

# **TASK** **MANAGEMENT** **FOR** **PROJECT** **MANAGERS**

**BRENDAN TONER**

**DONEBEFOREBREKKY.COM**

# TASK MANAGEMENT FOR PROJECT MANAGERS

How to bridge the gap between long term planning activities and what one does at 8:20am on Tuesday morning? How can project planning that stretches over months or years remain relevant and account for new ad-hoc activities that arrive every day? In this post covering effective Task Management for Project Managers, I will explain what I have been successfully using for the last few years to bridge this chasm.

## Table of Contents

[Introduction](#)

[Task Management Methods \(Waterfall, SCRUM\)](#)

[Project Management Takeaway from Getting Things Done \(GTD\)](#)

[Macro to Micro Task Management System](#)

[Managing medium term task management](#)

[Software for Long Term and Daily Project Management](#)

[How to Implement Macro to Micro Project Planning](#)

[Managing daily ad-hoc task entry into project plans](#)

[Tracking Progress of a project with Abstractspoon TodoList](#)

# TASK MANAGEMENT FOR PROJECT MANAGEMENT INTRODUCTION



Generally task management falls into two camps, project management which may span months/years and day to day task management. Neither is particularly appropriate to represent the other. Project planning has little concern about hour long tasks and daily todo lists certainly cannot forecast if your present hour long task will mean completing the project on time.

Yet, frequently people find themselves in the position where we must juggle both. This may mean dodging in and out of different systems, glancing/reviewing project plans and taking tasks from one system to another. It can be both complex and time consuming and lead to frequent contemplation of project plan tasks and their relation to present actions

I think there is an elegant way to handle this using a single system. The features of this system are the following,

- It combines project tasks with ad-hoc daily items under a single framework

- Since project/daily tasks are under one system, daily/weekly review of project lists can be removed
- The basis of scheduled activities is a project plan so important projects tasks are covered by default
- It allows for taking project plan tasks and breaking them down to actionable tasks that are conducive to daily work
- It is quick to both enter new tasks and determine which tasks to execute

To get such advantages we must utilise the strengths of different methods of task management. We will merge three different approaches to create a unified approach for task management that spans project activities spanning years to daily tasks in the order of minutes.

## **TASK MANAGEMENT METHODS (WATERFALL, SCRUM)**

The three approaches that we will cover in this article are listed below. Each method has application over different timeframes and by combining these we can construct a system to span a timeframe from days to years.

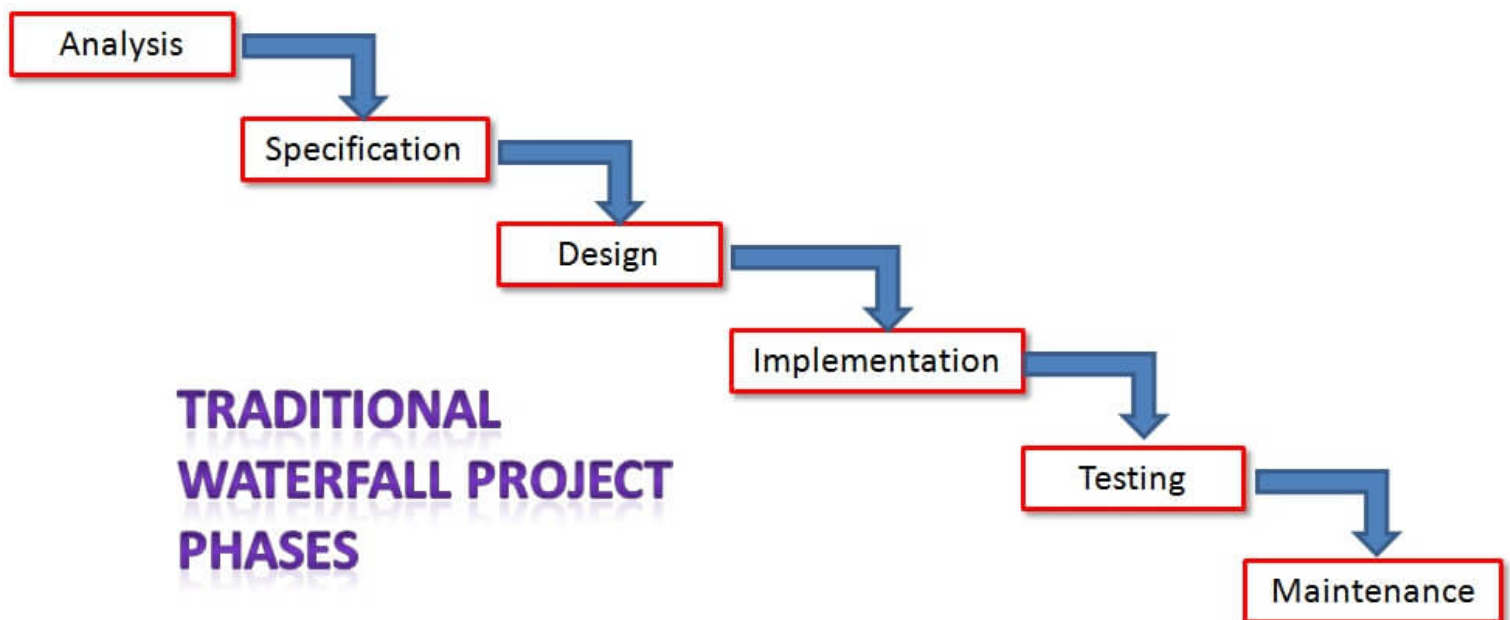
- Waterfall – Long term sequential plans over months/years
- SCRUM – Short term iterative sprints over a few weeks
- Simplified Getting Things Done – Daily task management

Let's go through each in turn, detail their approach and what will be extracted from each method.

## **WATERFALL PROJECT PLANNING**

This is a sequential approach to project execution and involves several phases. The stereotypical ones are shown below and often at the end of each phases a milestone is achieved and review meetings held before progressing with the next phase. This type of planning lends itself nicely to traditional Gantt chart tools and with that various other advantages such as,

- Resource planning
- Determining serial/parallel tasks
- Critical path analysis
- Linking of tasks/phases for long term forecasting of completion
- Systematic entry/exit criteria for each phase
- A transparent overview of project activities
- Tracking of activities vs. baseline planning



My takeaway from this approach is this: the ability to sketch out medium and long term plans to account for the major project activities, the big rocks so to speak. Any general resource or task conflicts can also be highlighted from this. So, the waterfall approach planned with a Gantt chart sets the medium and long term landscape for my activities.

