Horizon Planning Spreadsheet

By
David A. Penny
(https://www.modsoftman.com)

The following Excel file download is a fully practical sample Horizon Planning Spreadsheet that is typical of the ones I have used many times to manage software products.

HP-SampleDownload

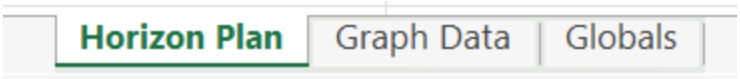
Please feel free to download it and adapt it for your software projects.

For a short intro to method and the benefits of Horizon Planning see

Horizon Planning Intro.

I will now go through the spreadsheet in detail and explain how it's put together and used.

There are three tabs on the spreadsheet.

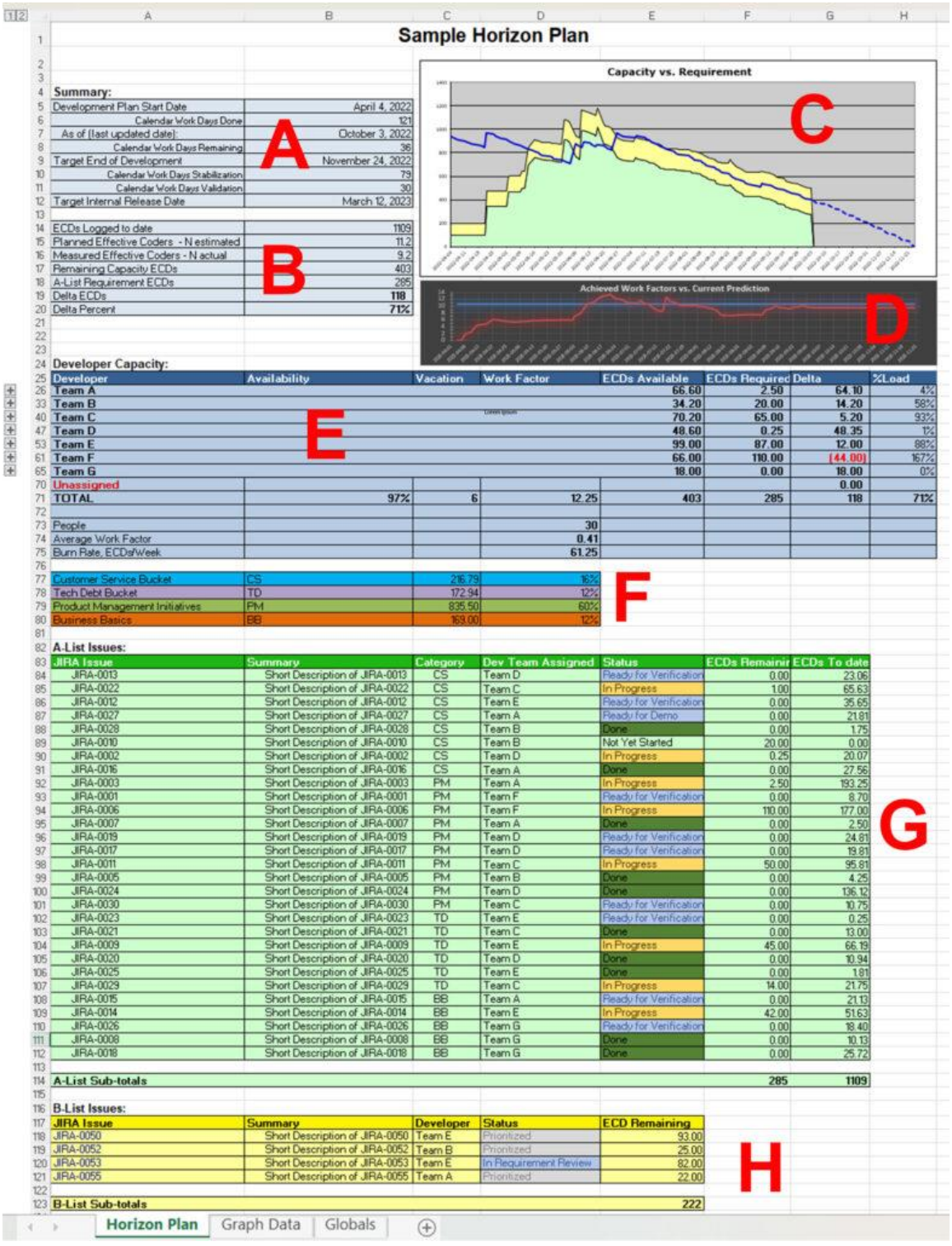


- Globals Tab
  Contains reference constants – the holiday schedule.
- Graph Data Tab
  Contains historical data used to plot the graphs.
- Horizon Plan Tab
  Where all the action is.

We'll come back to **Graph Data** and **Globals**. For now, let's focus on the **Horizon Plan** tab.

Horizon Plan Tab

It’s major sections are as follows.



Sample Horizon Plan

Summary:

Development Plan Start Date	April 4, 2022
Calendar Work Days Done	121
As of (last updated date):	October 3, 2022
Calendar Work Days Remaining	36
Target End of Development	November 24, 2022
Calendar Work Days Stabilization	79
Calendar Work Days Validation	30
Target Internal Release Date	March 12, 2023

A

ECDs Logged to date	1109
Planned Effective Coders - N estimated	11.2
Measured Effective Coders - N actual	9.2
Remaining Capacity ECDs	403
A-List Requirement ECDs	285
Delta ECDs	118
Delta Percent	71%

B

C

D

Developer Capacity:

E

Developer	Availability	Vacation	Work Factor	ECDs Available	ECDs Required	Delta	%Load
Team A				66.60	2.50	64.10	4%
Team B				34.20	20.00	14.20	58%
Team C				70.20	65.00	5.20	93%
Team D				48.60	0.25	48.35	1%
Team E				99.00	87.00	12.00	88%
Team F				66.00	110.00	(44.00)	167%
Team G				18.00	0.00	18.00	0%
Unassigned						0.00	
TOTAL	97%	6	12.25	403	285	118	71%
People			30				
Average Work Factor			0.41				
Burn Rate, ECDs/Week			61.25				

F

Customer Service Bucket	CS	216.79	16%
Tech Debt Bucket	TD	172.94	12%
Product Management Initiatives	PM	835.50	60%
Business Basics	BB	169.00	12%

A-List Issues:

