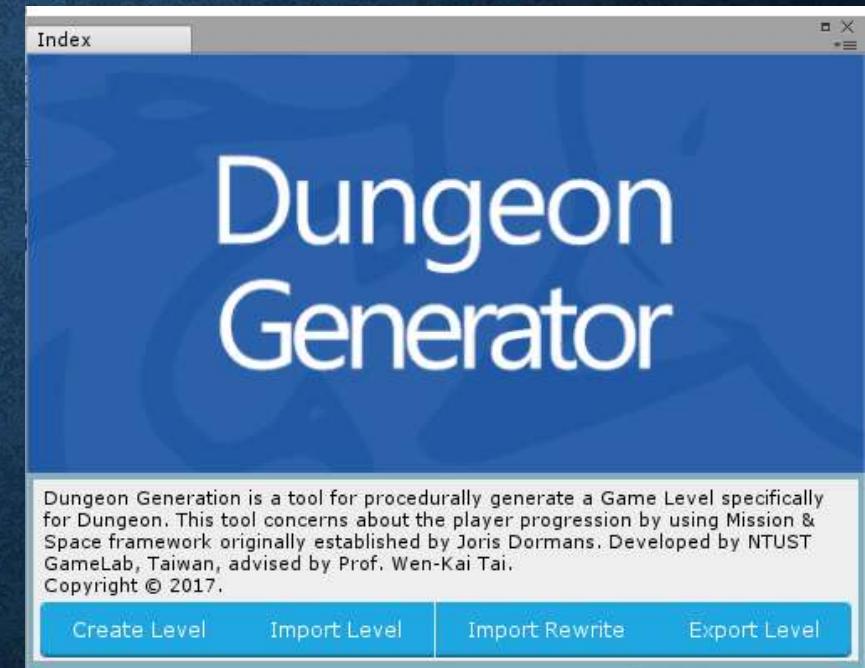


DUNGEON GENERATOR

Mission Grammar System

DUNGEON GENERATOR

- Released version 0.1.0
 - Completely optimized UI



COMPLETELY OPTIMIZED UI

Mission alphabet

Nodes **Connections**

LIST OF NODES

- <0> any (?)
- <0> none (none)
- <0> en entrance (en) **(highlighted)**

Add New Modify Delete

Symbol Type: Terminal
Name: entrance
Abbreviation: en
Description: System default.

Outline Color:
Filled Color:
Text Color:

The data is up to date.

Update the changes

Alphabet

Mission rules

Current Group: Master Dungeon
 Current Rules: Entrance

SOURCE **REPLACEMENT**

```

graph LR
    S["<1> S"] --- EN["<0> en"]
    EN --- X["<3> X"]
    X --- CR1["<4> CR"]
    CR1 --- F["<5> F"]
    F --- G["<7> G"]
    G --- GO["<2> go"]
  
```

Add Node Add Connection Copy Delete

LIST OF NODES

- <0> any (?)
- <0> none (none)
- <0> en entrance (en) **(highlighted)**

Add New Modify

Info: The name has been used.

Apply

Rules

Generate missi

Mission Graph **Space Graph**

Starting Node: Start (S)

Mission Graph

```

graph LR
    S["<0> en"] --- X["<0> X"]
    X --- CR1["<0> CR"]
    CR1 --- F["<0> F"]
    F --- CR2["<0> CR"]
    CR2 --- GO["<0> go"]
    GO --- G["<0> G"]
  
```

No error occur!

