@ ? d: -> =>

Lonelog: Dungeon Crawling Add-on

Optional Room Tracking for Dungeon Exploration

Version 1.2.0



Table of contents

Dungeon Crawling Add-on **1**
Overview 1
1. The Room Tag 4
2. Room Connections 8
3. The Dungeon Status Block 10
4. Room Tags in Play 12
Cross-Add-on Interactions 14
System Adaptations 17
Best Practices 20
Quick Reference 23
FAQ 26
Credits & License 28

Dungeon Crawling Add-on

Overview

Lonelog was built for narrative-driven solo play—the kind where you care more about *what happened and why* than *where am I and what do I have left*. Dungeon crawling pulls in a different direction: spatial relationships matter, rooms accumulate state, and you're constantly asking "did I already loot that chest?"

That tension is real. Text notation is never going to replace a good map. But it *can* help you track **room state**—what you found, what you cleared, what's still waiting behind a locked door—alongside the narrative Lonelog already handles well.

This add-on introduces one new persistent element: the Room tag `[R:]`. That's it. Everything else—actions, rolls, oracle questions, consequences, threads, clocks—stays exactly the same. You're adding rooms to the vocabulary of things Lonelog can track.

If the dungeon is more narrative backdrop than mechanical challenge, or if you're moving through spaces quickly without tracking state, stick with core Lonelog. The `[L:]` location tag handles simple spatial context fine. This add-on earns its keep when rooms have meaningful state that changes over time.

What This Add-on Adds

- `[R:ID|status:state]` — Track a room's current exploration state. Example: `[R:3|status:cleared]`
- `[R:ID|status:state|desc]` — Room tag with optional description. Example: `[R:3|status:cleared|library]`
- `exits:DIR ID` — Record connections between rooms (optional). Example: `exits:N R2, E R3`
- `[#R:ID]` — Reference a previously established room. Example: `[#R:3]`
- `[DUNGEON STATUS]` / `[/DUNGEON STATUS]` — Session-opening snapshot of all room states. Example: Multiple `[R:]` tags grouped

No new core symbols. This add-on introduces no additions to `@`, `?`, `d:`, `->`, or `=>`.

Design Principles

State over space. A map handles spatial relationships better than text ever will. Room tags track what *changed*—cleared, looted, locked—not where rooms sit relative to each other. Use the right tool for each job.

Minimal vocabulary. One new tag covers the entire add-on. Every other element—scene headers, oracle questions, consequences, clocks—uses the core notation you already know. The cognitive overhead of adding dungeon crawling to a Lonelog session is one tag format.

Map-friendly by design. Room tags are built to work alongside a visual map, not replace one. Room IDs connect your log to your map. Mark them on graph paper; reference the same IDs in your tags. The two systems reinforce each other.

Block tags follow the core convention. Structural blocks use `[BLOCK]` / `[/BLOCK]` bracket syntax — identical to `[COMBAT]` / `[/COMBAT]` in the Combat Add-on. For analog notebooks, use `--- BLOCK ---` / `--- END BLOCK ---` dashed separators instead.

1. The Room Tag

The Room tag `[R:]` tracks a room's current state. It works like any other Lonelog persistent element: you tag it when something changes, update it as the situation evolves, and reference it later.

Format:

```
[R:ID|status:state|description|exits:DIR
ID, DIR ID]
```

Fields — the ID comes first; the rest can come in any order, because each one says what it is:

- `ID` — unique identifier for the room, typically a number (`1`, `2`, `3`) matching your map. The tag's `R` prefix is part of the name you say and write elsewhere: `[R:3]` is room **R3**, in exits, in prose and on a Maplog map.
- `level:` — optional, for multi-level dungeons: `[R:8|level:2]`
- `status:` — current exploration state (see below)
- description — optional brief label to jog your memory, written as a plain value
- `exits:` — optional connections to adjacent rooms

`status:` and `exits:` are categories (core §4.1): writing one again replaces what it held. That's what lets a room go from unexplored to cleared without ending up both.