G

JIRA Issue	Summary	Category	Dev Team Assigned	Status	ECDs Remainir	ECDs To date
JIRA-0013	Short Description of JIRA-0013	CS	Team D	Ready for Verification	0.00	23.06
JIRA-0022	Short Description of JIRA-0022	CS	Team C	In Progress	1.00	65.63
JIRA-0012	Short Description of JIRA-0012	CS	Team E	Ready for Verification	0.00	35.65
JIRA-0027	Short Description of JIRA-0027	CS	Team A	Ready for Demo	0.00	21.81
JIRA-0028	Short Description of JIRA-0028	CS	Team B	Done	0.00	1.75
JIRA-0010	Short Description of JIRA-0010	CS	Team B	Not Yet Started	20.00	0.00
JIRA-0002	Short Description of JIRA-0002	CS	Team D	In Progress	0.25	20.07
JIRA-0016	Short Description of JIRA-0016	CS	Team A	Done	0.00	27.56
JIRA-0003	Short Description of JIRA-0003	PM	Team A	In Progress	2.50	193.25
JIRA-0001	Short Description of JIRA-0001	PM	Team F	Ready for Verification	0.00	8.70
JIRA-0006	Short Description of JIRA-0006	PM	Team F	In Progress	110.00	177.00
JIRA-0007	Short Description of JIRA-0007	PM	Team A	Done	0.00	2.50
JIRA-0019	Short Description of JIRA-0019	PM	Team D	Ready for Verification	0.00	24.81
JIRA-0017	Short Description of JIRA-0017	PM	Team D	Ready for Verification	0.00	19.81
JIRA-0011	Short Description of JIRA-0011	PM	Team C	In Progress	50.00	95.81
JIRA-0005	Short Description of JIRA-0005	PM	Team B	Done	0.00	4.25
JIRA-0024	Short Description of JIRA-0024	PM	Team D	Done	0.00	136.12
JIRA-0030	Short Description of JIRA-0030	PM	Team C	Ready for Verification	0.00	10.75
JIRA-0023	Short Description of JIRA-0023	TD	Team E	Ready for Verification	0.00	0.25
JIRA-0021	Short Description of JIRA-0021	TD	Team C	Done	0.00	13.00
JIRA-0009	Short Description of JIRA-0009	TD	Team E	In Progress	45.00	66.19
JIRA-0020	Short Description of JIRA-0020	TD	Team D	Done	0.00	10.94
JIRA-0025	Short Description of JIRA-0025	TD	Team E	Done	0.00	1.81
JIRA-0029	Short Description of JIRA-0029	TD	Team C	In Progress	14.00	21.75
JIRA-0015	Short Description of JIRA-0015	BB	Team A	Ready for Verification	0.00	21.13
JIRA-0014	Short Description of JIRA-0014	BB	Team E	In Progress	42.00	51.63
JIRA-0026	Short Description of JIRA-0026	BB	Team G	Ready for Verification	0.00	18.40
JIRA-0008	Short Description of JIRA-0008	BB	Team G	Done	0.00	10.13
JIRA-0018	Short Description of JIRA-0018	BB	Team G	Done	0.00	25.72
A-List Sub-totals					285	1109

B-List Issues:

H

JIRA Issue	Summary	Developer	Status	ECD Remaining
JIRA-0050	Short Description of JIRA-0050	Team E	Prioritized	93.00
JIRA-0052	Short Description of JIRA-0052	Team B	Prioritized	25.00
JIRA-0053	Short Description of JIRA-0053	Team E	In Requirement Review	82.00
JIRA-0055	Short Description of JIRA-0055	Team A	Prioritized	22.00
B-List Sub-totals				222

Horizon Plan | Graph Data | Globals

Contents:

A – Dates

	A	B
5	Development Plan Start Date	April 4, 2022
6	Calendar Work Days Done	121
7	As at (last updated date):	October 3, 2022
8	Calendar Work Days Remaining	36
9	Target End of Development	November 24, 2022
10	Calendar Work Days Stabilization	79
11	Calendar Work Days Validation	20
12	Target Internal Release Date	April 17, 2023

Development Plan Start Date, also known as **fork**, is the date on which the planning horizon starts.

As at is the date the plan was last updated.

Calendar Work Days Done are the number of working days (excludes weekends, statutory holidays, and any days where it is known in advance no coding work on the features is expected to be done, *e.g.,* a big company offsite).

Target End of Development, also known as **dcut**, is the date we are aiming to be done the major coding. All the devs working on all the A-List Features need to be able to say they are not aware of any more code that needs to be merged in (including test code) for the feature development to be complete.

Calendar Work Days Remaining are the number of working days remaining between **as-at** and **dcut**.

Target Internal Release Date is the date that the dev team plans to declare they are fully done developing and testing the software. This date is computed based on the three fields above it.

Calendar Work Days Stabilization is the number of workdays estimated to stabilize the software after **dcut**. In this sample plan it is computed as 50% of the workdays between **fork** and **dcut**. During this time, the devs are all expected to be available to fix any issues discovered as a first priority. With any spare time, they can do further test development or get a jump on the next horizon.

Calendar Work Days Validation is a constant number of workdays tacked onto the end of stabilization for a final QA run-through.

Note that this particular incarnation of the Horizon Plan (HP) is designed for mission-critical enterprise on-premises software installed into diverse customer environments and upgraded in diverse ways by the customer, hence the lengthy stabilization and validation phases towards the end.

For cloud-based software with more continuous delivery-to-production over the horizon, these phases are eliminated. In part they are not needed as there is only one self-hosted operating environment for the software, and in part these dev teams will typically spend much more in the Dev phase on infrastructure-as-code, various types of automated testing, automated deployment code, and production monitoring code. In this case, the Horizon does not represent a “Release” of the software, but rather a business-convenient planning horizon (*e.g.,* 6 months).

B – Summary

	A	B
14	ECDs Logged to date	1109
15	Planned Effective Coders - N estimated	10.1
16	Measured Effective Coders - N actual	9.6
17	Remaining Capacity ECDs	365
18	A-List Requirement ECDs	285
19	Delta ECDs	**80**
20	Delta Percent	**78%**

ECDs Logged To Date is the number of “Effective Coder Days” (logged ideal hours spent coding divided by 8) across the entire team from **fork** to **dcut**.

Planned Effective Coders – N estimated is a *forward-looking estimate* of the average number of ECDs the team will put in per workday from **as-at** to **dcut**. It is computed by dividing the estimated remaining capacity ECDs (365 – computed in section E and brought up here) by the remaining workdays (36 – computed in section A).

Measured Effective Coders – N actual is an average of the number of ECDs the Devs have been putting in per recent workdays (it is an exponentially decaying average computed on the **Graph Data** tab).