Master Dungeon

- Entrance
- More Paths
- Fight boss
- Fork
- Gate

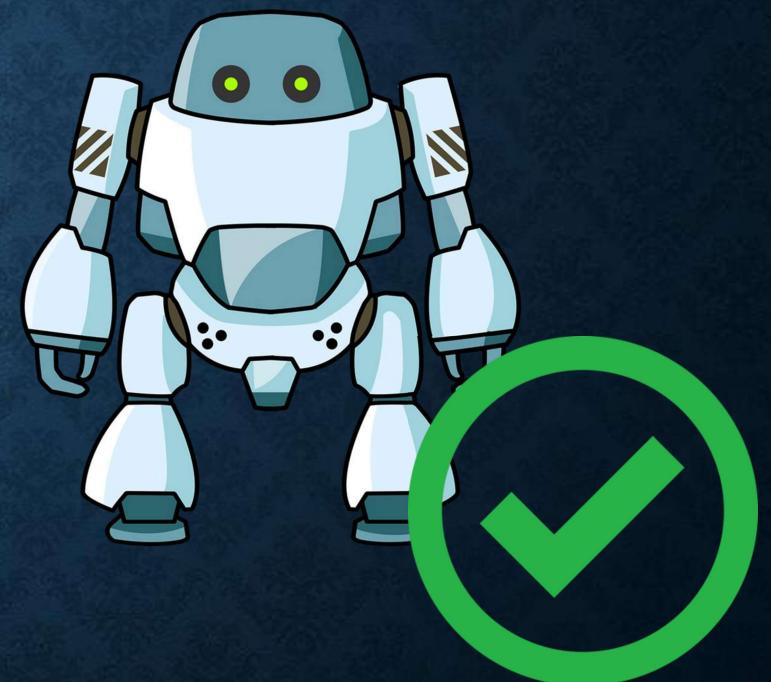
Initial **Iterate** **Complete**

Save

Rewrite System

NEXT TASKS

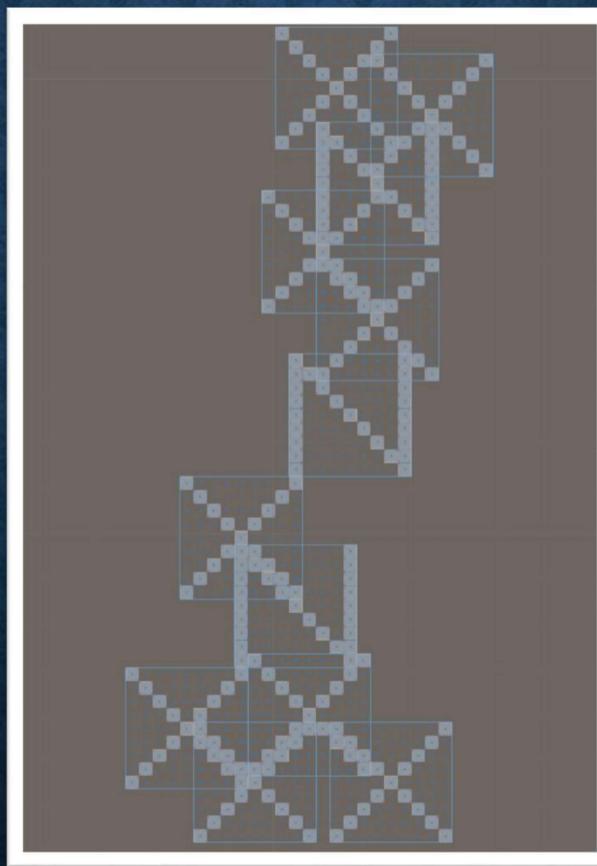
- Validation system
 - Auto-detect the **illegal rules** and ignore them
 - Hint and guide the correct design patterns
- Optimize the user experience
 - Based on the feedback



