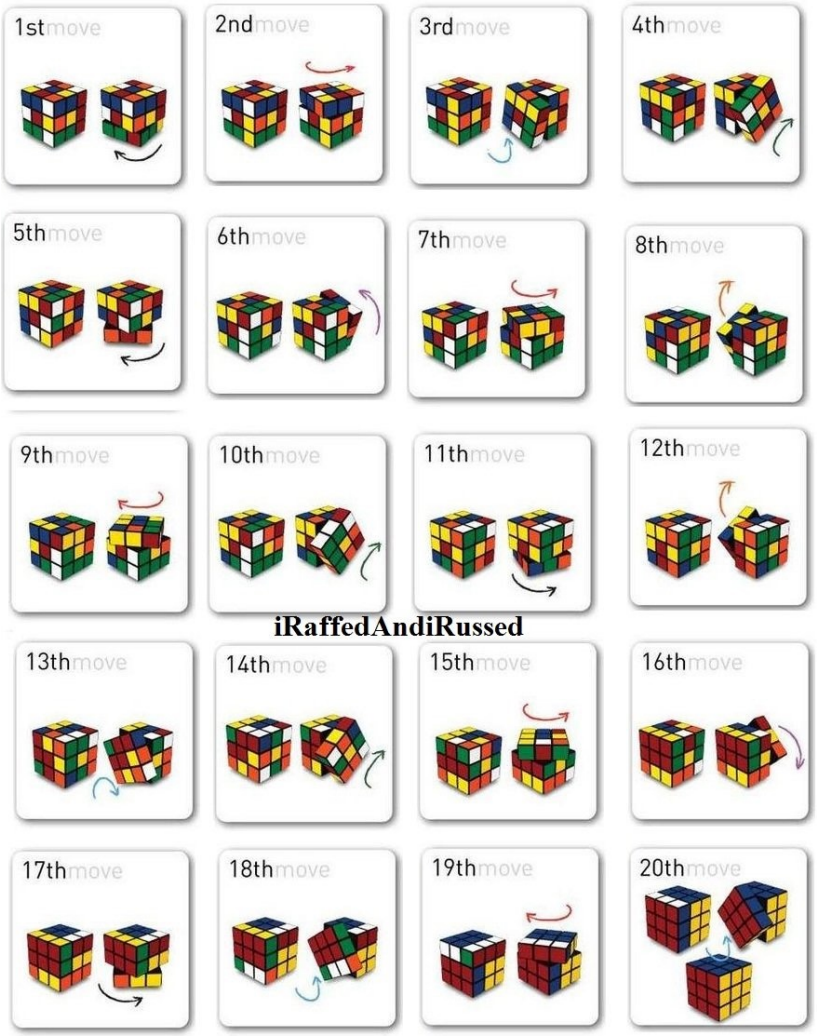
- Symbols are colored nodes
  - Either terminal or not
  - Edges replace word order
- Words are now graphs
  - Productions on subgraphs
  - RHS alters the node
- Output built as before
  - But rule matching harder
  - Graph equivalency