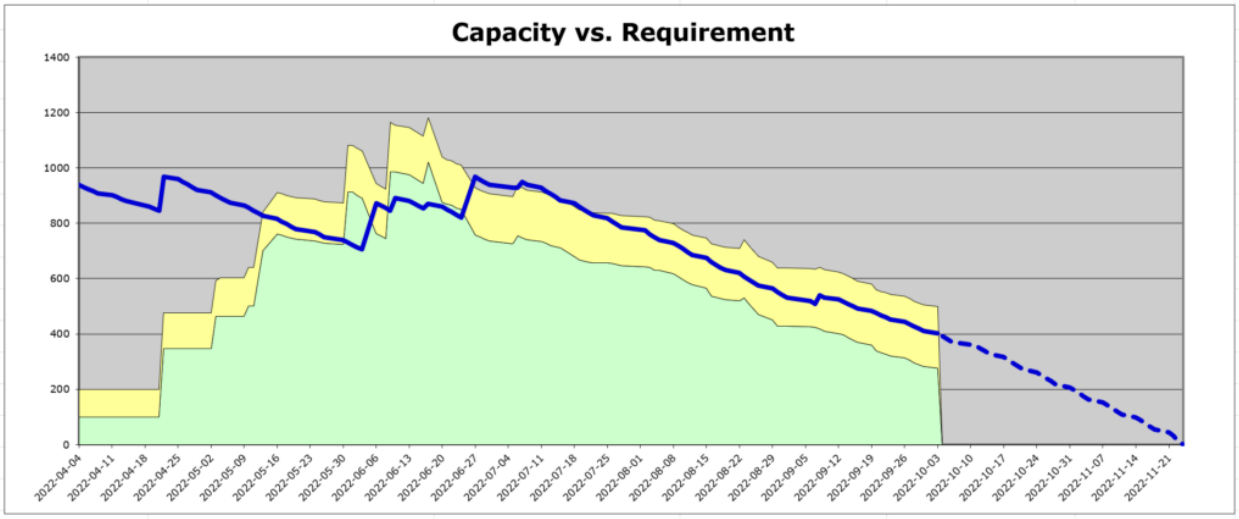
Remaining Capacity ECDs is the total estimated remaining capacity of the team from **as-at** to **dcut**. This takes into account the remaining workdays, any holidays, any estimated vacation during that period, any partial allocations of time to other pursuits, and the Dev work factors (how many ECDs per workday they estimate they can get done averaged over the remaining workdays). It is computed in section E and brought up here.

A-List Requirement ECDs is the total estimated remaining requirement for ECDs to complete all the “A-List” features. It is computed in section G and brought up here.

Delta ECDs is the difference between the above two. A negative value here is bad.

Delta Percent is the ratio between estimated remaining requirement and capacity. We generally want to keep this around 70% to provide a safety buffer for the A-List and allow time for some B-list items to make it.

C – Capacity/Requirement Burndown Chart



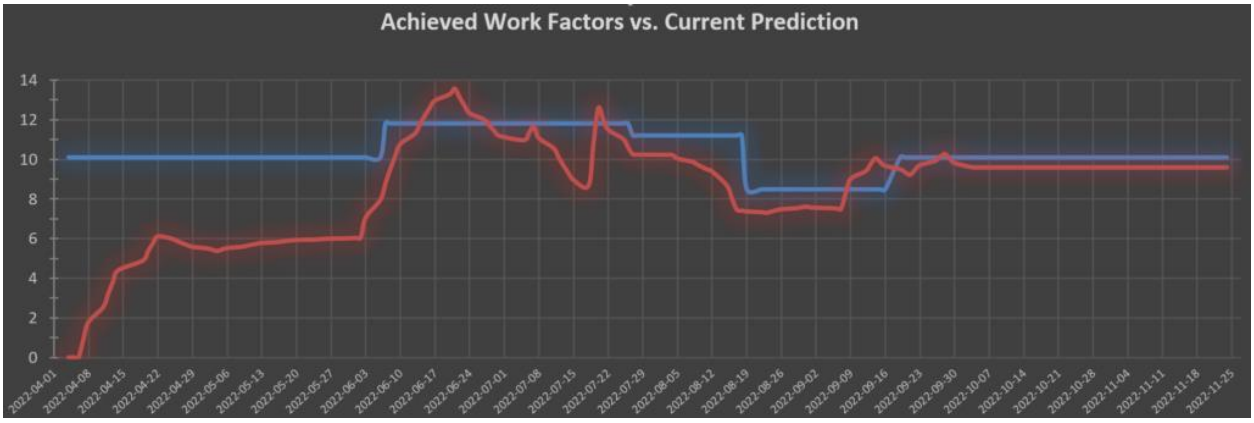
This chart records historical values of estimated capacity (**solid blue line**) and estimated remaining requirement for the A-List features (**green**) and B-list features (**yellow**). Dotted blue is the projected capacity decrease as we reach **dcut**. This chart can be used to get a quick visual indication if the plan seems to be on-track (the blue and green decreasing at approximately the same rate).

In this sample plan, looking at the green, work started at 04-14 but the plan was not yet full and requirements kept coming in until around 06-20. Meanwhile, looking at the blue, there appears to have been some adjustments of the end date and some re-estimation up of work factors. By 07-11 the estimates seem to be better dialed in and estimated remaining capacity and requirements burn-down in tandem.

The data for this chart is sourced from the Graph Data Tab which which has a time series that is added onto on each plan update.

[Special thanks to my friend and colleague Matt Medland for adding this concept]

D – Measured versus Predicted Capacity



This chart plots the historical go-forward estimate of average daily capacity (blue) and the recent measured average daily capacity as of that date. It can be used to determine at a glance if our current estimates of daily capacity seem to be in accordance with our recent actual daily capacity. Our minds average out the blips.

The data for this chart is sourced from the Graph Data Tab which has a time series that is added to on each plan update. The red line is a smoothed exponential decaying average of instantaneous daily measured capacity ($\lambda = 0.1$).

E – Dev Capacity

	A	B	C	D	E	F	G	H
25	**Developer**	**Availability**	**Vacation**	**Work Factor**	**ECDs Available**	**ECDs Required**	**Delta**	**%Load**
26	**Team A**				**55.80**	**2.50**	**53.30**	4%
33	**Team B**				**27.00**	**20.00**	**7.00**	74%
40	**Team C**				**59.40**	**65.00**	**(5.60)**	109%
41	Team Lead 11	100%	0	0.20	7.20			
42	Dev 12	100%	0	0.45	16.20			
43	Dev 13	100%	0	0.45	16.20			
44	Dev 14	100%	0	0.35	12.60			
45	Intern 15	100%	0	0.20	7.20			
46								
47	**Team D**				**45.00**	**0.25**	**44.75**	1%
53	**Team E**				**95.40**	**87.00**	**8.40**	91%
61	**Team F**				**66.00**	**110.00**	**(44.00)**	167%
65	**Team G**				**16.20**	**0.00**	**16.20**	0%
70	**Unassigned**						**0.00**	
71	**TOTAL**	**97%**	**6**	**11.00**	**365**	**285**	**80**	**78%**
72								
73	Devs			**30**				
74	Average Work Factor			**0.37**				
75	Burn Rate, ECDs/Week			**55.00**				

Developer lists all the individuals who are expected to contribute coding work to the effort, grouped into teams (Team C is expanded). In an actual plan, these would be real names.

Availability is the fraction of the dev's work time expected to be devoted to this effort. For example, if a dev is split evenly across two initiatives it would go in as 50%.

Vacation is an estimate of the remaining vacation days to be taken by the dev from **as-at** to **dcut**.

Work Factor (**w**) is an estimate of the average number of ideal hours (8*w) per workday each dev expects to spend on individual contributor "coding work" on items in the plan from **as-at** to **dcut**. "Coding work" can be defined in various ways but will typically include writing code, tests, and docs, and excludes any meeting time or time spent conceiving of and/or writing specs and designs. Whatever the precise definition is, it must match the definition used when estimating feature effort.