1.1 Room Status

Status tells you the current state of a room at a glance. Use whatever terms fit your game:

- `[R:1|status:unexplored]` — Haven't entered yet
- `[R:1|status:active]` — Currently exploring
- `[R:1|status:cleared]` — Enemies defeated or threats resolved

- `[R:1|status:cleared, looted]` — Cleared and treasure taken
- `[R:1|status:locked]` — Can't get in yet
- `[R:1|status:trapped]` — Known hazard present
- `[R:1|status:safe]` — Verified clear, usable as refuge
- `[R:1|status:collapsed]` — No longer accessible

Combine statuses when it makes sense: `cleared, looted` tells you both that the fight is over *and* there's nothing left to grab.

Example — minimal:

```
[R:4|status:active]
```

Example — expanded:

```
S5 *North corridor*

@ Open the door carefully
? Is it locked?
d: d6=5 -> No
=> The door swings open easily.

[R:4|status:active|storage room, dusty
shelves|exits:S R2, E R5]

@ Search the shelves
d: Investigation d6=5 vs TN 4 -> Success
=> I find a rusty key behind a loose
brick.
(note: might open R5?)
```

```
? Are there enemies?
d: d6=6 -> No, and...
=> The room is completely undisturbed.
Years of dust.

[R:4|status:cleared, looted]
```

1.2 Updating Room Status

When a room's state changes, write the new status. It replaces the old one:

```
[R:1|status:unexplored]

... you enter and fight ...

[R:1|status:cleared]

... you search ...

[R:1|status:cleared, looted]
```

To add one state without rewriting the rest, put `+` inside the category; `-` takes one away:

```
[R:1|status:+looted]
```

1.3 Room Descriptions

The description field is optional and brief—just enough to jog your memory:

```
[R:1|status:cleared|guard room]
[R:2|status:active|library, candles still
lit]
[R:3|status:unexplored|sounds of water]
```

For richer descriptions, use regular Lonelog narrative after the tag:

```
[R:4|status:active|ritual chamber]
=> The air hums with energy. A circle of
runes glows faintly on the floor.
```

1.4 Reference Tags

Use # to reference a room you've already described, exactly as with NPCs and locations in core Lonelog:

The first mention establishes the room:

```
[R:3|status:cleared|armory|exits:N R2, W
R4]
```

Later, a reference points back to it:

```
[#R:3]
```

2. Room Connections

To record which rooms connect where, use the `exits:` category inside the tag. Each exit is a direction and a room, separated by a space:

```
[R:1|status:cleared|guard room|exits:N
R2, E R3, S R4]
```

Read this as: "Room 1 is cleared, it's a guard room, with exits north to Room 2, east to Room 3, and south to Room 4."

Directional shortcuts: `N`, `S`, `E`, `W`, `NE`, `NW`, `SE`, `SW`, `U` (up), `D` (down).

If direction doesn't matter, just list the rooms:

```
[R:5|status:unexplored|exits:R4, R6, R7]
```

You don't have to record exits at all. If you're keeping a visual map, the exits live there and you only need the Room tag for status. `exits:` is for when you want a fully text-based record, or when you discover a specific connection mid-play. Writing `exits:` again replaces the whole list, so add a single new exit with `+`:

```
=> I find a hidden passage behind the
bookshelf!
[R:3|exits:+E R7 secret]
```

Example — minimal:

```

```

```
[R:6|status:unexplored|exits:R5, R8]
```

Example — expanded:

```
@ Check the far wall for exits
d: Perception d6=6 vs TN 4 -> Success
=> A concealed door—well-disguised but
not locked.
[R:6|status:active|crypt|exits:W R5, N
R8, E R9 secret]
(note: update map with R9)
```

3. The Dungeon Status Block

When you sit down for a new session mid-dungeon, you need a snapshot: which rooms are cleared, where haven't you been, what's still locked. The **Dungeon Status Block** gives you that at a glance.