## SCRUM: USEFUL ASPECTS FOR LONG TERM PROJECT PLANNING

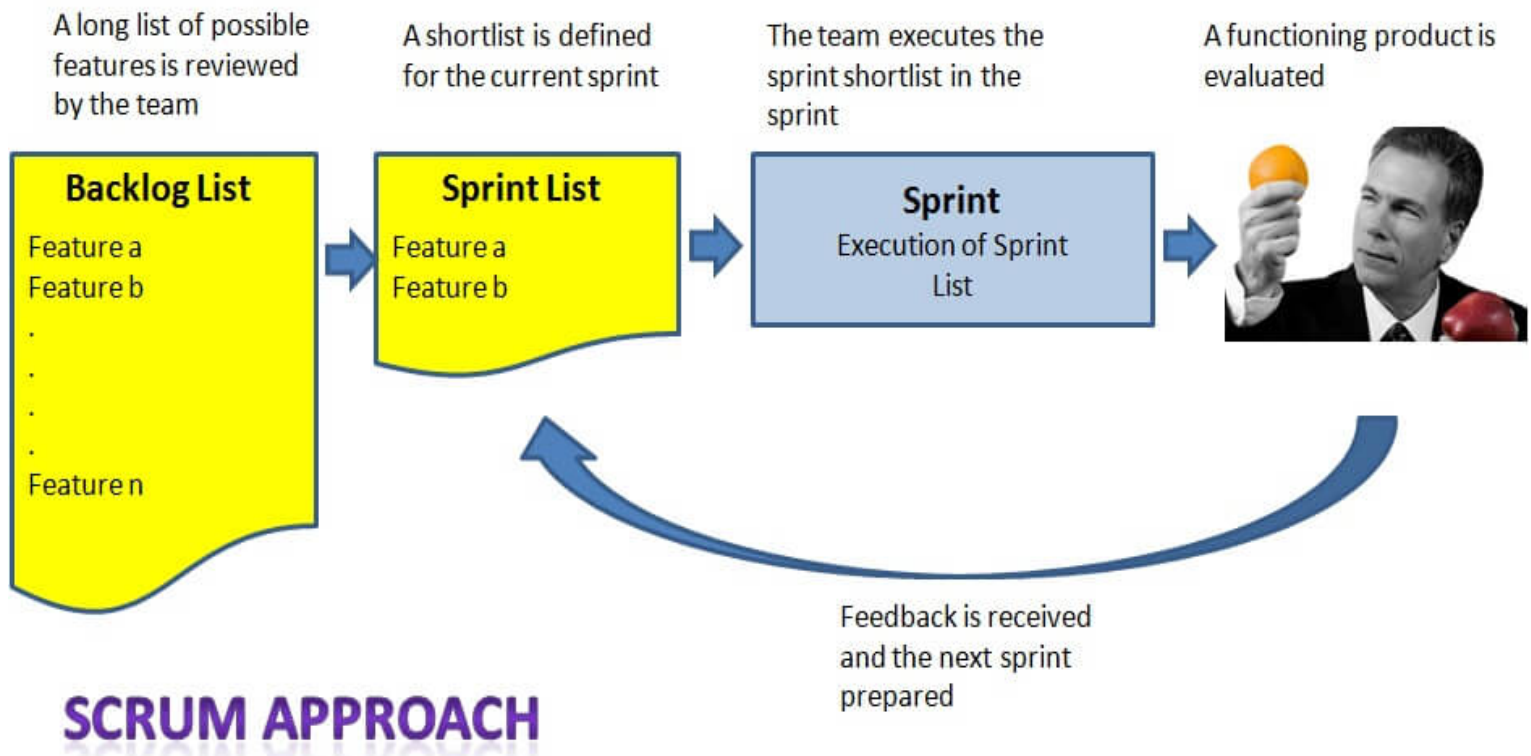
There are several criticisms often levelled at the waterfall approach such as,

- The product is only delivered at the end of the project
- Requirements must be known at the start of the development
- Changes in customer requirements are difficult to implement possible causing a reversion to earlier project stages
- Perhaps less suited to projects requiring intensive collaborative efforts and customer involvement

To counter such points, especially in the field of software development different approaches have been taken. One of which is SCRUM. Inspired by it's sporting title, SCRUM is an iterative development model in which small collaborative teams produce working product at the end of short, intensive development activities. These short development phases are known as sprints. There are a few characteristics of SCRUM development including,

- Appointment of a SCRUM master who keeps the team aligned and focussed on the goals of the sprint.
- A "Backlog" is a list of activities from which the team select which tasks to be handled in each sprint.
- Each sprint is intensive and usually a short time period, typically a few weeks.
- Daily meetings are held to keep the team aligned
- At the end of each sprint, a review meeting is held.
- Following the review, improvements are noted, a shortlist of backlog items are identified for the next sprint and the process is repeated.

This development cycle can be represented as follows,



Although SCRUM is usually considered in terms of software development there is no need to have this restriction and indeed several of it's philosophies can be applied to general project activities. The points that I take away from SCRUM and apply to my own method are the following,

- The concept of short, intensive activities on a defined list of items
- That these items are shortlisted from a reservoir of tasks (backlog). In my case the source of the activities is the big picture project plan.
- That each short term sprint of a few weeks is defined and planned at the beginning with minimal changes as it proceeds.

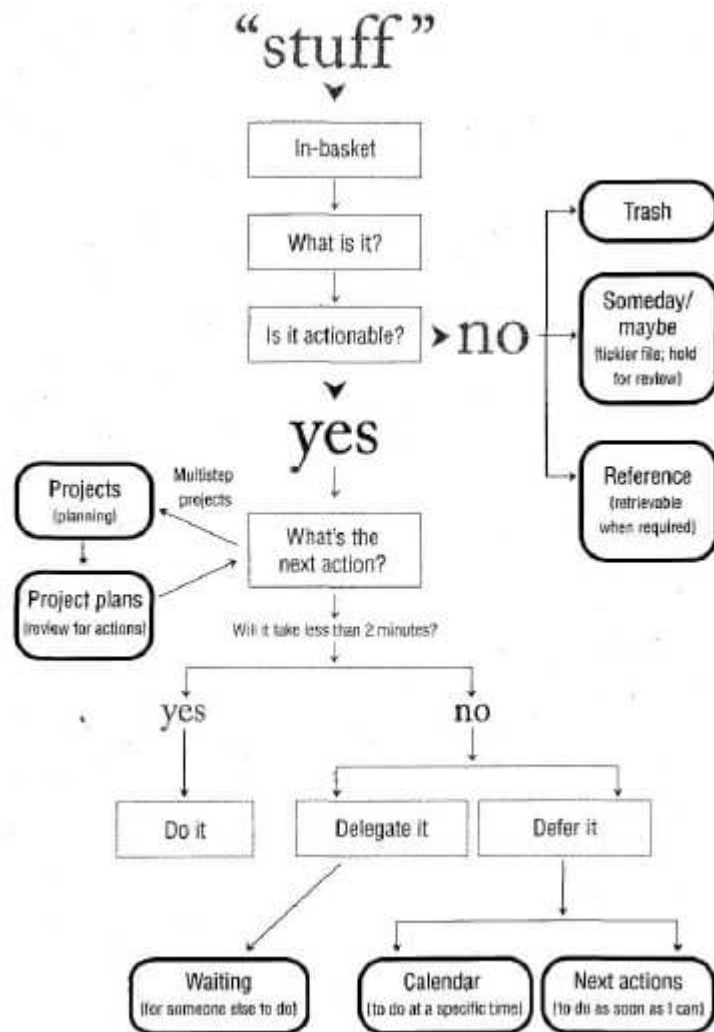
## PROJECT MANAGEMENT TAKEAWAY FROM GETTING THINGS DONE (GTD)

Getting Things Done (GTD) is a popular task management system from David Allen. A complete book was written on the topic and so I can only but summarise a couple of key items that are directly relevant for what I want to take into my task management system.

1. A fundamental point that David covers is to have a system that will cover all of your tasks. One must do a “brain dump”, to enter all these tasks into the appropriate lists. This should leave an uncluttered mind allowing focus to be placed on the task at hand.
2. How to organise tasks. As the saying goes, a picture is worth a thousand words and this is also true of how the tasks are to be handled. David provides a flow diagram for how each task is to be filed for future execution, see below. At the top we have an Inbox where all tasks are entered until time permits to organise and process them. When processing, each task is taken in turn and put into one of several lists
  1. Do – For immediate attention
  2. Defer – Scheduled date
  3. Someday – Distant goals
  4. Delegate – Waiting on activities from another person
  5. Reference – Non actionable information

Project lists are also available for review and next actions from those considered. In this context, a project is defined as a task requiring more than one step.