Note that work factors are typically lower for team leads, and will also be lower for architects or devs who do a disproportionate of other types of work (*e.g.,* triaging and fixing defects found in the field, pipeline maintenance, production operations, and so on). A low work factor may also be coupled to a particularly productive dev who accomplishes a great deal in a short time. It is not a "value judgment", it is a quantity defined and measured via granular time tracking.

ECDs Available is computed by (remaining workdays – vacation) * w * availability.

ECDs Required is computed at the team level only and is derived by summing the remaining estimates for all A-list features assigned to that team in section G.

Delta is the team-level difference between the **ECDs Available** and **ECDs Required**. Negative values indicates the team is currently overcommitted.

%Load is the ratio of **ECDs Available** to **ECDs Required**. Values greater that 100% indicates that the team is currently overcommitted.

The **Unassigned** row lists the sum of the estimates for any features not assigned to any team.

The **TOTAL** row is either an average or a sum depending on context.

Devs is a count of the number of individuals listed in the plan.

Average Work Factor is the average work factor across all the devs.

Burn Rate is the expected number of ECDs that can be accomplished by the entire team during a typical 5-day work week.

F – Feature Effort Summary

	A	B	C	D
77	Customer Service Bucket	CS	217	16%
78	Tech Debt Bucket	TD	173	12%
79	Product Management Initiatives	PM	836	60%
80	Business Basics	BB	169	12%

Gives the ECDs to-date plus estimated remaining (from section G) across broad categories in the A-list, expressing it both as absolute ECDs and the ECDs as a percentage of the total.

The categories should be setup depending on what has the most value. For this example, these categories are defined as follows.

- **Product Management Initiatives**
  New features coming from product management designed to increase the marketability of the software.
- **Customer Service Bucket**
  New features championed by the customer service team to make life easier for themselves or for specific customers, but may not have broad marketability benefits.
- **Business Basics**
  Things that need to be done to the software to keep it current (*e.g.,* upgrade the operating system or libraries to their latest stable releases, or keeping up with a partner's latest releases of their software).
- **Tech Debt Bucket**
  Initiatives championed by the dev team to improve the architecture and performance of the software and clean up bad code or dead code.

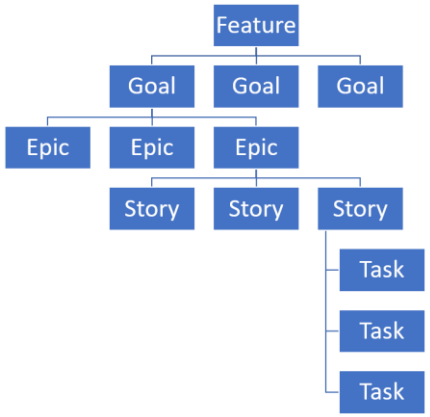
Note that there is no category for "Defect Fixing". In the exceptional case that a defect requires a high-effort fix, it will be treated like a feature. If a defect is found that is associated with a new in-plan feature, it is included as part of the development of that feature. Any effort spent on other defects is considered part of the "background noise" and is estimated by reducing the work factors accordingly.

G – A-List Feature Requirement

	A	B	C	D	E	F	G
83	JIRA Issue	Summary	Category	Dev Team Assigned	Status	ECDs Remaining	ECDs To-Date
84	JIRA-0013	Short Description of JIRA-0013	CS	Team D	Ready for Verification	0.00	23.06
85	JIRA-0022	Short Description of JIRA-0022	CS	Team C	In Progress	1.00	65.63
86	JIRA-0012	Short Description of JIRA-0012	CS	Team E	Ready for Verification	0.00	35.65
87	JIRA-0027	Short Description of JIRA-0027	CS	Team A	Ready for Demo	0.00	21.81
88	JIRA-0028	Short Description of JIRA-0028	CS	Team B	Done	0.00	1.75
89	JIRA-0010	Short Description of JIRA-0010	CS	Team B	Not Yet Started	20.00	0.00
90	JIRA-0002	Short Description of JIRA-0002	CS	Team D	In Progress	0.25	20.07
91	JIRA-0016	Short Description of JIRA-0016	CS	Team A	Done	0.00	27.56
92	JIRA-0003	Short Description of JIRA-0003	PM	Team A	In Progress	2.50	193.25
93	JIRA-0001	Short Description of JIRA-0001	PM	Team F	Ready for Verification	0.00	8.70
94	JIRA-0006	Short Description of JIRA-0006	PM	Team F	In Progress	110.00	177.00
95	JIRA-0007	Short Description of JIRA-0007	PM	Team A	Done	0.00	2.50
96	JIRA-0019	Short Description of JIRA-0019	PM	Team D	Ready for Verification	0.00	24.81
97	JIRA-0017	Short Description of JIRA-0017	PM	Team D	Ready for Verification	0.00	19.81
98	JIRA-0011	Short Description of JIRA-0011	PM	Team C	In Progress	50.00	95.81
99	JIRA-0005	Short Description of JIRA-0005	PM	Team B	Done	0.00	4.25
100	JIRA-0024	Short Description of JIRA-0024	PM	Team D	Done	0.00	136.12
101	JIRA-0030	Short Description of JIRA-0030	PM	Team C	Ready for Verification	0.00	10.75
102	JIRA-0023	Short Description of JIRA-0023	TD	Team E	Ready for Verification	0.00	0.25
103	JIRA-0021	Short Description of JIRA-0021	TD	Team C	Done	0.00	13.00
104	JIRA-0009	Short Description of JIRA-0009	TD	Team E	In Progress	45.00	66.19
105	JIRA-0020	Short Description of JIRA-0020	TD	Team D	Done	0.00	10.94
106	JIRA-0025	Short Description of JIRA-0025	TD	Team E	Done	0.00	1.81
107	JIRA-0029	Short Description of JIRA-0029	TD	Team C	In Progress	14.00	21.75
108	JIRA-0015	Short Description of JIRA-0015	BB	Team A	Ready for Verification	0.00	21.13
109	JIRA-0014	Short Description of JIRA-0014	BB	Team E	In Progress	42.00	51.63
110	JIRA-0026	Short Description of JIRA-0026	BB	Team G	Ready for Verification	0.00	18.40
111	JIRA-0008	Short Description of JIRA-0008	BB	Team G	Done	0.00	10.13
112	JIRA-0018	Short Description of JIRA-0018	BB	Team G	Done	0.00	25.72
113							
114	**A-List Sub-totals**					**285**	**1109**

A table of all the A-List “Features” completed, in progress, or currently in-plan but not yet started.

A “Feature” is a cohesive unit of business value. It is typically at the top of a hierarchy of Jira issues, such as the following.



(with Goals and Epics as optional, and some teams preferring not to use Tasks).

Only the Feature level is exposed in the horizon plan. One can use a Jira plugin such as the Hierarchy Plugin to aggregate estimates and actual time up the hierarchy and pull a report that can be imported into the HP Spreadsheet on plan updates.