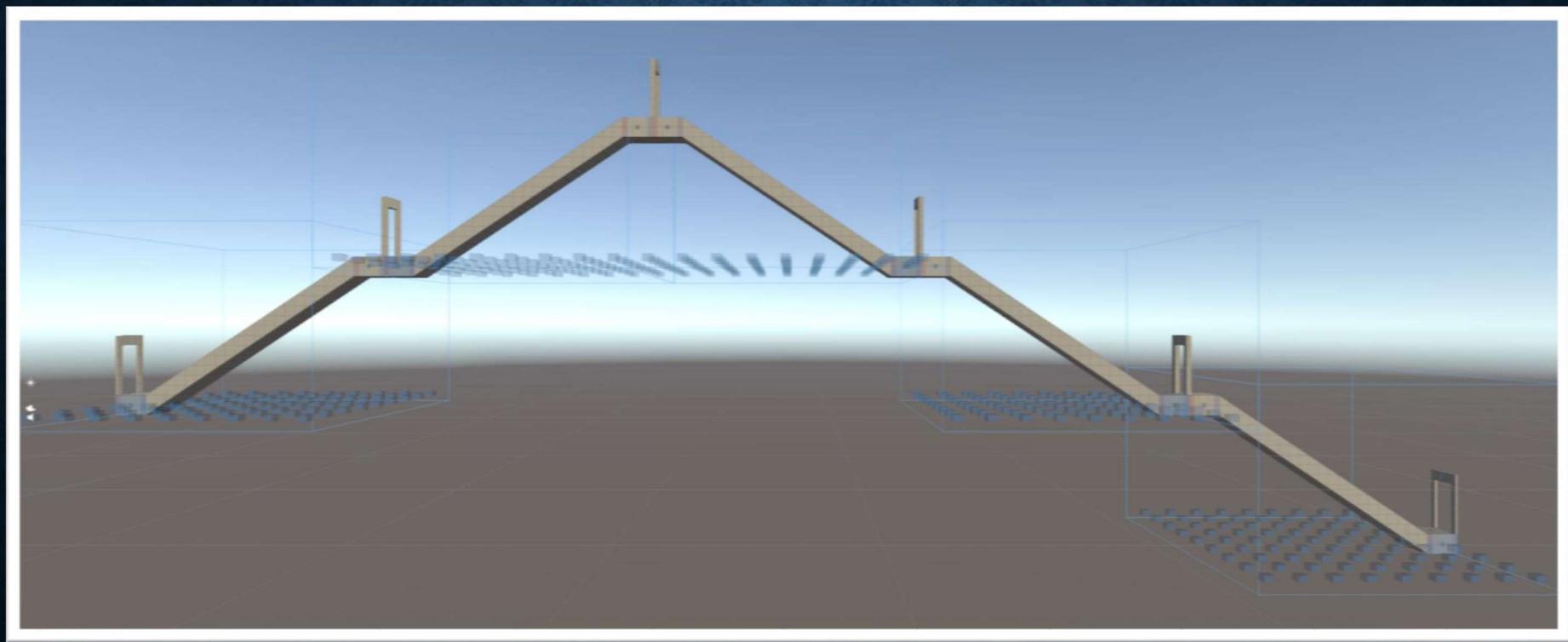
CREVOX

Automatically generated

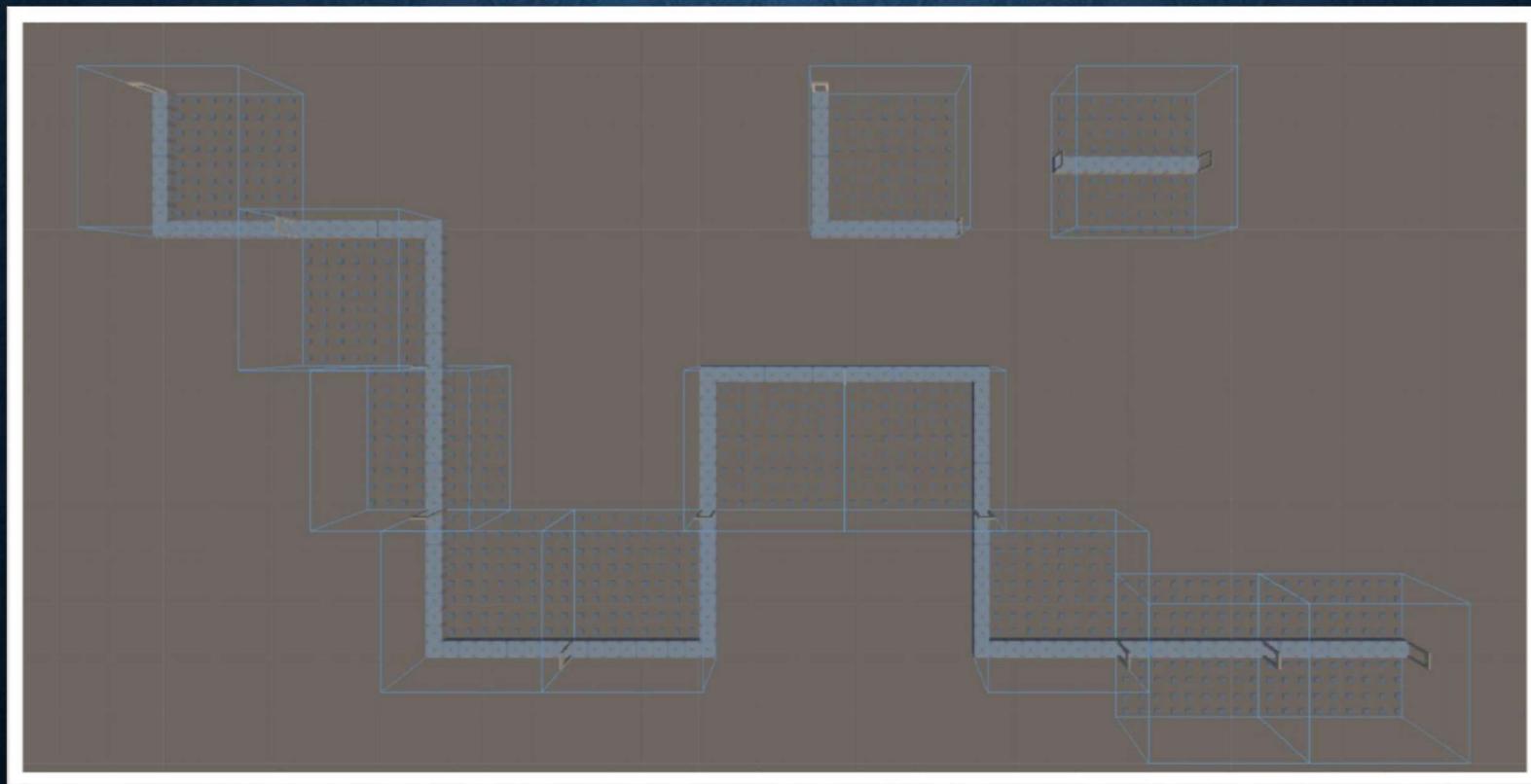
COLLIDATION



FLOOR



ROTATION

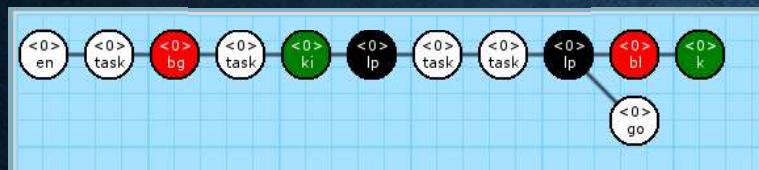


CREVOX

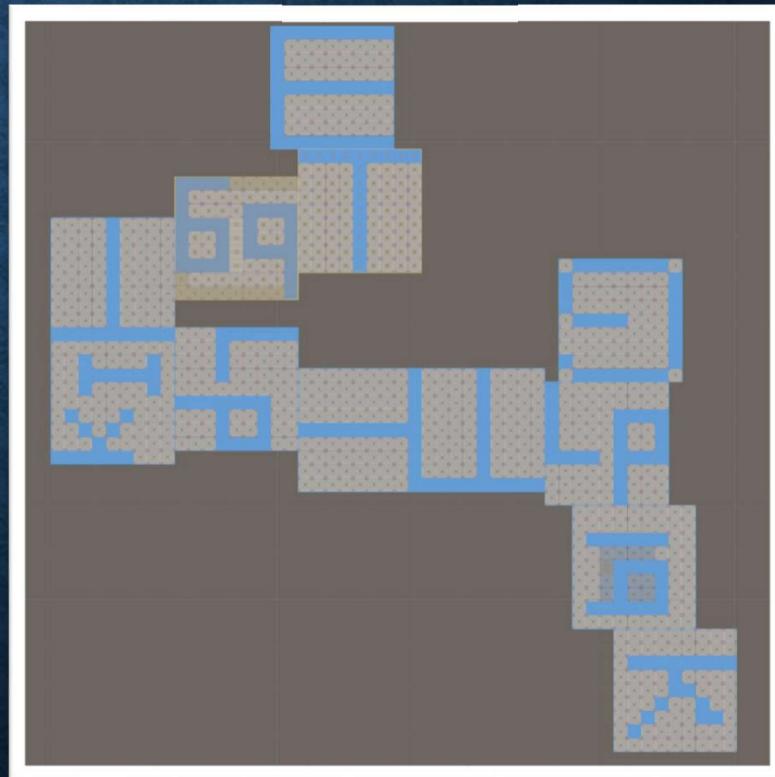