Format:

```
[DUNGEON STATUS]
[R:1|status:state|description]
[R:2|status:state|description]
...
[/DUNGEON STATUS]
```

When to open a block: At the top of a session when resuming a dungeon crawl, before your first scene.

When to update it: At session end, or at the start of the next session, to reflect what changed.

Example — digital:

```
[DUNGEON STATUS]
[R:1|status:cleared, looted|entry cave|
exits:N R2, E R3]
[R:2|status:cleared|guard room|exits:S
R1, W R5]
[R:3|status:active|library|exits:W R1]
[R:4|status:unexplored]
[R:5|status:locked|heavy door|exits:E R2]
[/DUNGEON STATUS]
```

Example — analog:

```
=== Session 3: The Orc Warren ===
[Date]        2025-10-15
[Duration]    1h30
[Recap]       Found the warren entrance.
Need to recover the stolen relic.

--- DUNGEON STATUS ---
R1: cleared, looted (entry cave)
R2: cleared (guard room)
R3: active (library)
R4: unexplored
R5: locked (heavy door)
--- END STATUS ---
```

This is a convenience, not a requirement. If your dungeon has three rooms, you don't need a status block—just tag rooms as you go. For larger dungeons, or when picking up a session after a break, it's a quick way to orient yourself.

4. Room Tags in Play

4.1 Rooms as Scene Headers

Use a Room tag directly in your scene header for compact context:

```
S5 [R:4|status:active] *Storage room,
dusty shelves*
```

This tells you exactly where the scene takes place without a separate tag line. Some players prefer this to separate scene descriptions and room tags.

4.2 Generated Dungeons

If you're generating the dungeon as you go (using random tables or an oracle), record the generation alongside the Room tag:

```
@ Open the east door
tbl: d20=14 -> "Large room, two exits"

? What's inside?
tbl: d100=67 -> "Library with a trapped
chest"

[R:6|status:active|library, trapped
chest|exits:W R3, N R7]
```

The generation rolls tell you *how* you discovered the room. The Room tag tells you *what it is now*.

4.3 Maps and Lonelog

Let's be direct: **a visual map is almost always better than text for spatial relationships.** Graph paper, a digital tool, a quick

sketch on a napkin—whatever works. The Room tag's job isn't to replace your map. It's to track state that a map handles poorly: whether a room is cleared, what you found there, how its status changed over play.

The recommended split:

- **Map** handles layout, spatial relationships, and navigation
- **Room tags** handle state, status changes, and narrative context
- **Core Lonelog** handles everything else (actions, rolls, consequences, story)

Mark room numbers on your map. Use those same numbers in your Room tags. Your map and your log now reference each other seamlessly:

```
S7 [R:4] *Checking the altar*

@ Search the altar
d: Investigation d6=6 vs TN 4 -> Success
=> Secret compartment! [Thread:Cult
Ritual|Open]
[R:4|status:cleared, looted]

(note: add the altar to map)
```

If you want a **fully text-based** dungeon log with no separate map, `exits:` lets you reconstruct the layout from your notes. But this is the exception, not the recommendation.

4.4 Exploration Turns

Many dungeon games run on exploration turns: each turn uses up light, and every so often something wanders by. Core Lonelog covers all of it, with nothing new to learn:

- **The turn number** goes in the scene context: `S4 *R3, crypt, turn 3*`.
- **Light** is a timer: `[Timer:Torch 6]` when you light it, `[Timer:Torch-1]` each turn.
- **Wandering monster checks** are verdicts: `d: Wandering d6=5 -> No`.
- **Long counts** that matter, such as turns until the dungeon reacts, are clocks: `[Clock:Alert+1]`.

```
S4 *R3, crypt, turn 3*
[Timer:Torch-1]
d: Wandering d6=1 -> Yes
tbl: Wandering Monster d8=4 -> Giant rats
```

If you also use the Resource Tracking Add-on, a torch burning out is `[Inv:Torch|×-1]`.