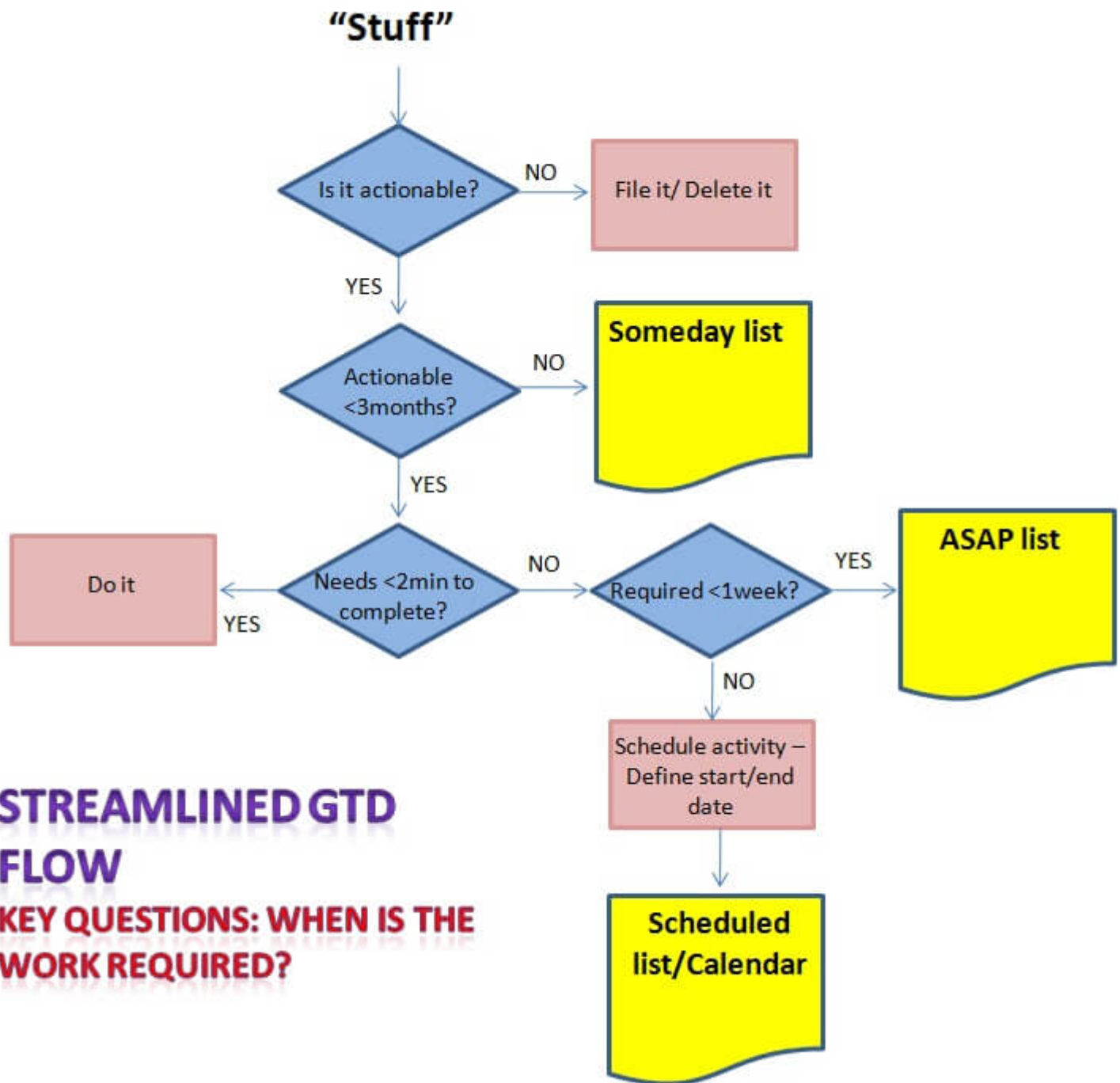
When it comes to executing tasks there are several methods available under the GTD method. Some examples of this include,

- Based on the context. The context refers to under what situation the task must be done. Examples would be at work or at home.
- Based on time available. So how much time do you have and what task fits to that time.
- Based on energy available. Aligning the most difficult tasks to times in the day when we can best handle them.

There are quite a few aspects of GTD which I also do not apply as they are not needed for my work. These points are noted below.

1. Inbox List. The system I present is so simple the processing can be done at the point of task receipt.
2. Reference List. When I reference something it is normally in the same ecosystem as the information. For example, I archive emails in my email program. So, I won't take them out of there to put in a reference list.
3. Projects. Defined in GTD as a task requiring more than 1 step to complete it. I don't need a separate list for this. Should a task require more than one step I chunk it into subtasks in whichever list it resides.
4. Delegate list. Any tasks that are delegated reside within the existing lists. I simply have a column indicating the person responsible. Having that column eliminates the Delegate list.
5. Methods for selecting which task to do next.
6. Any paper filing or tickler files. Everything I deal with is electronic.
7. Weekly review. Many of the items David covers in this review I find are covered during normal task entry, task selection and completion.

So, when I consider David Allen's workflow and apply the points above I end up with a different model, [Simple Getting Things Done](#), as shown below. Some may say this is not GTD or loosely influenced by GTD. Whichever the case GTD was the starting point and so I still attribute the idea to [David Allen's original work](#).



#### GTD Takeaway

The key items that I have learned and take away from David Allen's system are the following,

- One task management system to rule them all!
- The idea of different lists dependent on when the task is to be executed

- The concept of the next task. This should always be at the forefront of one's mind to ensure progress towards fulfilling an objective.

## MACRO TO MICRO TASK MANAGEMENT SYSTEM

Earlier in this post, I covered the various systems that provide the tools for task management ranging across several years down to tasks covering mere minutes. The takeaway elements for each were described. Now it is time to assemble those takeaway parts into a single, cohesive system that can operate in the real world and enable both long term and daily activities to be easily handled.

To avoid being overwhelmed I introduce this system bit by bit going through the following areas. Each will be accompanied by flow diagrams to explain the work flow.

- Entry of medium term project sub-phases
- Daily ad-hoc task entry
- Overall system flow diagram
- Task execution

## MANAGING MEDIUM TERM TASK MANAGEMENT

Let's start with how we go from long term waterfall planning to daily scheduled tasks. This is loosely following the idea from SCRUM to pull tasks from the backlog to apply to the current sprint.

This is shown in the diagram below. What I do here is to take a sub-phase of the project and plan that phase in depth. Normally these sub-phases span three or four months and as with SCRUM they are periods of intensive, collaborative team activity. This planning still has elements of traditional project planning though with consideration of dates, resources, conflicts etc. Each task entered needs the following attributes,

- Task title
- Responsibility
- Start and due dates

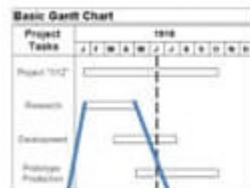
Task dependencies will have been dealt with in determining the start/due dates so are not a must in this case. One point to mention in this case. In long term project planning, the maintenance of task dependencies is a must as changes in project tasks can have a knock on effect. However, in short term plans/sprints, changes to activities are avoided or minimised, hence negating the need for task dependency.

This mapping out of the “sprint”, activity is done using a suitable tool such as MS Project. When complete it can then be exported to my task list and those tasks form the backbone of my tasks over that time period – typically 3 to 4 months. This act alone enables two important task features to be addressed

- Alignment with our top level goals
- Ensuring that the big rocks (important tasks) are covered in our work scheduling

Since we plan for the next 3 months or so it is only necessary to go through this process 4 times per year. Depending on your work it may be more or less but in general it is good to be able to align this timeframe with project sub-phases. These sub-phases should also have defined start dates and expected outcomes or results.

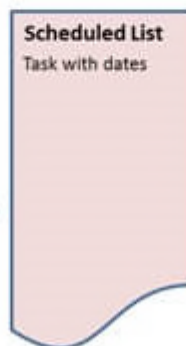
## Important Tasks from Top level objectives



A 3 month "sprint" list is defined and planned from main objectives. This includes dates.



**MIGRATE LONG TERM PROJECT  
PLANNING/OBJECTIVES TO A  
SHORT TERM SPRINT LIST  
(SCHEDULED LIST)**



## MANAGING DAILY AD-HOC TASK ENTRY INTO PROJECT PLANS

The previous section dealt with the creation of foreseeable tasks over the following three months covering the period of the sub-phase. In practice though, those will not be the only tasks that must be done. Every day, new tasks will emerge and these also need to be considered and entered into the system. It is these ad-hoc tasks that we now deal with.

