INFICON
Inspired by visions. Proven by success.

FabTime® Newsletter

Volume 27, No. 2 **June 2026**

Information

Publisher: Acquired by INFICON in early 2024, FabTime has been helping fabs with cycle time and performance improvement since 1999. FabTime's flexible reporting software, cycle time management course, and this newsletter are now part of the INFICON Intelligent Manufacturing Systems (IMS) group.

Editor: Jennifer Robinson, Cycle Time Evangelist for INFICON

Date: Tuesday, June 23, 2026 – Vol. 27, No. 2

Keywords: Queueing models; tool qualification; maintenance metrics; line balance; setup time

Table of Contents

Welcome

Welcome to Vol. 27, No. 2 of the FabTime Cycle Time Newsletter. For our main topic, we examine the cycle time benefits of qualifying a second tool when cross-qualification requires significant setup time. We also share discussion with subscribers about maintenance metrics, line balance, open lot cycle time, and more. We have announcements about FabTime version 2604 and the upcoming Fab Owners Alliance meeting in Maryland.

Thanks for reading! – Jennifer

Community News/Announcements

In this month's announcements, we share release notes for FabTime software version 2604, with an example from the new query-based Chart Assistant. We also have an announcement about the upcoming FOA meeting.

FabTime software version 2604 released to customers

The latest version of the FabTime flexible reporting software has been released and is installed on several customer servers. Contact your FabTime support engineer for information about getting this update installed at your fab. Highlights of this version include:

🔷 **Major Updates**

- Natural-language Chart Assistant, allowing users to request a chart by describing it via text
- Left-pane filter revamp, making chart filtering and customization easier for all users
- On-chart editing of custom titles
- Checkboxes for selecting filter values
- FabTime Oracle chart parity with FabTime SQL Server

🔹 **Minor Updates**

- Percentile line on Elapsed CT Chart
- Scaled turns for filtered shifts
- Planned moves added to Starts Lot List
- Chart title wrapping
- Chart thumbnail fallback

🐛 **Bug Fixes**

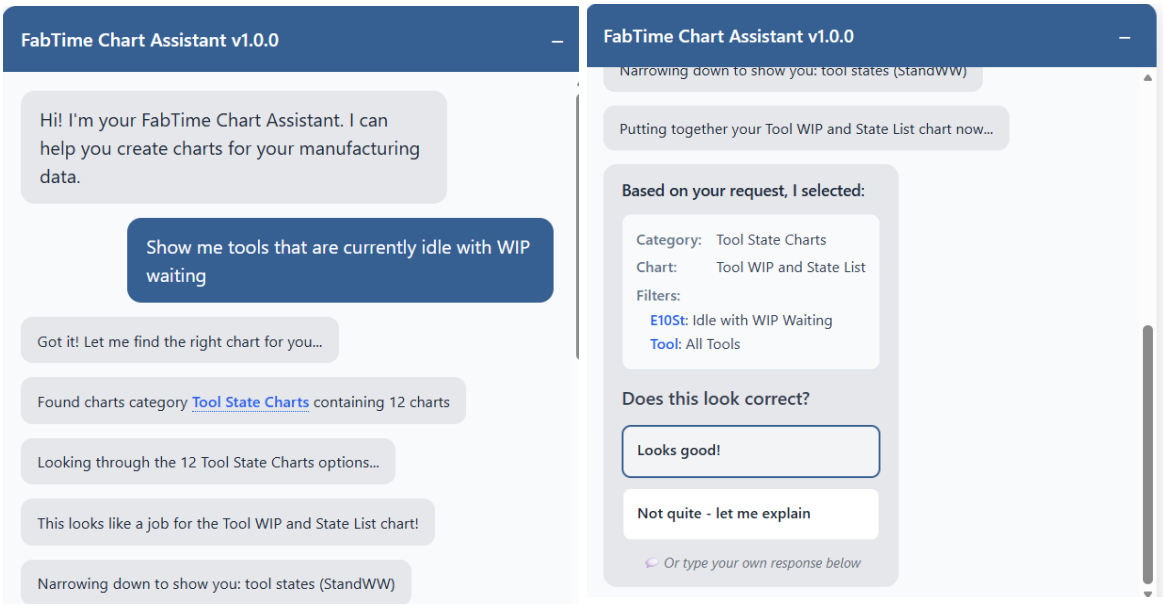
- Autologin with Windows authentication
- WIP pareto goal proration
- WIP Hours bad time U/M labels