Levels are a graph

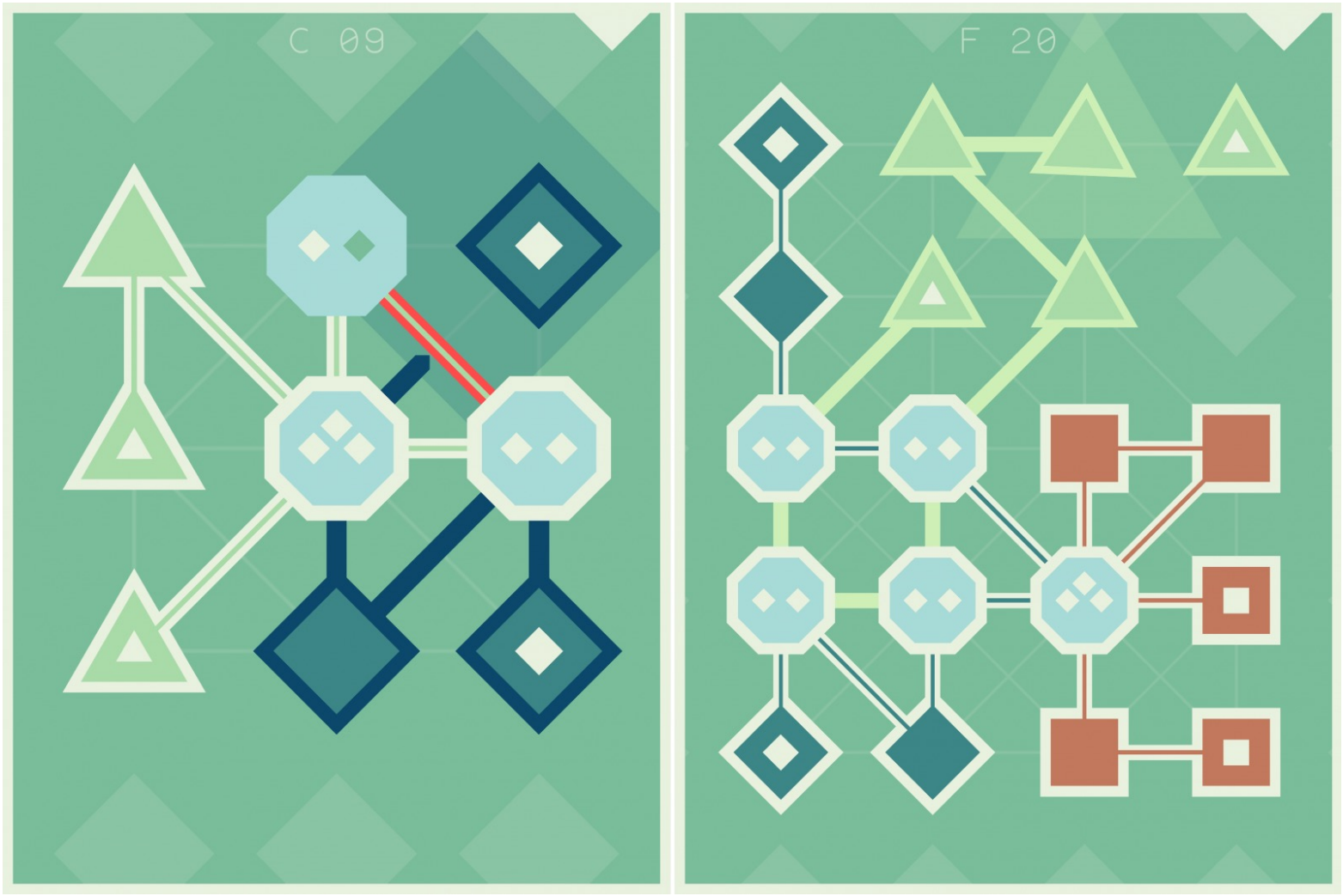


Puzzle Generation

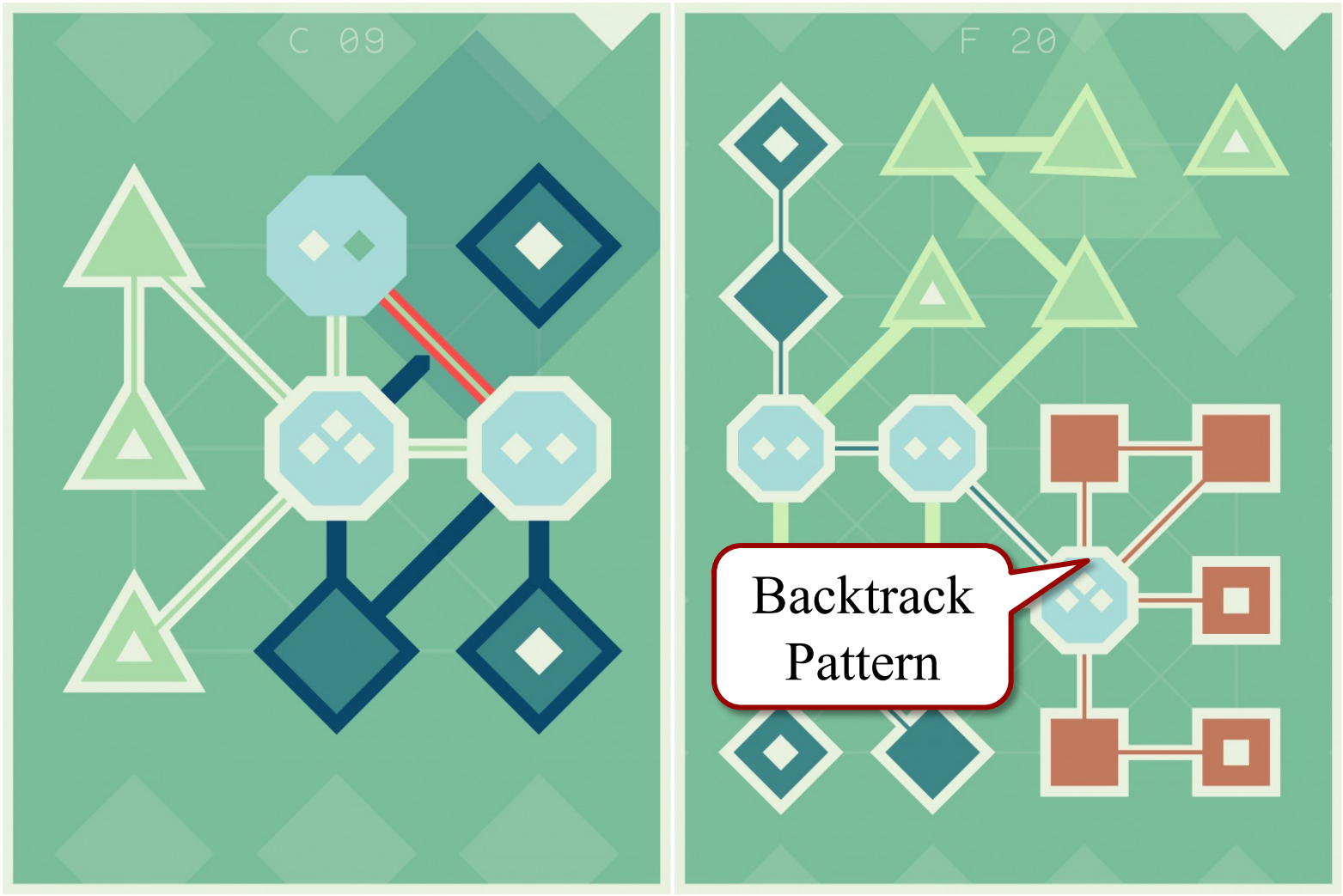
- Understand basic structure
  - Discrete actions/moves
  - Moves applied in sequence
  - **Goal**: get correct sequence
- Identify move sequences
  - Could be a loose category
  - Represent specific strategies
- Build up from sequences
  - Start from solved state
  - Invert moves (scrambling)
- Does require verification



Example: Lyne



Example: Lyne



Dynamic Challenges

- Dynamic: adaptive challenges
  - Become easier or harder
  - Or are just different
- **Example**: Autoleveling
  - NPCs have statistics
  - Adjust to character level
  - Difficulty always reasonable
  - Allows true "open" world
- Not always popular
  - Can lead to design recycling
  - Sense of risk is lost

ATK	1
DFN	0
HP	5

Rat: Level 1

ATK	30
DFN	10
HP	90

Rat: Level 50

Other Types of Dynamic Challenges

- **Composite Challenges**
  - Encounter is a collection of NPCs, obstacles
  - Add or remove individuals from encounter
- **Dynamic NPC AI**
  - NPCs have a choice of AI scripts
  - Choose one that matches the player
- **Player Boosting**
  - Change result of player actions, interactions
  - Modifications make challenges easier/harder