Connect with Mission Grammars

INSTRUCTION

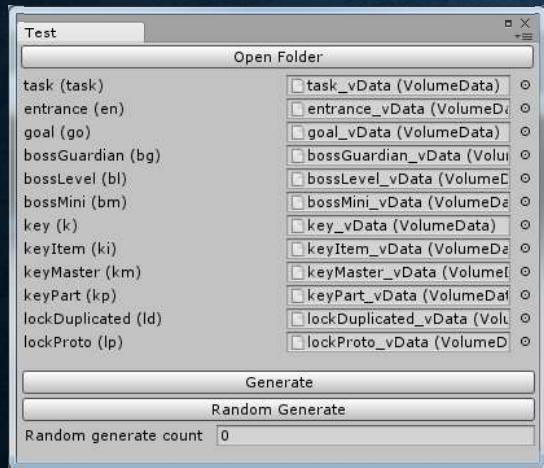
Current graph



Export

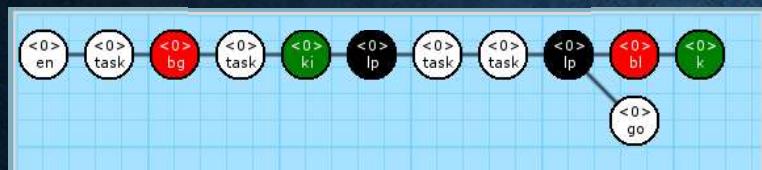


Instruction setting



INSTRUCTION

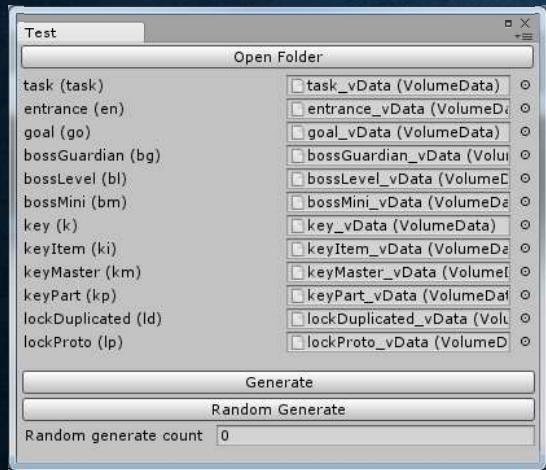
Current graph