*[one day **somebody** should write a Jira plugin for HP or an external SaaS system that integrates with Jira and do away with Excel. Now that I'm retired I'll write something myself at some point, but partners are welcome!]*

Jira Issue is a link back to the Feature-level Issue in Jira.

Summary is the short description of the Feature pulled from Jira.

Category is added in the release plan (or imported from a custom Jira field) and refers to the categories summarized in section F.

Dev Team Assigned is the dev team that is currently assigned to do the work. Often certain teams have expertise in certain areas, and an overcommitment on a certain team may indicate that people need to be moved and re-trained (or those types of features decreased).

Status is the current status of the Feature pulled from Jira and will differ based on local dev practices.

ECDs Remaining is the current estimate of the amount of effort required to complete the feature to **dcut**. The effort that is estimated should be compatible with how the **Work Factor** (**w**) is measured. Note that any ideal time estimate blends together the following three (inseparable) concepts:

- An estimate of the *inherent size* of the feature.
- An estimate of *who* will work on the feature.
- An estimate of how productive that person will be working on that feature.
    - (how productive they will be with each ideal hour they can devote to it, *NOT* how many ideal hours they can devote to it over time – that is estimated in the work factor).

ECDs To-Date is the aggregate of all the ideal time spent to-date on each feature. I advocate that devs do fine-grained time-tracking of this one thing. Jira provides a "worklog" feature to enable this along with re-estimates of effort remaining. Tracking this closes the loop on feature and work factor estimates and allows for incremental improvement.

A-List Sub-Totals are the sub-total of effort spent and effort remaining on the A-List features.

H – B-List Feature Requirement

	A	B	C	D	E	F	G
117	**JIRA Issue**	**Summary**	**Category**	**Dev Team Assigned**	**Status**	**ECDs Remaining**	**ECDs To-Date**
118	JIRA-0050	Short Description of JIRA-0050	PM	Team E	Prioritized	93.00	0.00
119	JIRA-0052	Short Description of JIRA-0052	PM	Team B	Prioritized	25.00	0.00
120	JIRA-0053	Short Description of JIRA-0053	TD	Team E	In Requirement Review	82.00	0.00
121	JIRA-0055	Short Description of JIRA-0055	BB	Team A	Prioritized	22.00	0.00
122							
123	**B-List Sub-totals**					**222**	**0**

The B-List are additional features that a dev may work on once they have no work remaining on any A-List feature. Rather than simply “padding” the plan we strive to 100% fill the plan with A+B Features, however it is understood that the B-List Features are a stretch goal.

Graph Data Tab

	A	B	C	D	E	F	G	H	I	J
1									exp decay factor:	0.10
2		Date	Workdays	A-List Requirement	B-List Requirement	Capacity	Predicted Capacity	ECDs logged to date	Estimated ECDs	Averaged Daily ECDs
3		April 4, 2022	1	100	100	939		0.0	10.7	0.0
4		April 5, 2022	2	100	100	931		0.0	10.7	0.0
5		April 6, 2022	3	100	100	922		0.0	10.7	0.0
6		April 7, 2022	4	100	100	915		9.4	10.7	0.9
7		April 8, 2022	5	100	100	908		18.8	10.7	1.8
8		April 11, 2022	6	100	100	901		28.2	10.7	2.5
9		April 12, 2022	7	100	100	894		37.6	10.7	3.2
119		September 27, 2022	117	305	222	435		1047.5	10.7	10.1
120		September 28, 2022	118	295	222	427		1059.3	10.7	10.3
121		September 29, 2022	119	289	222	419		1067.2	10.7	10.0
122		September 30, 2022	120	283	222	411		1075.1	10.7	9.8
123		October 3, 2022	121	277	222	403		1082.9	10.7	9.6
124		October 4, 2022	122				392	1082.9	10.7	9.6
125		October 5, 2022	123				381	1082.9	10.7	9.6
126		October 6, 2022	124				370	1082.9	10.7	9.6
127		October 11, 2022	125				359	1082.9	10.7	9.6
157		November 22, 2022	155				33	1082.9	10.7	9.6
158		November 23, 2022	156				11	1082.9	10.7	9.6
159		November 24, 2022	157				0	1082.9	10.7	9.6

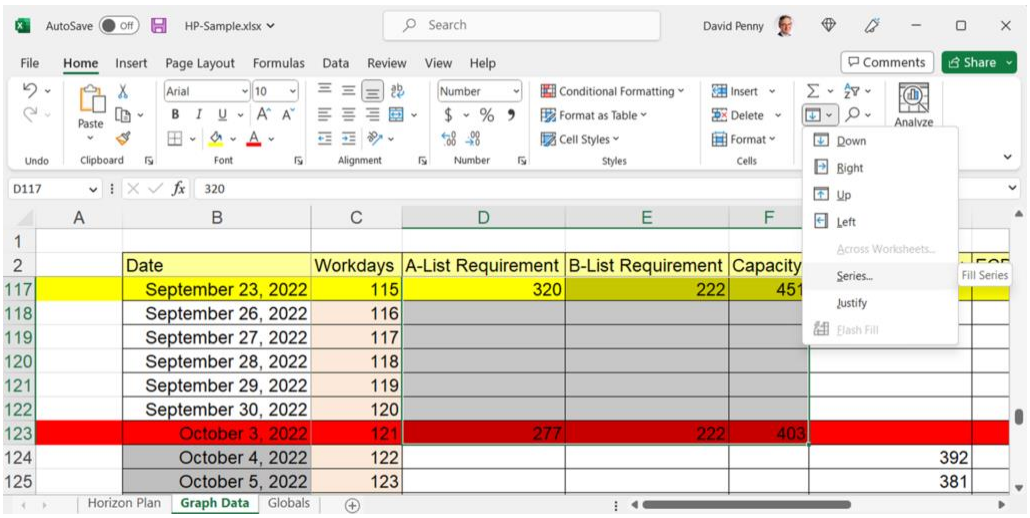
Release Plan | Graph Data | Globals

[Note the view above has certain rows hidden to enable it to fit on a screenshot]