For these tasks, I have assigned three lists,

ASAP List: For tasks that need to be done in the next week

Someday: Only need to be done after 3 months

Scheduled : Not this week but sometime over the next three months

On receipt of a new task there is only one question to be answered –

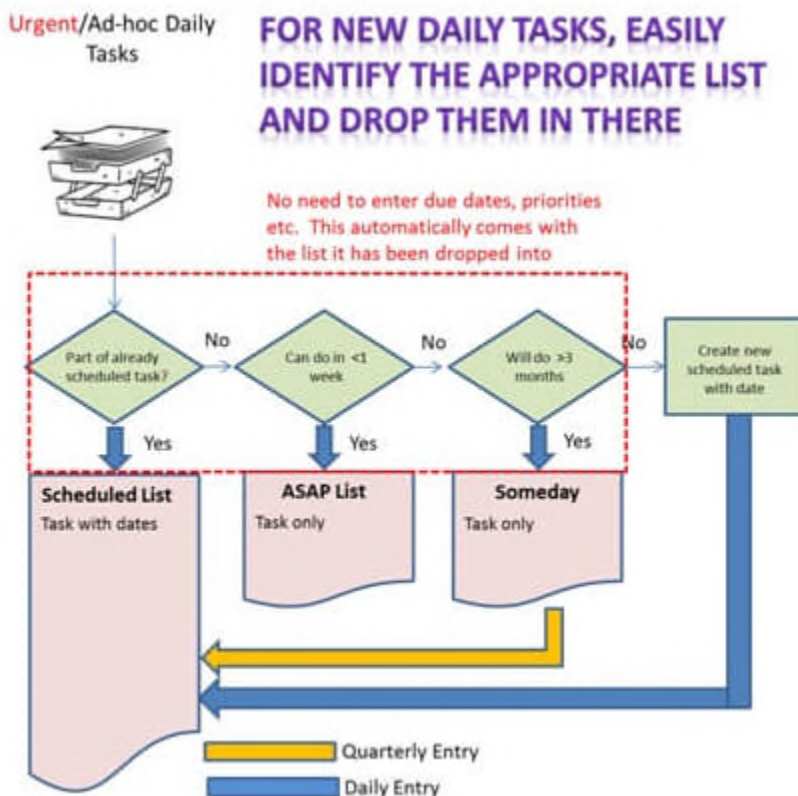
### When is the task needed?

The answer to this will directly determine which list the task goes into. Depending on the list, the information required for the task will change a little,

ASAP & Someday List: Enter task title and responsibility. Start/due dates NOT required

Scheduled List: Title, responsibility, start and due dates are required

This entering of tasks into the various lists can be represented in terms of a flow diagram, see below.



## OVERALL FLOW DIAGRAM FOR PLANNED AND AD-HOC TASKS

Pulling together all of the ideas presented earlier results in the following flow diagram. As described previously we can see this overall diagram in two sections.

Left side :Medium term planning where project sub phases are planned out and entered into our scheduled list roughly on a quarterly basis. We can also categorise these tasks as important and in alignment with longer term goals and objectives. These are the tasks that truly add to one's productivity.

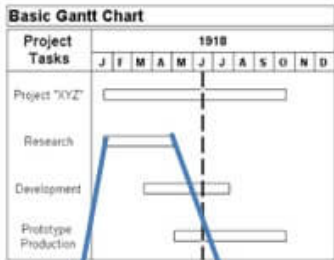
Right side : Daily addition of new tasks into the system. These go into one of the three lists, depending on when the work is to be done. For tasks going into the ASAP or to a lesser extent the Scheduled list, we can say they are so called urgent tasks.



Important Tasks from  
Top level objectives

Urgent/Ad-hoc Daily  
Tasks

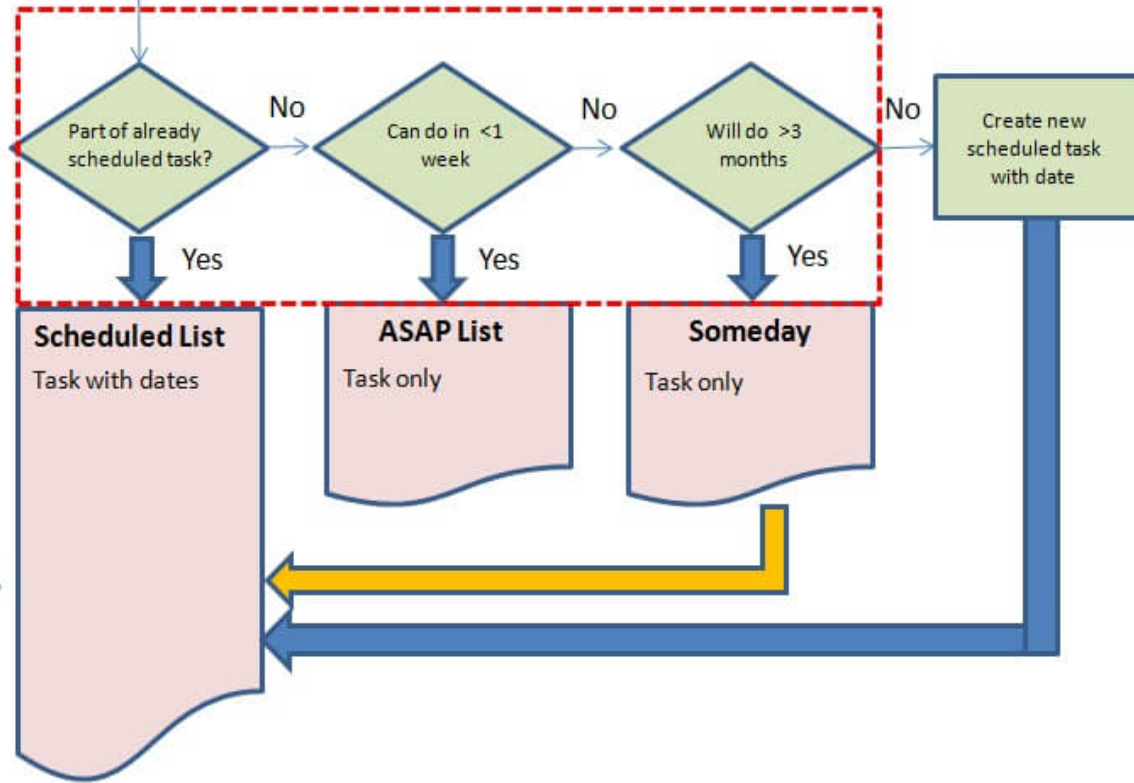
## COMPLETE TASK ENTRY FLOW



A 3 month "sprint" list is defined and planned from main objectives. This includes dates



No need to enter due dates, priorities etc. This automatically comes with the list it has been dropped into



## EXECUTING TASKS

When entering the ad-hoc daily tasks there was only one question governing the organisation of those tasks – When is the task needed? Guess what question is asked when determining the next task to do? Yip – When is the task needed! Easy to remember.

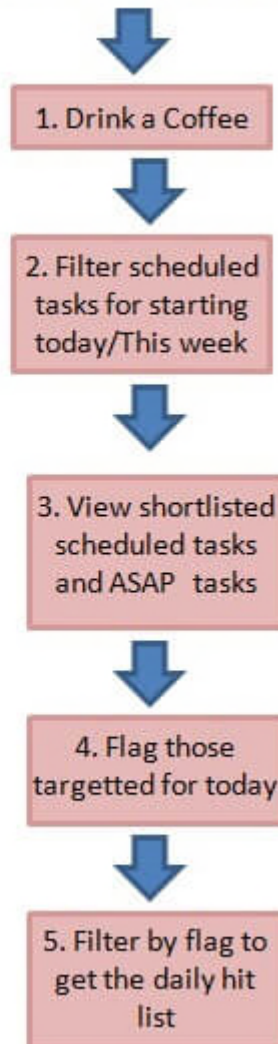
So, let's say we come in first thing in the morning at the crack of 9. First up, cup of coffee. The next three steps are the infamous 3 Fs, which get us from a fairly long task list to a daily hit list within seconds.

1. Filter your task list by date to show only those starting today/this week. This will dramatically shorten the Scheduled task list so you are left with a small subset of scheduled tasks and the ASAP list of tasks which are due this week. This not only identifies the tasks you need to be working on now but it also serves a second role – what are the other tasks that other people need to start/deliver. This is the SGTD equivalent of reviewing your delegation list in Getting Things Done.
2. Flag the tasks from this shortened list to identify those to be targeted today – your daily hitlist.