Export

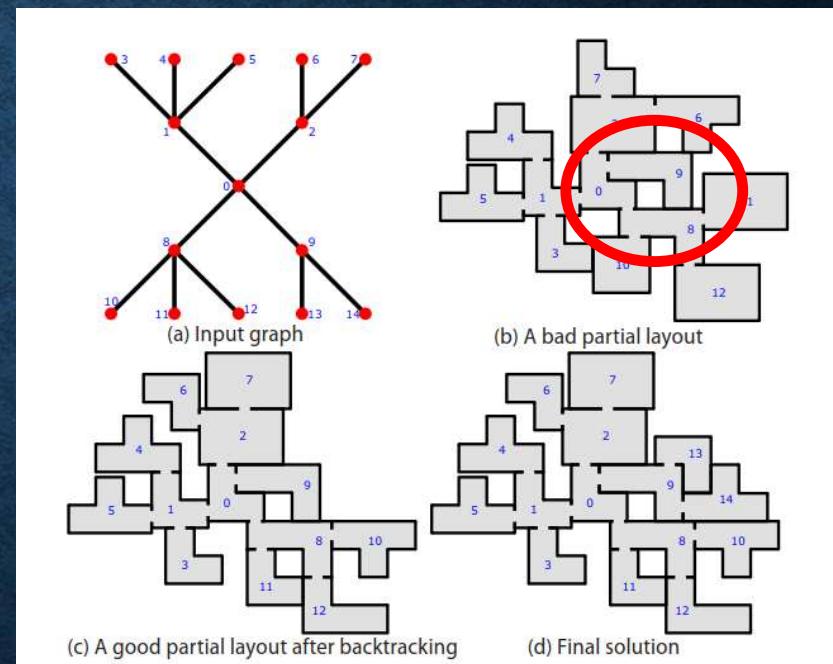


Instruction setting



[ISSUE] DEAD END

- Solution
 - Backtracking (Better than *DunGen*)
 - If backtracking fail
 - Find the **bottleneck** and report
 - Make advices for end user