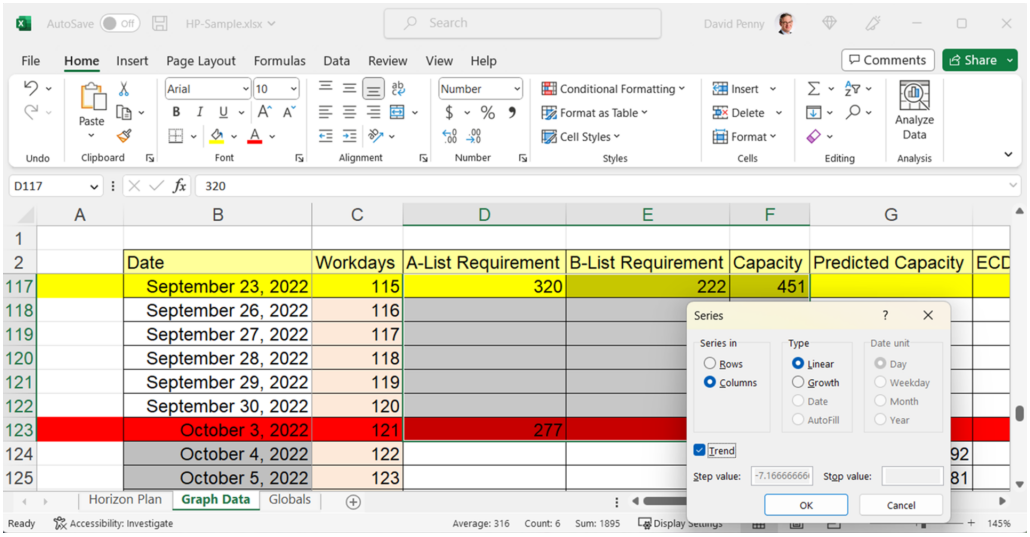
The Horizon Plan (HP) should be updated on as regular a basis as possible (some managers do it daily, but weekly at a minimum). The update should be saved in a common directory under a new name that incorporates the as-at date in a sortable yyyy-mm-dd format. This way, historical HPs can be compared to recent HPs to understand what caused the plan to to get to where it is (*e.g.,* compare the original estimates to the evolving estimates).

In addition to this, each plan update extends the history of key values in order to provide the historical graphs in sections C and D.

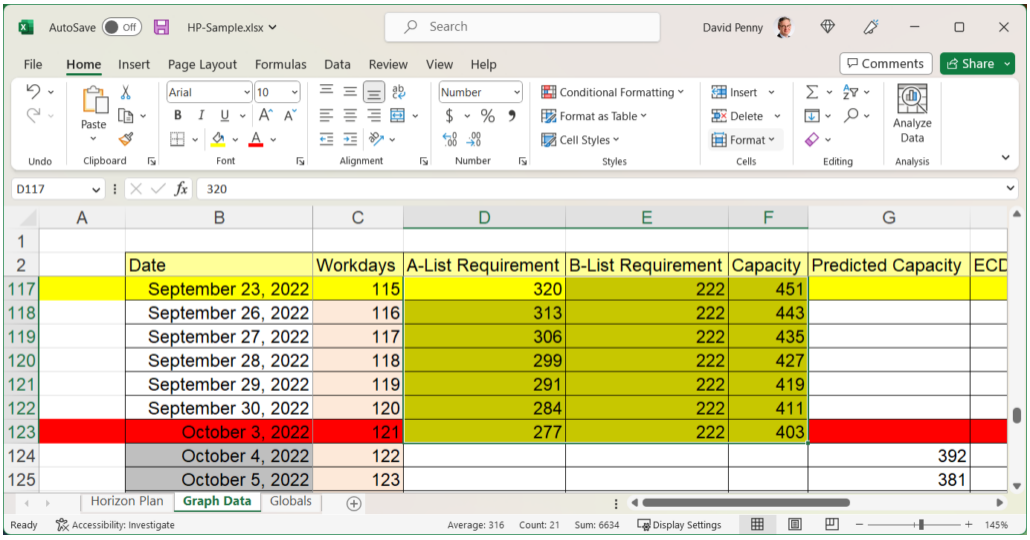
Initially, this tab is setup to include every workday (and only workdays) from **fork** to **dcut**. Once a plan update is made (such as the red row above), certain key values from the current plan are copied into this row. This will typically leave a number of blank entries between the previous update and the current one (e.g., cells D118:F122 in the below). These can be filled in using a Fill Series. Select the larger area contain the start cells and the end cells (D117:F123) and then Fill -> Series… from the Home ribbon.



This brings up the Fill Series dialog box.

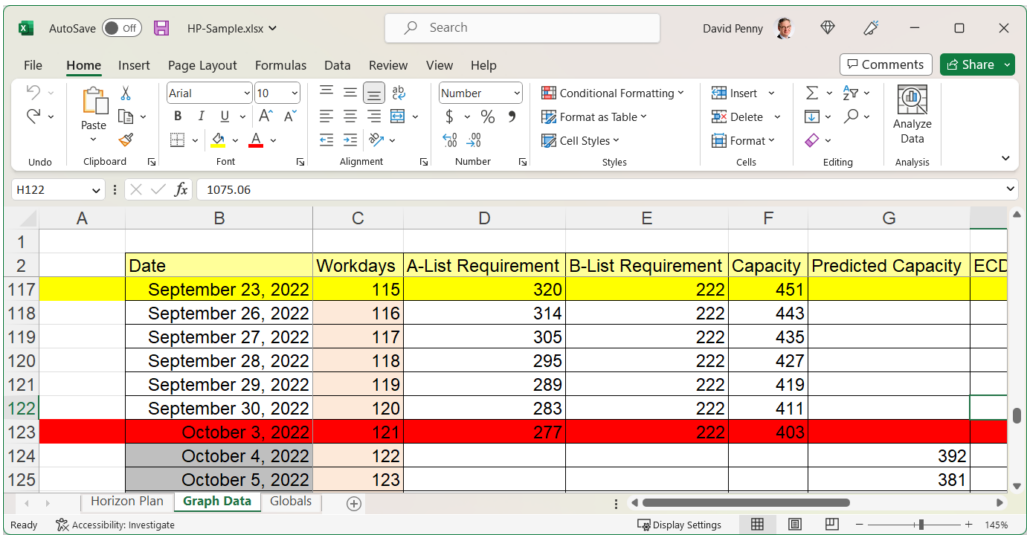


Choose Series in Columns, Type Linear, tick Trend, and hit OK. This fills in the missing values with a linear interpolation.



	Date	Workdays	A-List Requirement	B-List Requirement	Capacity	Predicted Capacity	ECD
117	September 23, 2022	115	320	222	451		
118	September 26, 2022	116	313	222	443		
119	September 27, 2022	117	306	222	435		
120	September 28, 2022	118	299	222	427		
121	September 29, 2022	119	291	222	419		
122	September 30, 2022	120	284	222	411		
123	October 3, 2022	121	277	222	403		
124	October 4, 2022	122				392	
125	October 5, 2022	123				381	

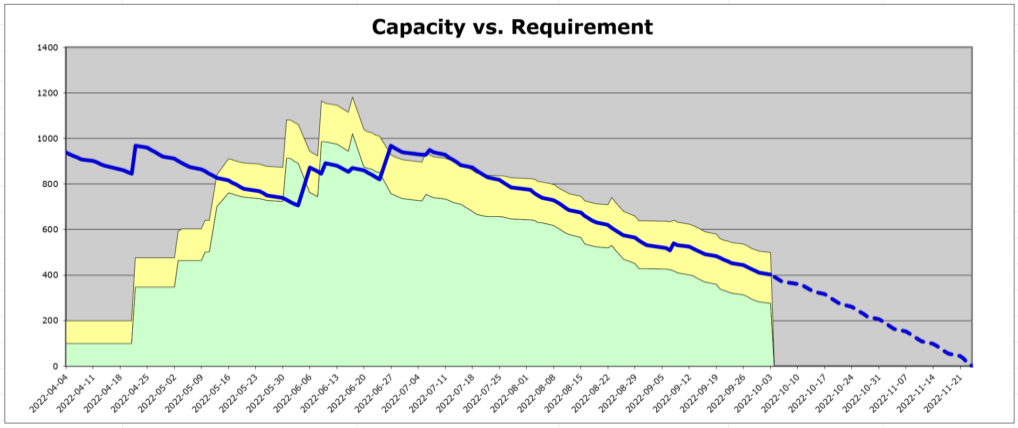
Reformat the cells to indicate the current update in red and the earlier updates in yellow.