Cross-Add-on Interactions

- Combat breaks out in a room — Dungeon Crawling + Combat — `[R:]`, `[COMBAT]`, `[F:]`
- Tracking torches and rations per room — Dungeon Crawling + Resource Tracking — `[R:]`, `[Inv:Name|#]`
- Combat while tracking resources — All three add-ons — `[R:]`, `[COMBAT]`, `[Inv:]`

Combined example — Dungeon Crawling + Combat:

```
S6 *Approaching the barracks*

@ Move quietly down the corridor
d: Stealth d6=4 vs TN 4 -> Success
=> I creep forward unseen.

[R:5|status:active|barracks, stench of
sweat|exits:S R3, W R6, N R7]

? How many guards?
tbl: d6=3 -> "Three enemies"
=> Two eating, one sharpening a blade.

[COMBAT]
[PC:Alex|HP 12|pos:Engaged]
[F:Guard 1|HP 5|pos:Close|eating]
[F:Guard 2|HP 5|pos:Close|eating]
[F:Guard 3|HP 5|pos:Close|alert]

Rd1
@ Ambush the alert guard first
d: d20+4=18 vs AC 12 -> Hit, 1d8=7
=> [F:Guard 3|dead]

@(Guard 1) Grabs weapon and charges
d: d20+3=11 vs AC 14 -> Miss

Rd2
@ Sweep at both remaining guards
d: d20+4=15 vs AC 12 -> Hit, 1d8=5 =>
[F:Guard 1|dead]
```

```
d: d20+4=12 vs AC 12 -> Hit, 1d8=3 =>
[F:Guard 2|dead]

[/COMBAT]
=> Room clear before they could raise an
alarm.
[R:5|status:cleared]

@ Search the barracks
tbl: d20=8 -> "Rations and a crude map"
=> A rough map scratched on hide. Shows
rooms to the west!
[R:5|status:cleared, looted]
[R:6|status:unexplored|marked on orc map]
(note: update my map with R6)
```

System Adaptations

OSR Systems (Old School Essentials, Basic/Expert D&D)

Classic dungeon crawl procedures map naturally to this add-on. Turn-based exploration, wandering monster checks, and the six-minute turn structure all generate natural tag update points. Use Room tags at the end of each turn or when something significant changes.

```
[DUNGEON STATUS]
[R:1|status:cleared, looted|entry hall|
exits:N R2, E R3]
[R:2|status:cleared|guard room|exits:S
R1, W R5]
[R:3|status:active|crypt|exits:W R1, D
R8]
[/DUNGEON STATUS]

S4 *R3, crypt, turn 3*
[Timer:Torch-1]

@ Search the sarcophagi
d: Secret door 1-in-6 d6=2 -> None found
d: Wandering d6=5 -> No

@ Light the wall sconces and examine the
inscriptions
d: INT check d20=8 vs DC 12 -> Fail
=> The script is too archaic—I can't make
```

```
sense of it.
[R:3|status:cleared]
```

Ironsworn / Starforged

Ironsworn's Delve mechanic uses themed sectors rather than numbered rooms. Map sectors to Room IDs and use the `Delve the Depths` move result to determine what you find. Progress tracks work naturally alongside Room status.

```
[Track:Iron Barrow 2/10]

[DUNGEON STATUS]
[R:1|status:cleared|Threshold — rusted
gate|exits:N R2]
[R:2|status:active|Shadow — collapsed
corridor]
[/DUNGEON STATUS]

@ Delve the Depths (Shadow theme)
d: d6+2=7 vs d10=4, d10=9 -> Weak Hit
tbl: Shadow Feature d100=44 -> "Symbols
of a forsaken god"
=> I find ritual markings. Progress, but
at a cost.
[Track:Iron Barrow+2]
[R:2|status:cleared|Shadow — forsaken
markings|exits:N R3]
[R:3|status:unexplored]
[Thread:The Forsaken God|Open]
```