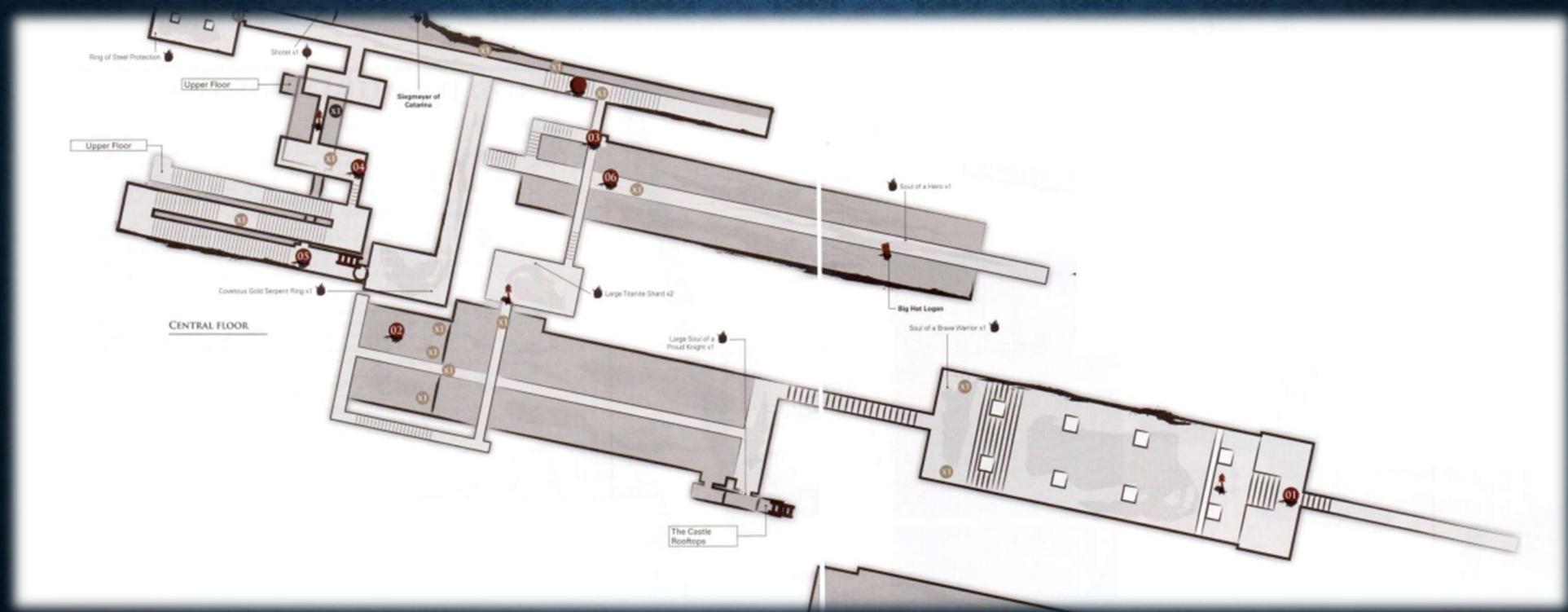
CREVOX

Live DEMO

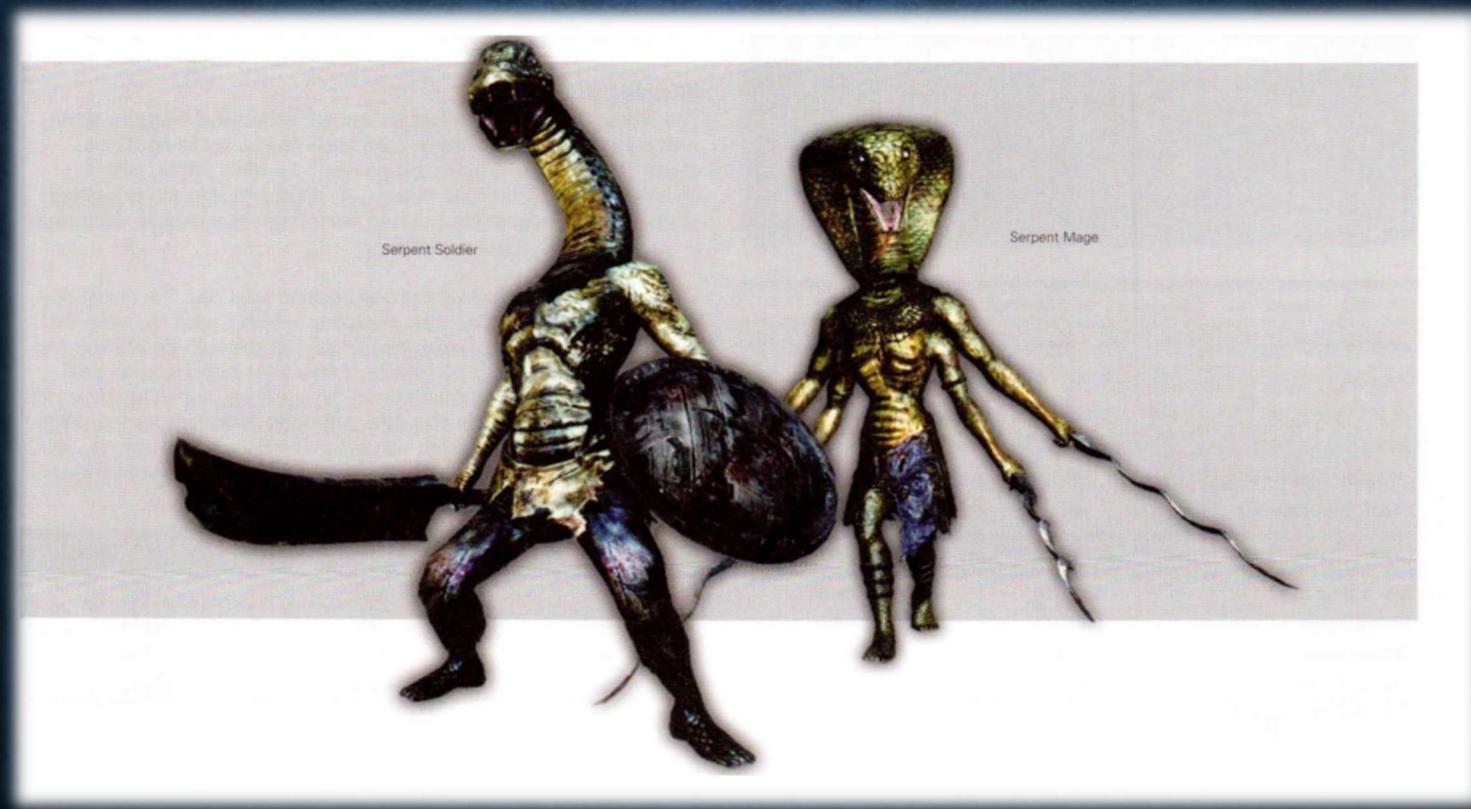
IDEA PITCHING

Game patterns of monster set

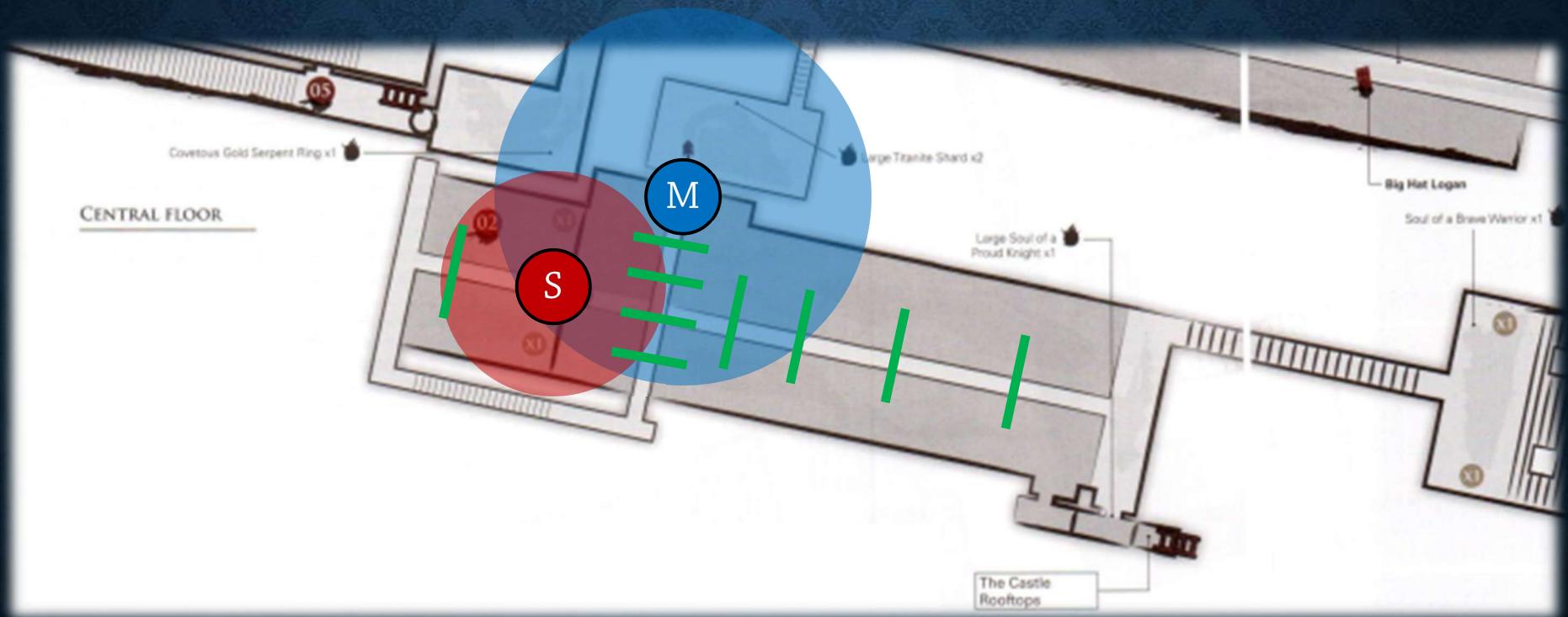
MAP: SEN'S FORTRESS



MONSTERS: SERPENT SOLDIER & MAGE



STAGE



S Soldier

M Mage

/ Sickle

OUR SOLUTION

- 線性規劃 (Linear Programming)
 - Optimization of a linear objective function
 - The greatest enemy set for map
 - The suitablest position
 - Friendly interface to set constraints

REQUIREMENT

- Current space
 - There're two floors in the room
- Target
 - Generate recommendation about the set of enemies
 - There are traps will support enemies in these space
 - Ignore the distance between trap and enemy

LP in Gameplay Patterns

Objective function:

$$T = T_A \times \log(N_{T1} \times T_{T1}) + T_B \times \log(N_{T1} \times T_{T1} + N_{T2} \times T_{T2})$$

T_m : Base threat value of enemy.

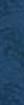
N_{Ti} : Amount of Trap in i-th floor.

T_{Ti} : Threat value of Trap in i-th floor.