Assigning Dynamic Challenges

Player

Extract feature vector from play history

$(a_1, a_2, a_3, \ldots, a_n)$

Match the challenge to the play style

$(b_1, b_2, b_3, \ldots, b_k)$

Challenge

Parameterize challenge difficulty

Assigning Dynamic Challenges

Player **Challenge**

Matching Function is hardest to balance

Extract feature vector from play history

Match the challenge to the play style

Parameterize challenge difficulty

$(a_1, a_2, a_3, \ldots, a_n)$

$(b_1, b_2, b_3, \ldots, b_k)$

Adaptive Difficulty

Player

Extract feature vector from play history

$(a_1, a_2, a_3, \ldots, a_n)$

Match via **machine learning**

$(b_1, b_2, b_3, \ldots, b_k)$

Challenge

Parameterize challenge difficulty

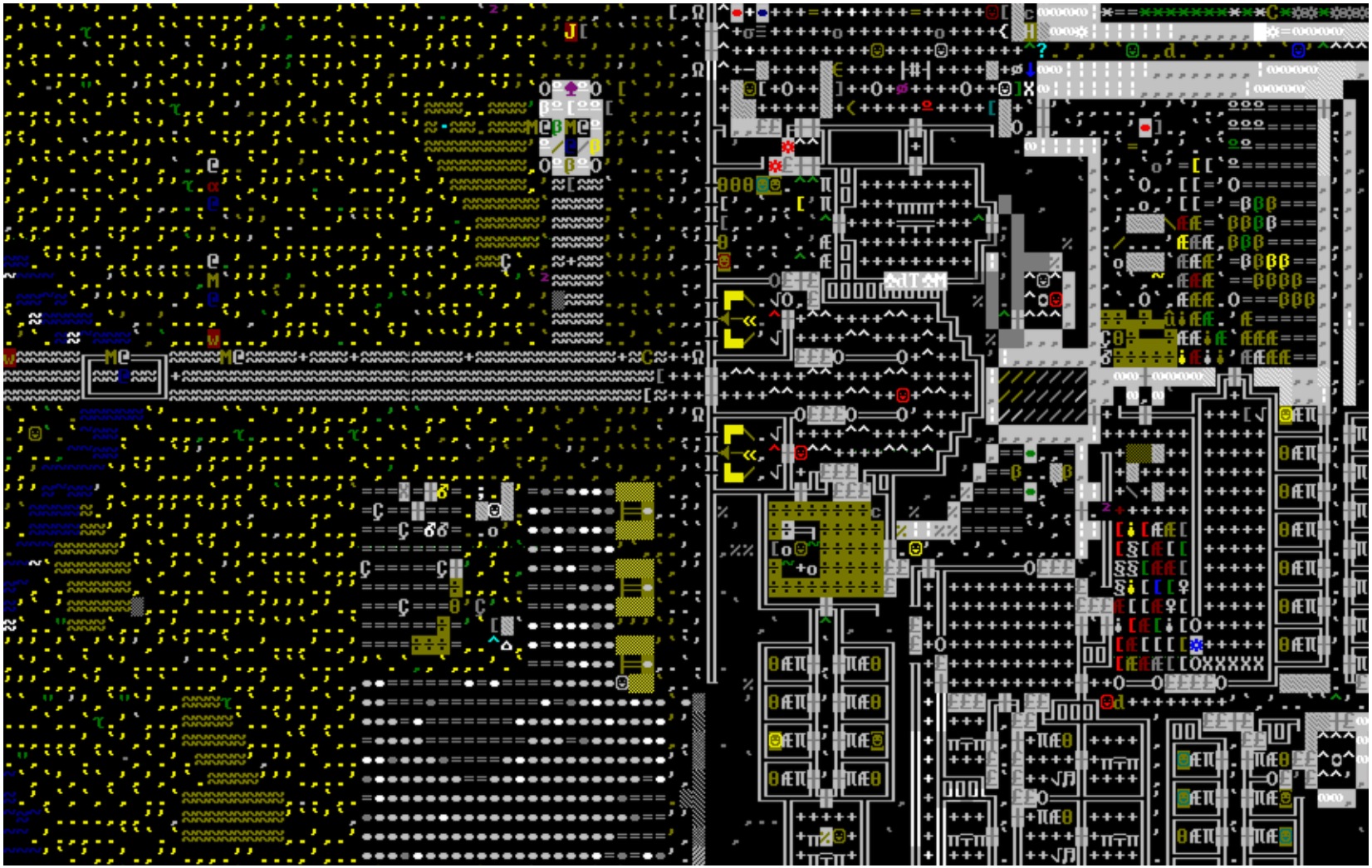
Adaptive Difficulty

- Manually define the **gameplay model**
  - Metrics that identify player behavior
  - Parameters that define challenge behavior
  - Also metrics to evaluate player success or failure
- **Goal**: Use ML to find player-challenge match-up
  - Use playtesting/beta to get a large training set
  - Create an initial model from these results
  - Adjust in the game according to current player
- This has become popular in the industry

Story Generation

- **Narrative** is tightly crafted
  - Must have emotional arc
  - Very hard to generate
- But **backstory** is looser
  - Collection of tales/subplots
  - Combine to form a story
  - Often displayed in a codex
  - Much easier to generate
- **Idea**: Create list of subplots
  - Pick some subset at a time
  - Mix with NLG techniques

Example: Dwarf Fortress



Generative AI Has Promise Here

- Area known as **N**atural **L**anguage **G**eneration
  - **Given**: complex set of data
  - **Outcome**: commentary on that data
  - *Exactly* the problem LLMs are designed for
- **Examples**
  - Sports commentary