Filter again, this time by Flag. So, after this step we will only list the tasks that

The following flowchart illustrates how we get to a daily hit list from an extended task list.

## **“Start the day”**



## **SOFTWARE FOR LONG TERM AND DAILY PROJECT MANAGEMENT**

In applying this method, I use two software tools:

- MS Project
- [Abstractspoon todolist](#)

Let me explain my choice of these two tools.

Ideally I would use one software package to implement this system. But, as explained earlier I struggle to find such a system that offers all the project planning and day to day task management functionality that I seek. Not willing to settle for compromises I ended up using two tools to try and get the best of both worlds.

First up MS project. For project planning, this tool has always been the first port of call for any project manager. Complete and comprehensive project planning functionality is inherent in this tool. I will not detail all features in general but simply to explain what I need from it,

- To map out the tasks in terms of sequence and dependencies
- To identify any conflicts for resources or tasks
- To enter responsibilities for each task
- To enter dates for each task
- Ability to export this information to other software

So, MS Project allows us to formulate the basic sub-phase planning and to export that information.

This data is then entered into my day to day task management workhorse – Abstractspoon todolist.

What I need from this tool is,

- Ability to import MS Project data seamlessly
- Be light and efficient for day to day task manipulation
- Be able to share the tasks with team members
- To generate reports
- Be able to filter and search tasks

In addition to these necessities, Abstractspoon todolist has a few other nice features including,

1. Can import mindmaps from Freemind. So, can readily taskify brainstorming sessions.
2. Has Gantt views. So, basic project management capability.
3. Can share task lists using a check in/check out function with other team members
4. Automatic backup functions
5. Basic Kanban and Burndown functionality

So, those are the tools of the trade in this case. Now, let's consider a sample implementation.

## HOW TO IMPLEMENT MACRO TO MICRO PROJECT PLANNING

Up until now you have been bombarded with flow diagrams to illustrate the theory behind being able to master long term and daily task management in a single system. In reality it is simpler than it seems and to prove that I will work through an example to illustrate. I will show the five steps needed to go from a long term project plan to how to define what you do on any given day. These steps are,

1. Planning of the sub-phase
2. Export of the sub-phase planning
3. Import of the sub-phase tasks
4. Addition of day to day tasks
5. Selection of tasks for execution

Lets go through step by step using the software tools I suggested earlier.

### PLANNING OF THE PHASE/SUB-PHASE ACTIVITIES

I make an assumption at this point that we start from some long term project or objectives plan. If not available, not a big deal. Take one step back and create it to span a few months/years.

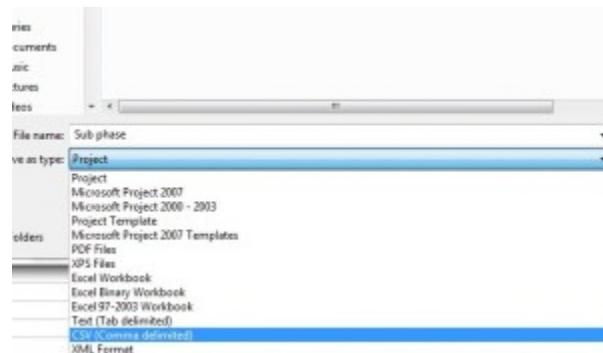
Right, plan now available. First step is to take a chunk of that plan for us to map out in detail spanning the next three months or so. The three months may cover an entire project phase such as the design or implementation stages or alternatively is may be a sub-phase of those stages. In my case and for the work I do it would be a sub-phase of a project stage. For the purposes of this example I will take one of the waterfall stages as our area of focus spanning a 3 months period. First step is to map out in detail what activities will be happening under that phase. Plan in as much detail as possible down to a 1 day resolution. Each of the tasks entered in this planning should have responsibilities and dates entered.



The “Analysis” Project stage is planned in detail for the 3 month “Sprint”.

## EXPORT OF THE SUB-PHASE PLANNING

Having completed detailed planning for the 3 month “sprint”, period we now export from the project management tool to our task management software where we will spend our time for the duration of the three months. In this example, it is achieved through saving the MS Project plan in csv format.



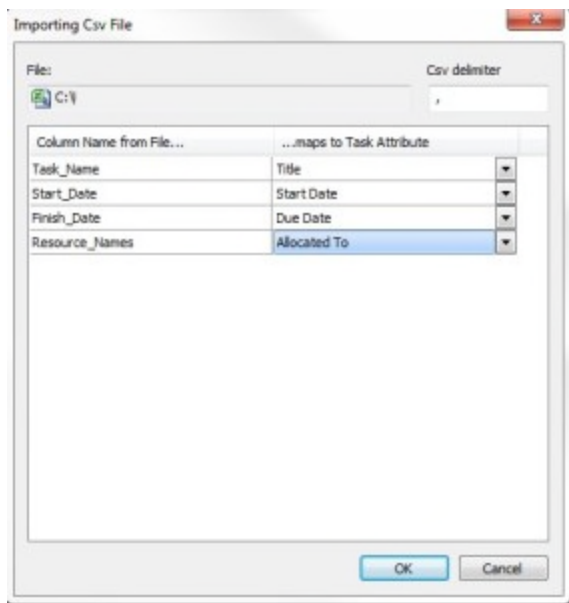
After selecting the save as csv option the MS project wizard will guide you through several steps to determining the content of the csv file. I use the following options,

1. Choose “Use Existing Map”
2. Under the new window – Choose a map for your data, select “Default task information
3. For Select the types of data you want to export, select “Tasks”. Other options can be left as default.
4. For Map Tasks Data – again leave as default.

All being well, having gone through those steps you will end up with a new csv file detailing the tasks, dates for the tasks and people responsible for them as defined in the main MS Project planning.

## IMPORT OF THE SUB-PHASE TASKS

Next we move onto our task management software. I use Abstractspoon todolist but others are possible where task import is possible such as HiTask. Here we simply use the import csv tool to read the plan just created in MS Project.



After importing place into the top level “Scheduled”, list. I manipulate the hierarchy to mirror that of the Gantt planning. Other lists for ASAP and Someday are also created at this point.

Options				
Any category C...				
Start	Due	To	By	Title
20/10/2015				<b>ASAP</b> <1 week
19/10/2015	19/2/2016			<b>Scheduled</b> >1 week and <3 months
19/10/2015	19/2/2016			<b>Analysis</b>
19/10/2015	30/10/2015	PM		Project scope
2/11/2015	6/11/2015	PM		Team formation
2/11/2015	11/12/2015	Marketing		Marketing research
9/11/2015	18/12/2015	PM		Resource planning
9/11/2015	1/1/2016	Development team		Solution evaluations
4/1/2016	15/1/2016	Controlling		Costing solutions
18/1/2016	19/2/2016	PM		Project proposal
21/10/2015				<b>Someday</b> >3 months

Congratulations, you have now just set up the most important tasks for this quarter in your task management system. Now on to the daily grind!

## ADDITION OF DAY TO DAY TASKS INTO PROJECT PLANS

No doubt it would be lovely to entertain oneself only with the important tasks just defined. In practice, many other tasks come along both self created and requests from other team members. These also have to be handled under the system and this is done so using the three lists explained earlier,

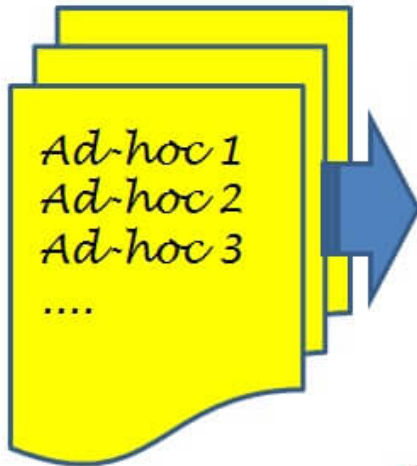
- ASAP list for tasks needed this week
- Someday list for tasks needed after 3 months
- Scheduled list for tasks after one week but before 3 months

Remember the simple question to identify the list – when is the task needed?