LP in Gameplay Patterns

Objective function:

$$T = \textcolor{teal}{T}_A \times \log(N_{T1} \times T_{T1}) + \textcolor{blue}{T}_B \times \log(N_{T1} \times T_{T1} + N_{T2} \times T_{T2})$$



$$T_B = c_1 \times \textcolor{red}{R}_B + c_2 \times \textcolor{red}{B}_B + c_3 \times \textcolor{red}{H}_B$$

$$T_A = c_1 \times \textcolor{red}{R}_A + c_2 \times \textcolor{red}{B}_A + c_3 \times \textcolor{red}{H}_A$$

R_m : Enemy's radius of active attack in vision.

B_m : Enemy's size of battle area.

H_m : Enemy's height of position.

LP in Gameplay Patterns

Enemy table:

	Fighter	Fury	Shooter
Radius of active attack in vision	15-20	10-20		30-50
Size of battle area	2-9	5-9		4-9
Height of position	0-5	0-5		0-7
:				

Trap table:

	Sickle	Arrow
Threat	5		2

Constraints:

1. $T > 50$
2. $T_A > 15$
3. $T_B > 30$
4. $B_A > 2$
5. $B_B > 2$
6. $0 < H_A < 6$
7. $H_B \geq 6$

Solution:

Enemy A = { Fighter }
Enemy B = { Shooter }
Trap = { Sickle }

FRAMEWORK

1. **Dungeon Generator** (1st Phase)
 - Output: Mission graph
2. Enhanced **CreVox** (2nd Phase)
 - Input: Mission graph
 - Output: Volume data map
3. **LP in Gameplay Patterns** (3rd Phase)
 - Input: Volume data map
 - Output: Final level

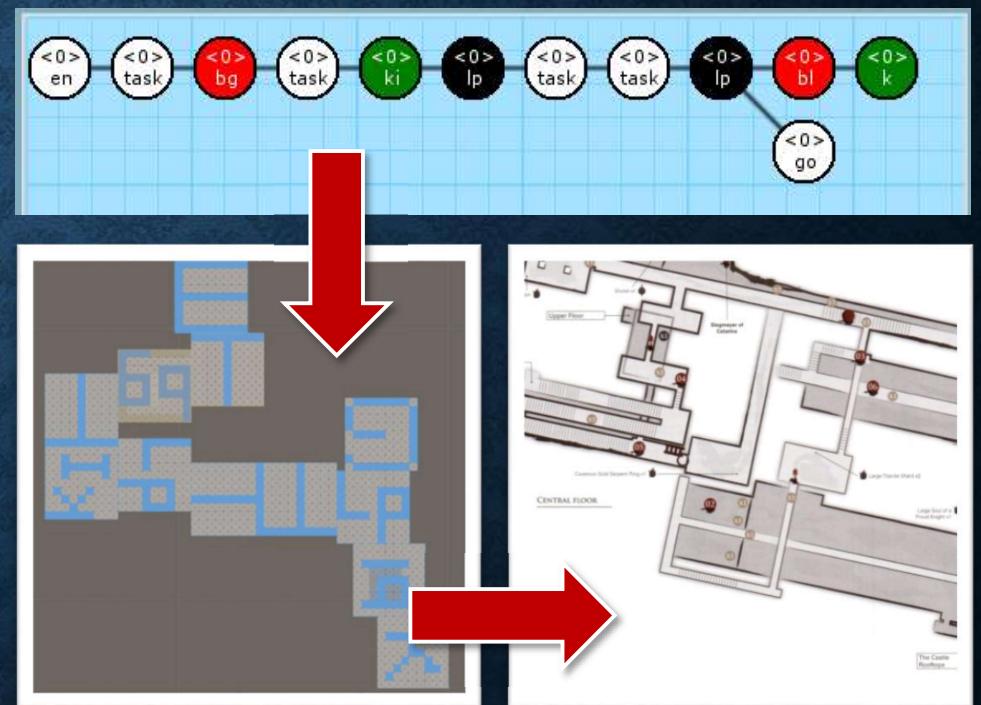


Table and Constraints

- Pre-construct the table
 - Enemies
 - Emergemce items
 - Treasures
- Semantic constraints
 - 當 Label A (ATK) 大於 N , 且小於等於 M 時。
 - 當 Label B (weapon type) 為 S 時。
 - 當 Label C (HP) + Label D (MP) 不超過 M 時。

Enemy table:

	Label A	Label Z
Enemy A			
Enemy B			
...			
Enemy Z			

Emergence item table:

	Label a	Label z
Item a			
...			

IDEA PITCHING

Enemy List for XAOCX

<http://goo.gl/9CDo61>

DISCUSSION