  - Party combat chatter
  - Intelligent townsfolk

- Major topic of research in game studios

How Far Can We Take It?

Leaked Internal Sony Video Uses Horizon's Aloy to Show AI-Powered PlayStation Character Prototype

"This is just a glimpse of what is possible..."

A leaked internal video has revealed Sony is experimenting with AI-powered PlayStation characters.

The Verge reported on the emergence of an internal video allegedly created by PlayStation Studios' advanced technology group that uses Aloy from the Horizon games to demonstrate AI-powered game characters.

The video was subsequently pulled from YouTube following a copyright claim from Muso, an internet enforcement company The Verge said lists Sony Interactive Entertainment as a client. This suggests the video is legitimate. IGN has asked Sony for comment.

In the video, as reported by The Verge, Sony Interactive Entertainment director of software engineering, Sharwin Raghoebardajal, has a conversation with an AI-powered Aloy via voice prompts and AI-generated speech and facial animations.

Storytelling Has Constraints Too!

- Conversations in games have **purpose**
  - Start/end a quest
  - Inform us about a location/challenge
  - Define our character
  - Elicit reactions to our behavior
- We want this dialog to be **economical**
  - The NPC should not ramble
  - The information should be pertinent
  - The delivery should be dramatic
- But these are constraints LLMs can handle

Generative Storytelling

- **Hidden Door:** Founded by Hillary Mason
  - Former founder of Fast Forward Labs
  - Combining NLG and game development
- Games to support roleplaying in literary worlds
  - Interactive, but narrative-driven games
  - Targeting the fan fiction market
- Available to play, but still proof of concept
  - https://www.hiddendoor.co

Hidden Door

Create your universe.