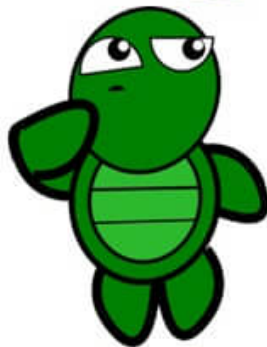
From the answer each task can then be assigned to the appropriate list. Let's add a few ad-hoc tasks into our existing list.



## Incoming Tasks



## Required Timeframe



Show	Title	Started by	Due by	Allocated to
A) All Tasks	<any>	<any date>	<any date>	<anyone>
Project: Type a descriptive name for your tasklist				
Due	To	Re	Title	
20/10/2015	Me		ASAP <1 week	Ad Hoc Task 1
20/10/2015	Me			Ad Hoc Task 2
24/10/2015	Me			Ad Hoc Task 3
21/10/2015			Scheduled >1 week and <3 months	
19/10/2015	19/02/2016		Analysis	
			Project scope	
			Ad Hoc Task 4	
			Ad Hoc Task 5	
			Team formation	
			Marketing research	
			Resource planning	
			Solution evaluations	
			Costing solutions	
			Project proposal	
			Someday >3 months	
			Ad Hoc Task 6	

So, by this stage we have now gone through all aspects of the implementation of the flow diagram. I have shown how to plan short sprints from a large project, how to implement these project activities into a to-dolist format and how they are then integrated with day to day tasks as indicated above. In effect, we have planned macro and micro tasks in our system.

Next up, doing stuff!

## SELECTION OF TASKS FOR EXECUTION IN A PROJECT PLAN

As explained in the “Executing tasks”, section we take a three step approach to identifying the tasks that we will do today.

1. Filter by starting date to find tasks commencing today or this week
2. From the filtered sublist, flag those tasks targeted for today
3. Flag again, this time by Flagged or highlighted tasks. This will then only list the tasks you have targeted for today and hide everything else. This is helpful in removing other distracting tasks and maintaining focus.

As per the previous implementation steps, I will show some examples of how this is done in practice. Again trusty Abstractspoon to-dolist will be the workhorse to demonstrate it.

## Step 1: Filter by “Starting today” or “Starting this week”.

The screenshot shows the Abstractspoon to-dolist interface. At the top, there are filter controls: 'Show' (set to 'A) All Tasks'), 'Title' (set to '<any>'), 'Started by' (set to 'D) End of This Week', 'Due by' (set to '<any date>'), and 'Allocated to' (set to '<anyone>'). The 'Started by' filter is highlighted with a red dashed box. Below the filters is a 'Project' field with the placeholder text 'Type a descriptive name for your tasklist'. The main task list is divided into two columns: a left column with dates and a right column with task titles. The left column has headers 'Start', 'Due', and 'To'. The right column has a header 'Title'. The tasks are categorized by urgency: 'ASAP' (lightning bolt icon), 'Scheduled' (calendar icon), and 'Someday' (lightbulb icon). The tasks listed are: 'Ad Hoc Task 1', 'Ad Hoc Task 2', 'Ad Hoc Task 3', 'Analysis', 'Project scope', and 'Ad Hoc Task 4'.

Start	Due	To	Task Title
21/10/2015			Ad Hoc Task 1
20/10/2015		Me	Ad Hoc Task 2
20/10/2015		Me	Ad Hoc Task 3
24/10/2015		Me	Ad Hoc Task 3
21/10/2015			Analysis
19/10/2015	19/02/2016		Project scope
19/10/2015	30/10/2015	PM	Ad Hoc Task 4
24/10/2015	28/10/2015	Tom	
21/10/2015			

Depending on the nature and duration of your tasks you can filter by either day or week. My preference is normally to filter by the week so I can get a bit of a heads up on tasks that may be one or two days further down the line. In some cases you may want to get a start on those and the prerequisite for that is being able to see them in the system!

## Step 2:

### Flag those tasks targeted for today

Show	Title	Started by	Due by	Allocated to
A) All Tasks	<any>		D) End of This We	<any date>
Project <input type="text" value="Type a descriptive name for your tasklist"/>				

Start	Due	To	Flag	Title [ASAP\]
21/10/2015				<b>ASAP</b> <1 week
20/10/2015		Me		Ad Hoc Task 1
20/10/2015		Me		Ad Hoc Task 2
24/10/2015		Me		Ad Hoc Task 3
21/10/2015				<b>Scheduled</b> >1 week and <3 months
19/10/2015	19/02/2016			Analysis
19/10/2015	30/10/2015	PM		Project scope
24/10/2015	28/10/2015	Tom		Ad Hoc Task 4
21/10/2015				<b>Someday</b> >3 months

In this program, Flag is used to select tasks. In other programs it may be referred to in other terms. But, whatever the labelling it is essentially the tasks that you want to focus on for that day.

## Step 3:

### Filter by flag to show only those tasks targeted for today. This helps remove other distracting activities from view

Show	Title	Started by	Due by	Allocated to
D) Flagged Tasks	<any>		D) End of This We	<any date>
Project <input type="text" value="Type a descriptive name for your tasklist"/>				

Start	Due	To	Flag	Title [ASAP\]
21/10/2015				<b>ASAP</b> <1 week
20/10/2015		Me		Ad Hoc Task 1
21/10/2015				<b>Scheduled</b> >1 week and <3 months
19/10/2015	19/02/2016			Analysis
19/10/2015	30/10/2015	PM		Project scope

## IN PRACTICE

Until now I have shown the workflow and given an example of it's implementation. In this section, I will cover a couple of topics that are relevant in it's real world application. Namely maintenance of the lists and tracking progress

## TASK LIST MAINTENANCE

There is only two reviews advocated in this approach,

- The daily morning review to identify the tasks for the day
- A quarterly review every 3 months of the someday list

That's it. One tip I would give though is to do the daily review at the end of a workday so coming in the next morning your tasks for the day have been identified and you can begin immediately.

## TRACKING PROGRESS OF A PROJECT WITH ABSTRACTSPOON TODOLIST

Given that we have defined all of the tasks necessary to complete a project sub-phase and the necessary tracking information such as responsibilities and dates, it is possible to tack progress towards reaching completion using both traditional and non-traditional ways.

### *Task list views and filtering options*

First and most obvious way to determine progress is to have a quick glimpse across the task lists and to avail oneself of the filtering options available for whatever tool you happen to be using. In my case with Abstractspoon todolist, late tasks are automatically shown in red text. So, if you have a sea of red before you that is a reasonable indication things are not running to plan.

In terms of filtering, many task management software solutions will come with some built in filters. An example is shown below, whereby you can filter for tasks past their due date.

Show	Title	Started by	Due by	Allocated to
A) All Tasks	<any>	<any date>	<any date>	<anyone>

Project	Start	Due	To	✓	Title [ASAP\]
	21/10/2015				<b>ASAP</b> <1
	20/10/2015		Me		Ad Hoc T
	20/10/2015		Me		Ad Hoc T
	24/10/2015		Me		Ad Hoc T
	21/10/2015				<b>Scheduled</b>
	19/10/2015	19/02/2016			Analysis
	19/10/2015	30/10/2015	PM		Proje
	24/10/2015	28/10/2015	Tom		Ad Hi
	27/10/2015	31/10/2015	Bob		Ad Hoc Task 5
	02/11/2015	06/11/2015	PM		Team formation
	02/11/2015	11/12/2015	Marketing		Marketing research
	09/11/2015	18/12/2015	PM		Resource planning
	09/11/2015	01/01/2016	Development team		Solution evaluations
	04/01/2016	15/01/2016	Controlling		Costing solutions
	18/01/2016	19/02/2016	PM		Project proposal
	21/10/2015				<b>Someday</b> >3 months