Dungeon World (PbtA)

Dungeon World generates dungeon content through fictional framing and GM moves rather than room-by-room procedure. Use Room tags loosely—tag significant locations as you discover them, without worrying about exhaustive status tracking.

```
@ Discern Realities as I enter the
chamber
d: 2d6+1=10 -> 10+ Hit: ask three
questions

? What here is useful or valuable?
-> A sealed chest on the altar dais.
? What is about to happen?
-> The shadows in the corner are moving.
? Who made this place?
-> Cult sigils everywhere—same as the
ones on the road.

[R:4|status:active|ritual chamber, living
shadows]
[Thread:The Cult's Purpose|Open]
```

Best Practices

Do: Use Room tags for state, your map for space

```
✔ [R:4|status:cleared, looted|exits:N R7]
  (note: add the door to R7 on map)
```

Don't: Try to replace your map with exit chains

A chain of exits is harder to navigate than a sketch:

```
✗ [R:1|status:cleared|exits:N R2, E R3]
  [R:2|status:cleared|exits:S R1, N R4, E
R5]
  [R:3|status:active|exits:W R1, N R6, E
R7, SE R8]

✔ Draw the map. Use Room tags only for
state.
```

Do: Update status immediately when it changes

```
✔ @ Search the room after the fight
  d: Investigation d6=5 vs TN 4 ->
Success
  => I find a key and some coin.
  [R:3|status:cleared, looted]
```

Don't: Leave status ambiguous and reconstruct later

```
✗ (Three scenes of exploration with no
tag updates)
  (note: I think R3 and R4 are cleared?
need to check)

✔ Tag each change as it happens.
```

Do: Open a Dungeon Status Block when resuming a session

```
✔ === Session 4 ===
  [DUNGEON STATUS]
  [R:1|status:cleared, looted|entry]
  [R:2|status:cleared|barracks]
  [R:3|status:unexplored]
  [R:4|status:locked|iron door]
  [/DUNGEON STATUS]
```

Don't: Reconstruct dungeon state by scanning the whole log

```
✗ (Scrolling back through two sessions of
notes to figure out
   which rooms are cleared)

✔ Write the status block at session end.
Start the next session
  with a current picture.
```

Do: Keep descriptions brief in the tag, expand in prose

```
✔ [R:7|status:active|throne room]
  => The ceiling is forty feet high. A
cracked obsidian throne
  dominates the far wall. Something has
nested in it recently.
```

Don't: Overload the tag with narrative detail

```
✗ [R:7|active|massive vaulted throne
room, cracked obsidian throne
   on a raised dais, old nesting material
and bones, forty-foot ceiling,
   faded tapestries depicting a forgotten
dynasty, cold draft from the north]
```

Quick Reference

New Tags

- `[R:ID|status:state]` — Room with status only. Example: `[R:3|status:cleared]`
- `[R:ID|status:state|desc]` — Room with description. Example: `[R:3|status:cleared|library]`
- `[R:ID|status:state|desc|exits:DIR ID]` — Room with connections. Example: `[R:3|status:cleared|library|exits:W R1]`
- `[R:ID|status:+state]` — Add one status without replacing the others. Example: `[R:3|status:+looted]`
- `[#R:ID]` — Reference an established room. Example: `[#R:3]`

Status Terms

Status	Meaning
`unexplored`	Haven't entered yet
`active`	Currently exploring
`cleared`	Threats resolved
`cleared, looted`	Cleared and searched
`locked`	Cannot enter yet
`trapped`	Known hazard present
`safe`	Verified clear, usable as refuge
`collapsed`	No longer accessible

Structural Blocks

Block	Opens When	Closes When
`[DUNGEON STATUS]` / `[/DUNGEON STATUS]`	Session start when resuming a dungeon	`[/DUNGEON STATUS]` after the last room; write a fresh block next session

Directional Shortcuts

`N` `S` `E` `W` `NE` `NW` `SE` `SW` `U` (up) `D` (down)

