

PARA + Daily Task Management System (Complete Guide)

File Structure

Single text file: `~/home/tasks.txt` with flexible, hierarchical organization:

```
# DAILY
## 15-01-2025
- [x] completed tasks
- [~] progressed tasks
- [ ] not started tasks

# MAIN
## INBOX
- [ ] unsorted tasks #001
## PROJECTS
### Website Redesign
- [ ] project tasks #002
### Marketing Campaign
- [ ] more project tasks #003
## AREAS
### Health
- [ ] ongoing responsibilities #004
## RESOURCES
- [ ] reference materials #005
## ZETTELKASTEN
- [ ] knowledge development #006

# ARCHIVE
## ARCHIVED COMPLETED TASKS
## ARCHIVED DAILY SECTIONS
```

Task Syntax

```
markdown

- [ ] incomplete task @15-01-2025 #001
- [x] completed task @15-01-2025 #002
- [ ] snoozed task @15-01-2025 snooze:20-01-2025 #003
- [ ] recurring task @15-01-2025 (daily) #004
- [ ] deadline task @15-01-2025 due:25-01-2025 #005
```

Daily Progress Tracking

The key innovation of this system is distinguishing between completion and progress:

In Daily Sections

- ☒ = **Completed**: Task is finished, will mark main task complete when synced
- ☐ ~ = **Progress**: Made meaningful progress but task not done, updates engagement date only
- ☐ = **Not started**: No work done on this task yet

Examples

markdown

15-01-2025

- ☒ morning workout (from: Health) #004 # Done for today
- ☐ ~ write chapter 3 (from: PROJECTS) #023 # Worked on it, not finished
- ☒ call dentist (from: INBOX) #045 # Completed, can mark done
- ☐ review budget (from: AREAS) #067 # Haven't started yet

Sync Behavior

When you run `tasks sync`:

- **Completed (☒)**:
 - Non-recurring tasks: Main task becomes ☒
 - Recurring tasks: Main task stays ☐ , date updates
- **Progress (☐ ~)**:
 - All tasks: Main task stays ☐ , date updates to show recent engagement
- **Not started (☐)**: No changes to main task

Before Sync (Main List)

markdown

- ☐ write chapter 3 @10-01-2025 #023
- ☐ morning workout @14-01-2025 (daily) #004

After Working (Daily Section)

markdown

- [~] write chapter 3 (from: PROJECTS) #023 # Made progress
- [x] morning workout (from: Health) #004 # Completed today

After Sync (Main List)

markdown

- [] write chapter 3 @15-01-2025 #023 # Still incomplete, date updated
- [] morning workout @15-01-2025 (daily) #004 # Recurring, date updated

Hierarchical Organization

- **Level 1 (#):** Major sections (DAILY, MAIN, ARCHIVE)
- **Level 2 (##):** Subsections (INBOX, PROJECTS, date sections)
- **Level 3 (###):** Project names, area categories
- **Order-agnostic:** Script finds sections by headers, not position
- **Flexible:** Can reorganize sections without breaking functionality

ID System

- **Simple sequential:** #001, #002, #003, etc.
- **Auto-generated:** Finds highest existing ID and increments
- **Cross-references:** Daily tasks reference main tasks by same ID
- **Global scope:** IDs are unique across entire file

Date System

- @date = last engagement/completion (for staleness tracking)
- snooze:date = hide from stale reports until this date
- due:date = external deadline (optional)
- Recurrence: (daily), (weekdays), (weekly:sun), (monthly:1st)

Daily Workflow

1. **Morning:** (tasks daily) - auto-adds recurring tasks from MAIN section
2. **Work:** Use daily section, mark tasks as you work:
 - [x] for completed tasks
 - [~] for tasks you made progress on but didn't finish

3. **Evening:** `tasks sync` - updates MAIN list from daily progress

4. **Planning:** `tasks stale` - see what needs attention

Key Commands

bash

Daily workflow

`tasks daily` *# Add today's section with recurring tasks*

`tasks stale` *# Show oldest tasks from MAIN section*

`tasks sync` *# Update MAIN from daily progress*

Task management

`tasks complete 042` *# Mark task #042 as complete in main list*

`tasks add-main "text"` *# Add to INBOX*

`tasks add-main "text" PROJECTS` *# Add to specific section*

`tasks add-main "fix sink" PROJECTS:HOME` *# Add to subsection*

`tasks add-daily "text"` *# Add to today's section*

Organization

`tasks snooze 042 3` *# Hide task #042 for 3 days*

`tasks snooze 042 25-12-2025` *# Hide until specific date*

`tasks show 042` *# Show task details*

`tasks sections` *# List all available sections*

`tasks archive` *# Clean up old content*

`tasks help` *# Show detailed help*

Recurring Patterns

markdown

Basic patterns

(daily) Every day

(weekdays) Monday-Friday only

Weekly patterns

(weekly) Every Sunday (default)

(weekly:tue) Every Tuesday

(weekly:mon,wed,fri) Multiple days per week

Monthly patterns

(monthly) 1st of month (default)

(monthly:15th) 15th of every month

General recurrence patterns

(recur:3d) Every 3 days

(recur:2w) Every 2 weeks

(recur:1m) Every month

(recur:6m) Every 6 months

(recur:1y) Every year

(recur:1y,3m) Every 1 year and 3 months (combined intervals)

Time units: d=days, w=weeks, m=months, y=years

Core Concepts

Projects vs Areas

- **Projects:** Have endpoints, get archived when complete
- **Areas:** Ongoing responsibilities, never "done"
- **Both:** Can have tasks that benefit from progress tracking

Recurring vs Progress

- **Recurring tasks:** You know ahead of time they'll never be "finished" (exercise, review)
- **Progress tasks:** You intend to complete them, but work happens incrementally (write chapter, build feature)

Staleness Tracking

- Only tracks tasks in MAIN section (ignores DAILY/ARCHIVE)
- Shows days since last `@date` update
- Ignores future-snoozed tasks

- Progress updates ([~]) prevent tasks from appearing stale

The Power of Progress Tracking

This system solves a key problem: **How do you handle tasks where you make meaningful progress but don't finish?**

Without progress tracking:

- Mark incomplete: Task appears stale even though you worked on it
- Mark complete: Dishonest, task isn't actually done

With progress tracking:

- Mark ([~]): Shows engagement, keeps task active, maintains honesty
- Removes pressure to artificially complete tasks
- Encourages regular engagement over forced completion

Benefits

- **Visual clarity:** Today's work first, organized reference below
- **Flexible organization:** Rearrange sections without breaking scripts
- **Honest progress:** Distinguish between engagement and completion
- **Robust automation:** ID-based commands, hierarchical parsing
- **Scalable structure:** Add projects/areas as needed
- **Plain text:** Portable, future-proof, tool-agnostic
- **PARA methodology:** Organized by actionability, not category

Example Day

```
bash
```

Morning

tasks daily

Creates today's section with recurring tasks

During work - edit daily section manually

- [x] morning workout (from: Health) #004
- [~] write blog post (from: PROJECTS) #023 *# Made progress, not done*
- [x] call client (from: INBOX) #045 *# Completed*
- [] review budget (from: AREAS) #067 *# Didn't get to it*

Evening

tasks sync

Updates main list

Result: #004 and #045 update dates, #023 stays incomplete but date updates

Weekly planning

tasks stale

See what's been neglected

tasks archive

Clean up old content

This system provides structure without rigidity, automation without lock-in, and honest progress tracking that matches how complex work actually gets done.