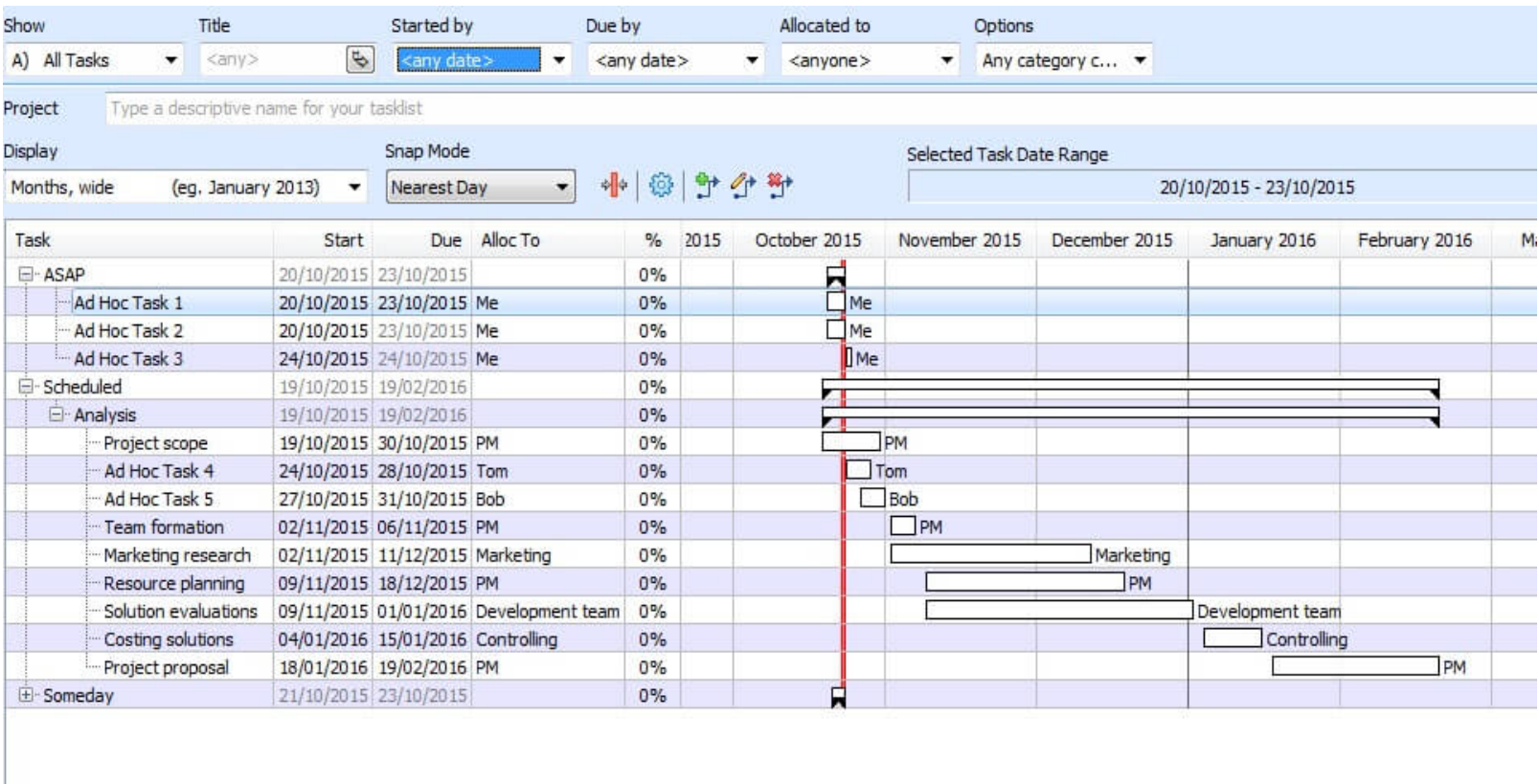
### Report Generation

Another great tool is the ability to generate reports on the status of completed, in progress or upcoming tasks. Many task management tools already have this in place. These days, with the fascination for web based apps, the ability to view in real time the on-going activities of all team members adds another dimension to this.

### Gantt Chart

Let's start with the tried and trusted Gantt chart. Depending on the tool used it should be possible to view the tasks in Gantt view which shows the timeline and the tasks accompanying that. Earlier Gantt charts were used to map out the activities in the sprint. In this case though they are merely a visual aid to indicate which tasks are behind schedule. We try to minimise the shuffling around of the scheduled activities but in practice though, I see it is still necessary to do some editing.



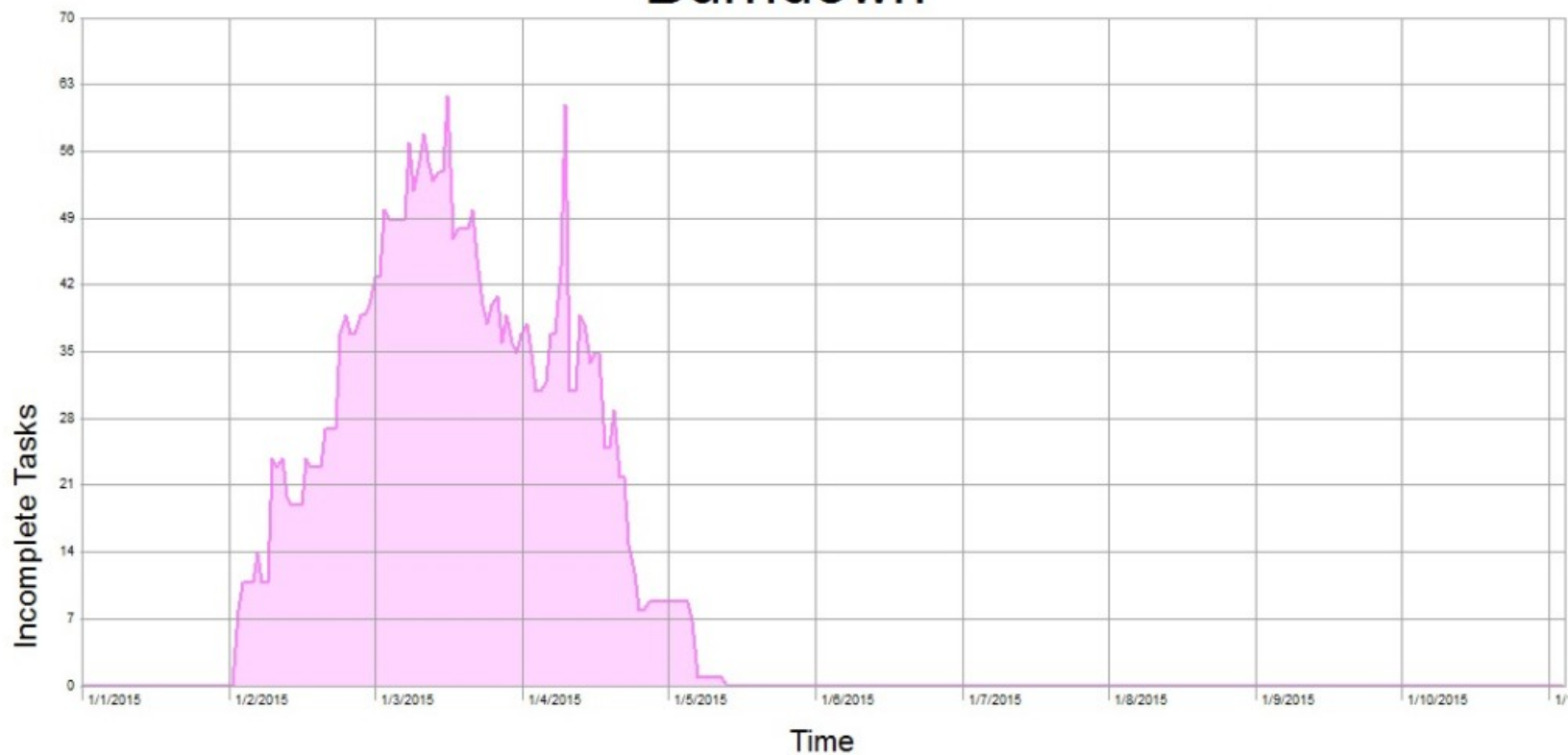


## Burndown charts

Burndown charts are commonly used in SCRUM sprints to indicate progress towards completing the tasks defined in that sprint. The software I use for task management, Abstractspoon todolist, has basic charting functionality for this to show the number of open tasks. Personally I don't use this but not to say it cannot be used. Anyway, I show some interesting results for this for real world projects that I have been running.