Complete Example

```
[DUNGEON STATUS]
[R:1|status:cleared, looted|entry cave|
exits:N R2, E R3]
[R:2|status:unexplored] [R:3|
status:unexplored]
[/DUNGEON STATUS]

S7 *North passage*

@ Approach R2 quietly
d: Stealth d6=4 vs TN 4 -> Success

[R:2|status:active|barracks|exits:S R1, W
R4]
? Enemies?
d: d6=2 -> Yes
=> Two guards, armed.
```

```
...combat...
[R:2|status:cleared, looted]
[R:4|status:unexplored|marked on guard's
map]
```

FAQ

Q: Do I need a separate map, or can I use exits to track layout? A: A visual map is almost always better for layout and navigation. Use exits in your Room tags only when you want a fully text-based record, or when you discover a specific connection (like a secret door) that's worth noting in the log. For everything else, draw the map.

Q: What Room ID system should I use? A: Numbers are simplest: `[R:1]`, `[R:2]`, `[R:3]` are rooms R1, R2, R3, matching a numbered map. Sub-rooms can take letters (`[R:1A]`, `[R:1B]`). For several levels, keep the numbers and add a `level:` category, `[R:8|level:2]`, which matches the level tags (`L2`) Maplog writes on a map. Use whatever helps you connect tags to your map at a glance.

Q: Can I use this alongside the Combat Add-on? A: Yes—they're designed to work together. When combat breaks out in a room, open a `[COMBAT]` block inside the scene. When it ends, close it and update the Room tag status. The Cross-Add-on Interactions section has a full example.

Q: What if I'm using a published dungeon with its own room numbering? A: Use the published room numbers as your IDs. `[R:12|status:cleared|Goblin Warrens p.4]` works perfectly—the published map handles layout, the tag handles state.

Q: How does the Dungeon Status Block work in an analog notebook? A: Write it with dashed separators: `--- DUNGEON STATUS ---`. List one room per line with status and a brief label. Cross out or update entries as rooms change. At the start of each new session, write a fresh status block with current information rather than patching the old one.

Q: Do I need to tag every room I pass through? A: Only rooms where something worth tracking happens. If you walk through an empty corridor, you don't need a tag—just describe it in prose. Tag

rooms when their status has meaning: when you clear them, find something, lock them, or need to return later.

Q: What about dungeon features that aren't rooms—corridors, traps, doors? A: Use core Lonelog notation. A trap that springs or a door that gives way is an event (`[E:Trap Sprung]`); otherwise, keep them as temporary context in prose. Corridors usually don't need persistent tracking. The Room tag is specifically for spaces with ongoing state that matters across multiple scenes or sessions.

Credits & License

© 2025 Roberto Bisceglie

This add-on extends Lonelog by Roberto Bisceglie.

Written to address the request of u/Jollto13 on r/Solo_Roleplaying.

Version History:

- v 1.2.0: `[R:3]` is room R3 everywhere; `level:` category for multi-level dungeons; new §4.4 on exploration turns with core notation (timers, wandering checks); Dungeon World answers use `->`. Aligned with Lonelog v1.7: `status:` and `exits:` are categories, so a new status replaces the old one and fields can come in any order; exits are written `exits:N R2, E R7 secret`; `[E:]` marks events that happened; table result format; standard `[Track:]` counters; explanations moved out of code blocks.
- v 1.1.1: Map reminders note what to draw, not room state; state stays in the Room tag
- v 1.1.0: Block tags unified with `[BLOCK]`/`[/BLOCK]` convention; design principles updated
- v 1.0.0: Rewritten as a compliant add-on (previously "Dungeon Crawling Module")

This work is licensed under the **Creative Commons Attribution-ShareAlike 4.0 International License**.

You are free to share and adapt this material, provided you give appropriate credit and distribute adaptations under the same license. Session logs and play records created using this add-on's notation are your own work and are not subject to this license.