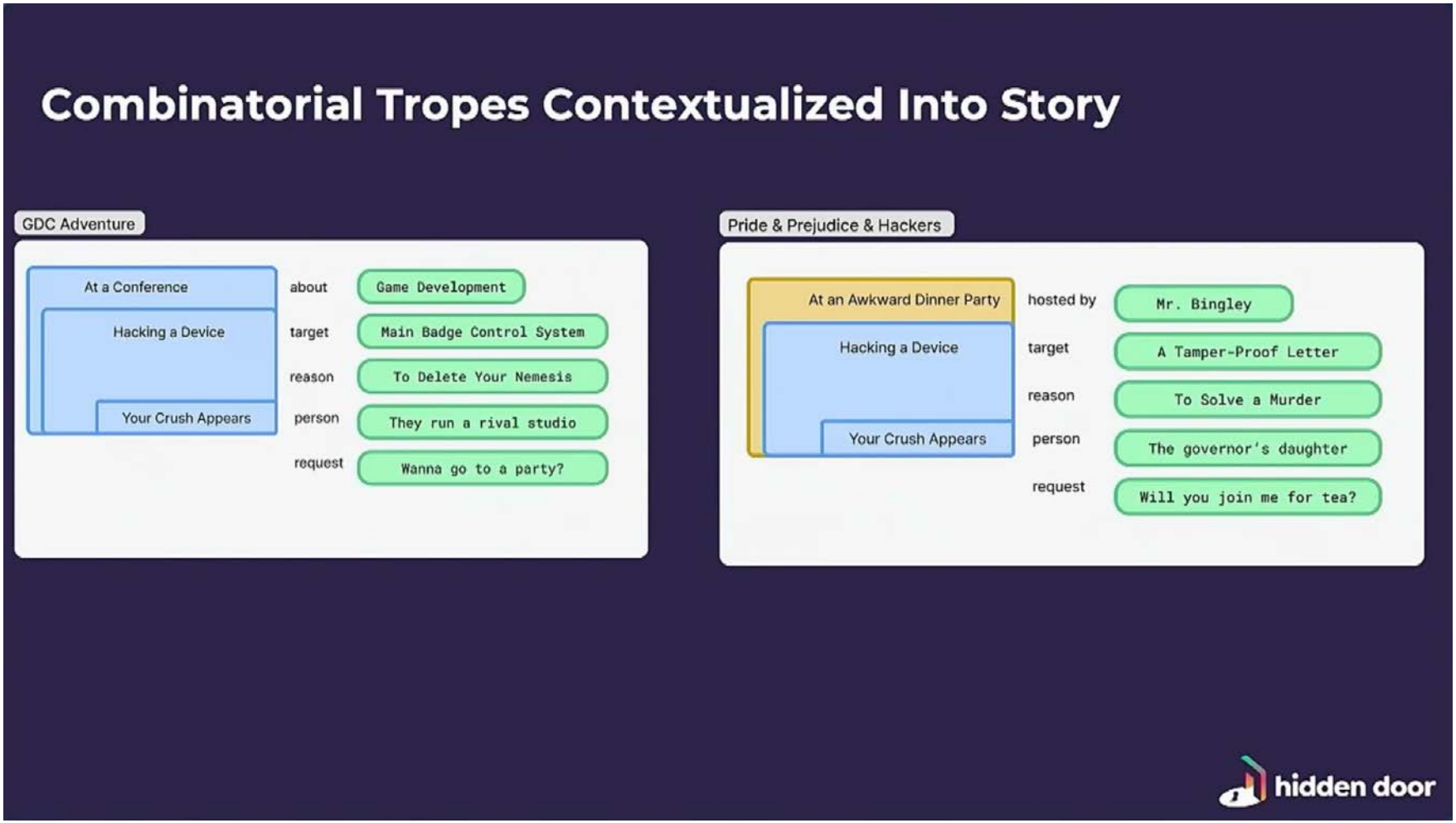
Hidden Door is a platform for fans to roleplay in the worlds of their favorite books, movies, and TV shows.

Working directly with authors and IP holders, our Narrative AI allows us to collaboratively adapt existing works in just a few hours.