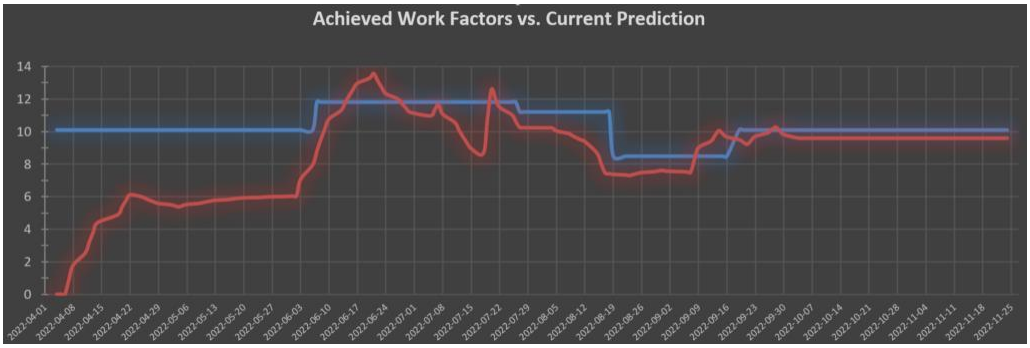
	Date	Workdays	A-List Requirement	B-List Requirement	Capacity	Predicted Capacity	ECD
117	September 23, 2022	115	320	222	451		
118	September 26, 2022	116	314	222	443		
119	September 27, 2022	117	305	222	435		
120	September 28, 2022	118	295	222	427		
121	September 29, 2022	119	289	222	419		
122	September 30, 2022	120	283	222	411		
123	October 3, 2022	121	277	222	403		
124	October 4, 2022	122				392	
125	October 5, 2022	123				381	

Blank the Predicted Capacity up to the current as-at (it is used to create the dotted blue line on the graph and is not needed before as-at).

The values that are copied in on each update are **A-List Requirement** (from B18), **B-List Requirement** (from F123), **Capacity** (from B17), **ECDs logged to date** (from B14), and **Estimated ECDs** (from B15). The first three plus the Predicted Capacity are used to create the chart in section C.



The ECDs logged to date and Estimated ECDs is used to create the chart in section D.



The blue line is **Estimated ECDs** plotted. I do not trend this series for the infill, I just carry the previous value forward, and update the values at and after the as-at date to be the current value.

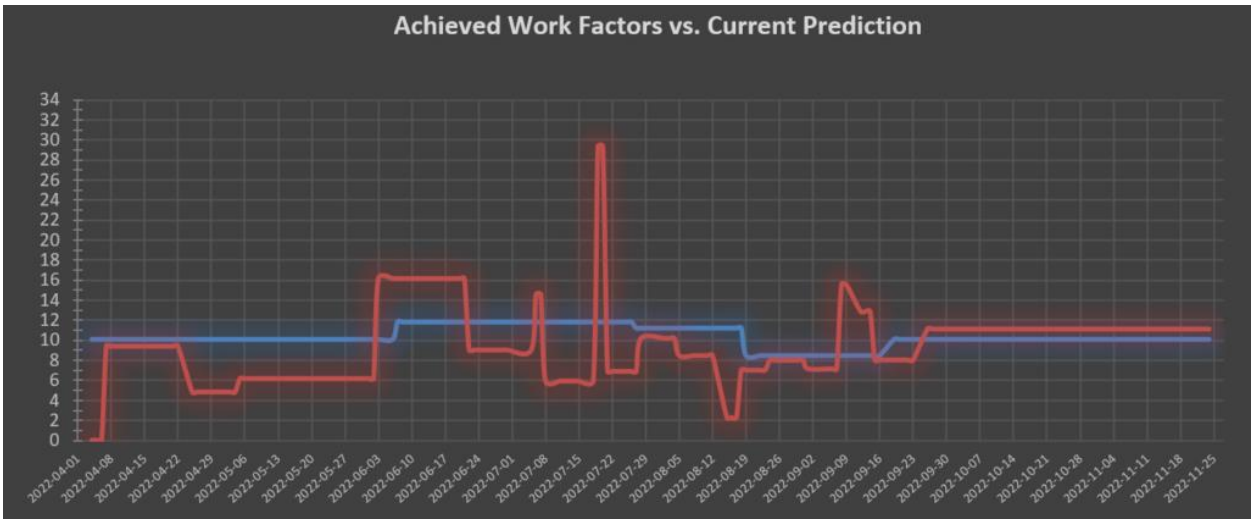
The red line is **Averaged Daily ECDs** plotted. This is a column that's computed from **ECDs logged to date** and **exp decay factor** (in cell J1).

I do a linear interpolation for the **ECDs logged to date** as shown above. Because each row is the next workday, the difference between any two adjacent values is the instantaneous (interpolated) ECDs logged that workday. You can see that by setting the exponential decay factor to 1 which means only the latest value is included in the "average" and no weight at all is given to history.

	A	B	C	H	I	J
1					exp decay factor:	1.00
2		Date	Workdays	ECDs logged to date	Estimated ECDs	Averaged Daily ECDs
114		September 20, 2022	112	992.3	10.1	8.0
115		September 21, 2022	113	1000.3	10.1	8.0
116		September 22, 2022	114	1008.3	10.1	8.0
117		September 23, 2022	115	1016.3	10.1	8.0
118		September 26, 2022	116	1027.4	10.1	11.1
119		September 27, 2022	117	1038.5	10.1	11.1
120		September 28, 2022	118	1049.6	10.1	11.1
121		September 29, 2022	119	1060.7	10.1	11.1
122		September 30, 2022	120	1071.8	10.1	11.1
123		October 3, 2022	121	1082.9	10.1	11.1
124		October 4, 2022	122	1082.9	10.1	11.1
125		October 5, 2022	123	1082.9	10.1	11.1
126		October 6, 2022	124	1082.9	10.1	11.1