Let's take a single subphase, see below. Here you can see the tasks begin at zero at the beginning of activities, rise to a peak and then decrease as the tasks are executed and we move towards the end of the sub-phase. This is quite different as to what one would normally see in such plots.

# Burndown

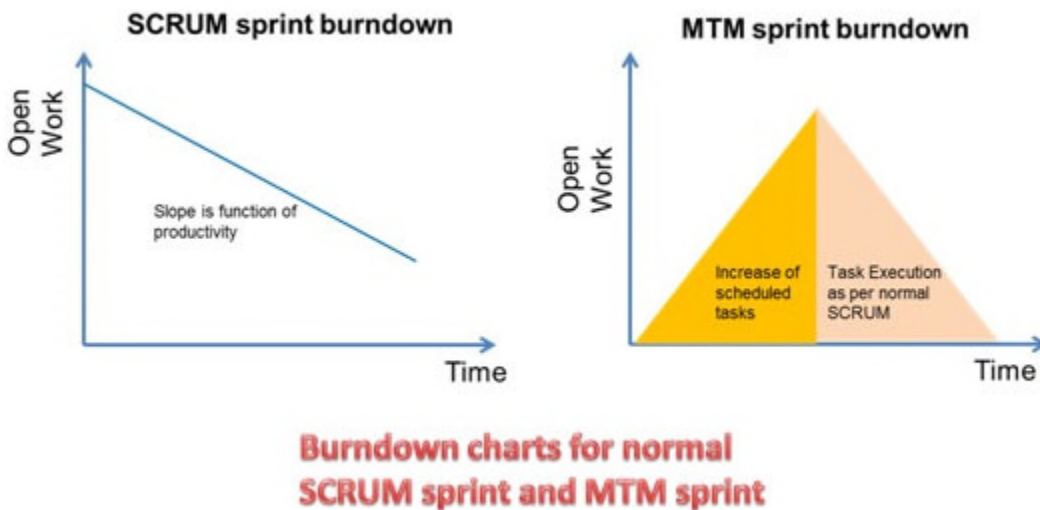


The difference between normal SCRUM burndown charts and the MTM burndown can be shown in the diagram below. On the left a traditional burndown plot. A batch of tasks are defined at the start of the sprint and as the sprint progresses these are completed until none are left at the end. The speed at which the tasks/work is completed indicates the productivity or velocity of the team towards meeting the goal.

In contrast, the MTM sprint is more like a triangle. Having seen this interesting result I concluded this was due to three factors,

- As activities begin they are chunked (divided into smaller actionable subtasks for execution)
- As time progresses, new scheduled activities emerge
- There is task overlap between workers

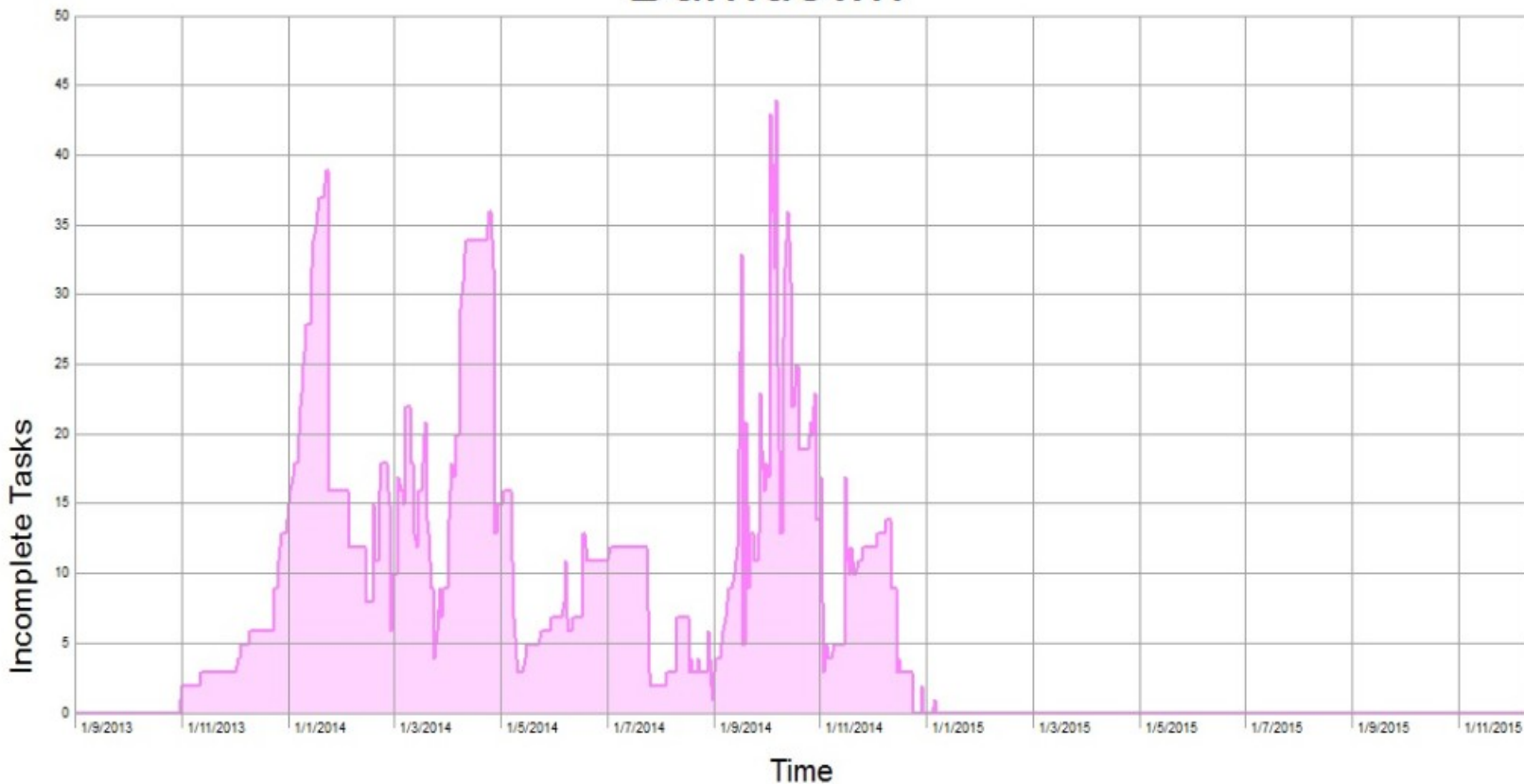
So, if one simply looks at the number of tasks it would probably take this shape. However, I think if it were plotted in terms of open work or hours required then we would probably be back to the traditional SCRUM burndown chart as at the start we would have defined x amount of work and as time progresses x would be diminished, even if tasks were sub-divided into smaller actionable steps. That being the case it would enable the same SCRUM progress tracking methodologies to be applied also to this MTM approach.



When looking at the burndown chart across several sub-phases we see something like the chart below. This is for activities I did in 2014, again real world examples of this approach in action. Here we see the sub-phase triangles merging with one another. So, one "sprint", completes and then this is repeated through the project.



## Burndown



Burndown Chart for Macro to Micro Task Management

## TASK MANAGEMENT FOR PROJECT MANAGERS SUMMARY

Wow. Long post. Congratulation for reaching the end and if you have a beer on hand, cheers!

At the start of the article I made some bold claims as to what this system could deliver. Going through the article I explained the approach with flow diagrams, gave real world examples of how it pans out in practice and gave a simple illustration on how to go from a long term project plan to what to do at 8:20am on Tuesday morning. I hope this Project Management Methodology covering Task Management will be of use to you and feel free to comment below on any ideas for improvement!

## ABOUT THE AUTHOR



### **DONE BEFORE BREKKY**

Brendan Toner is the bloke behind the alternative productivity blog, DoneBeforeBrekky.com. For movers and shakers who value their time, DBB serves up an effervescent cocktail of information on the best tools and techniques to do it.

Learn more: <http://donebeforebrekky.com/>