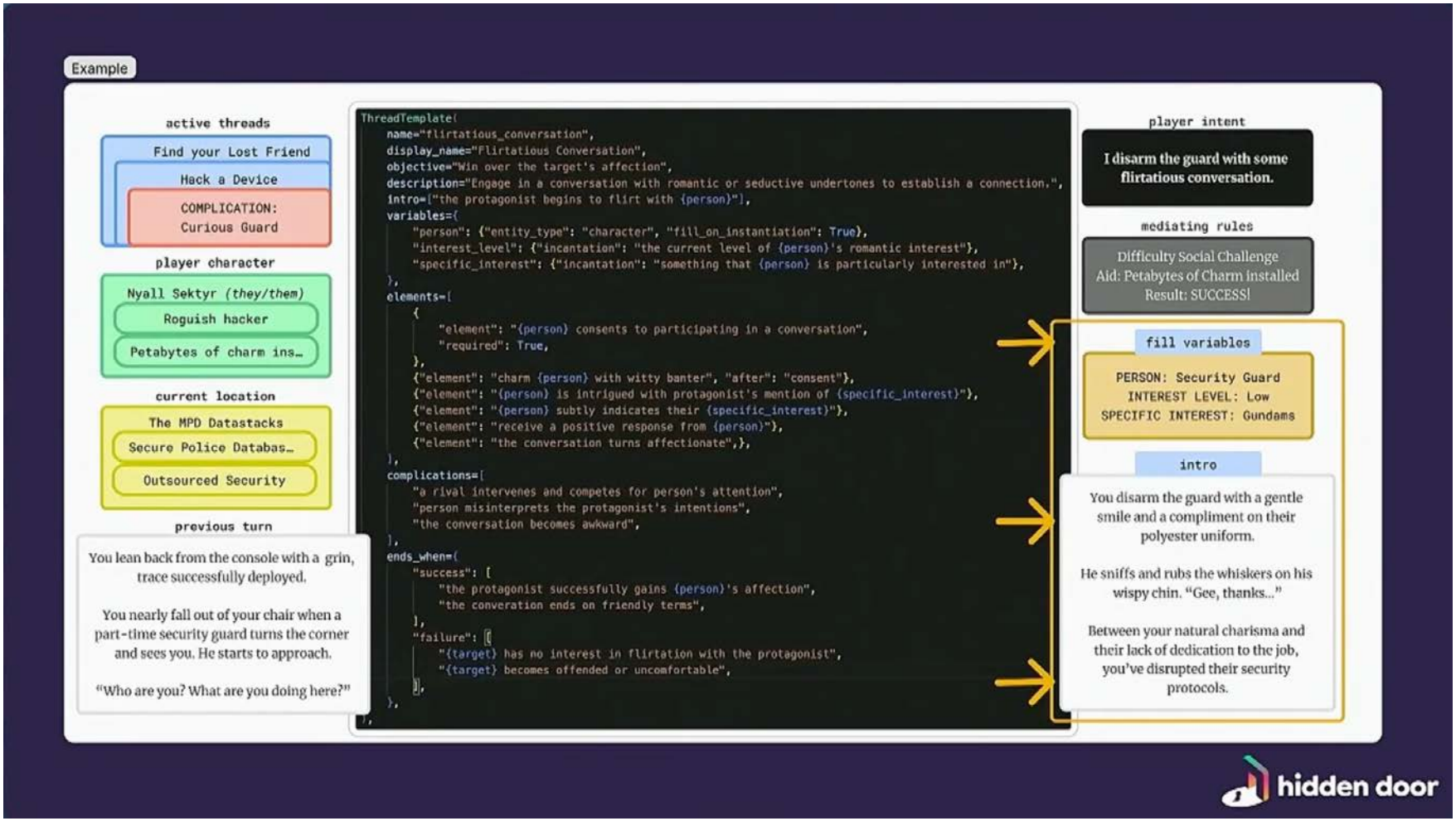
The game is a tabletop-like social adventure, with an "AI Dungeon Master" who provides controls for the rules of the world and safety.

hidden door

Basic Idea: Story Templates



ML/LLMs to Convert “In Universe”



Summary

- Procedural content started with Rogue(likes)
  - Randomly generated, but reasonable-looking maps
  - Coupled with permadeath, horizontal design
- Content generation requires **constraints**
  - Unconstrained levels are not solvable
  - Current tech cannot adapt to LLMs
- **N**atural **L**anguage **G**eneration is the hot topic
  - This is the application LLMs were designed for
  - Technology here is bleeding-edge, proof-of-concept

Summary

- Procedural content started with Rogue(likes)
  - Randomly generated, but reasonable-looking maps
  - Coupled with permadeath, horizontal design
- Conten
  - Unco
  - Curre
- **N**atural **L**anguage **G**eneration is the hot topic
  - This is the application LLMs were designed for
  - Technology here is bleeding-edge, proof-of-concept

Procedural Content Wiki:
http://pcg.wikidot.com