Horizon Plan | Graph Data | Globals

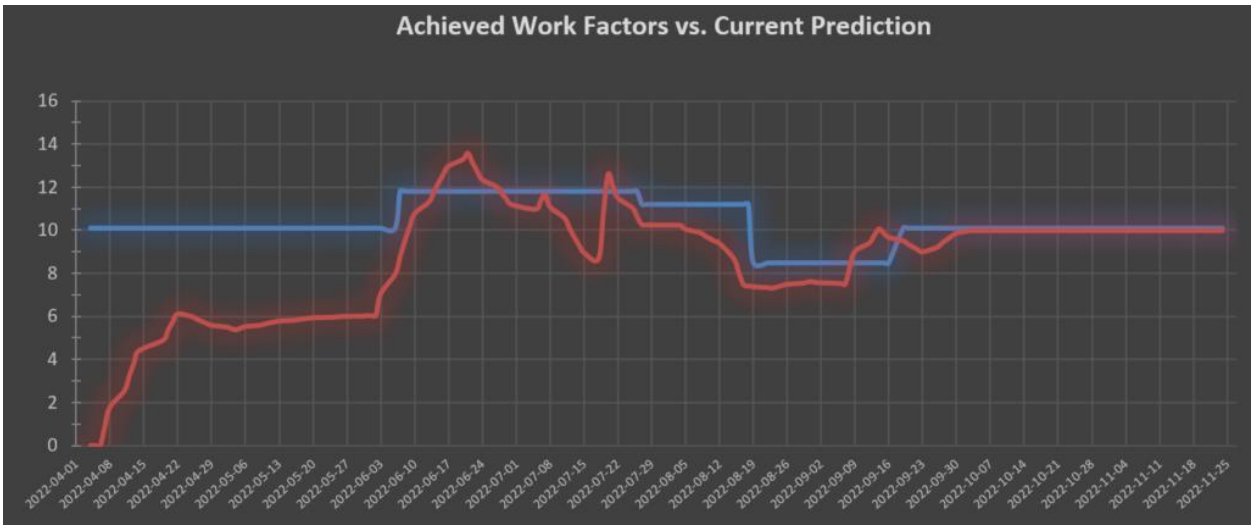
We can see that between Sep 23 and Oct 3 there were 11.1 ECDs logged per workday on average (they are identical because of the linear interpolation). The last value in the red row is just carried forward to the end by a formula. Keeping this factor at 1, we would get a very bumpy graph!



Some large spikes are because an update was made that added a lot of historical time tracked in on one day that was missed in earlier updates. This is the reason for smoothing these values with a factor of 0.1.

	A	B	C	H	I	J
1					exp decay factor:	0.10
2		Date	Workdays	ECDs logged to date	Estimated ECDs	Averaged Daily ECDs
114		September 20, 2022	112	992.3	10.1	9.4
115		September 21, 2022	113	1000.3	10.1	9.2
116		September 22, 2022	114	1008.3	10.1	9.1
117		September 23, 2022	115	1016.3	10.1	9.0
118		September 26, 2022	116	1027.4	10.1	9.2
119		September 27, 2022	117	1038.5	10.1	9.4
120		September 28, 2022	118	1049.6	10.1	9.6
121		September 29, 2022	119	1060.7	10.1	9.7
122		September 30, 2022	120	1071.8	10.1	9.9
123		October 3, 2022	121	1082.9	10.1	10.0
124		October 4, 2022	122	1082.9	10.1	10.0
125		October 5, 2022	123	1082.9	10.1	10.0
126		October 6, 2022	124	1082.9	10.1	10.0

With 0.1, the most current value is only given 10% of the weight in the average, and the previous average is given 90%. So every value back to the beginning is somewhere in the average, it's weight just drops off exponentially. This creates a more informative smoothed chart.



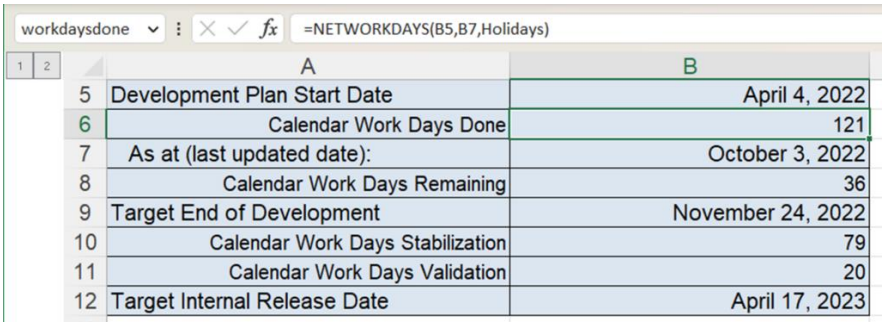
Globals Tab

	B	C
3	Holidays:	
4	2022-04-15	Good Friday
5	2022-04-18	company day
6	2022-05-23	Victoria Day
7	2022-05-27	company day
8	2022-07-01	Canada Day
9	2022-07-04	company day
10	2022-07-29	company day
11	2022-08-01	Civic holiday
12	2022-09-02	company day
13	2022-09-05	Labour Day
14	2022-10-07	company day
15	2022-10-10	Thanksgiving Day
16	2022-12-25	Christmas Day
17	2022-12-26	Boxing Day
18	2023-01-01	New Year's Day
19	2023-02-20	Family Day
20	2023-04-07	Good Friday
21	2023-04-10	company day
22	2023-05-22	Victoria Day

Horizon Plan | Graph Data | Globals

This tab is intended to collect global values referenced from elsewhere in the spreadsheet. Currently it only includes the holiday schedule which is used to compute the workdays net of the listed holidays between two dates.

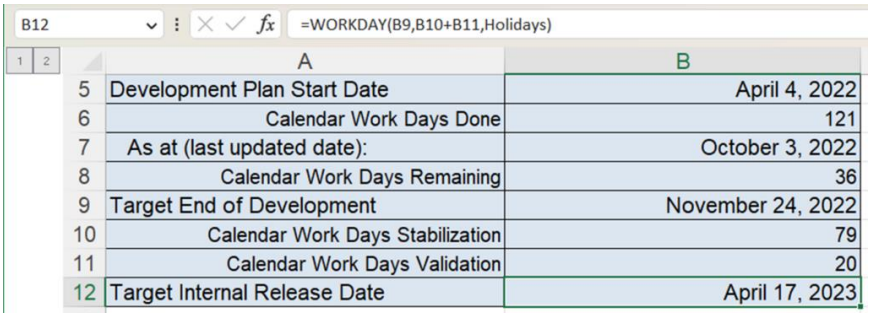
A good example is cell B6 in the Dates section whose NETWORKDAYS formula is shown referencing the range “Holidays” (Globals!B4:B22).



workdaysdone | =NETWORKDAYS(B5,B7,Holidays)

	A	B
5	Development Plan Start Date	April 4, 2022
6	Calendar Work Days Done	121
7	As at (last updated date):	October 3, 2022
8	Calendar Work Days Remaining	36
9	Target End of Development	November 24, 2022
10	Calendar Work Days Stabilization	79
11	Calendar Work Days Validation	20
12	Target Internal Release Date	April 17, 2023

The opposite use is for computing the target end date in cell B12 after adding the stabilization and validation days.



B12 | =WORKDAY(B9,B10+B11,Holidays)

	A	B
5	Development Plan Start Date	April 4, 2022
6	Calendar Work Days Done	121
7	As at (last updated date):	October 3, 2022
8	Calendar Work Days Remaining	36
9	Target End of Development	November 24, 2022
10	Calendar Work Days Stabilization	79
11	Calendar Work Days Validation	20
12	Target Internal Release Date	April 17, 2023