Chart Assistant Example:

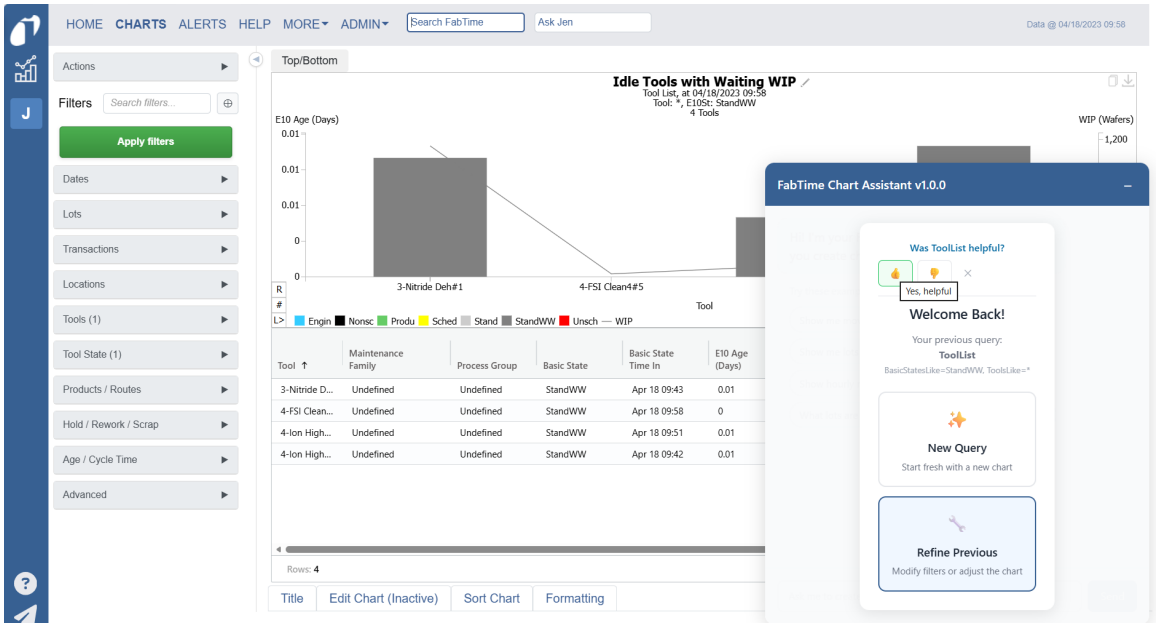
Here's a detailed example of using the new Chart Assistant to generate a chart that shows **all tools currently idle with WIP waiting**.

1. Input the requested chart type in the chat window ("Show me tools that are currently idle with WIP waiting").

2. Approve or refine the proposed selection by the Chart Assistant.



3. View the resulting chart. Note that the text from the original request is used, by default, to create a custom chart title. This custom title can be edited as needed by clicking within the title.



The next SEMI Fab Owners Alliance meeting is in late July

The next FOA meeting will be held July 28-30 at the Northrop Grumman ATL facility in Maryland. The INFICON team attending will include **Jennifer Robinson, John Behnke,** and **Mario Faria**. We hope to see you there!

Interesting Reads

Semiconductor Insights and Productivity Tips

Recent articles shared on Jennifer's LinkedIn include:

- A recent study from INFICON that used 1.5 years of real factory data to explore the factors driving cycle time in a high mix fab. The study found that "One clear takeaway is that process time variability plays a larger role than the earlier analytical model suggested. In contrast, availability and utilization exert less influence." The conclusion from Gero Grau's recent presentation of this work at the apc|m Europe Conference concluded that "Variability reduction is the most effective lever for improving cycle time performance across the fab." See the article on INFICON's website for details.
- A TrendForce News article about how "the world's first industrial wafer fab dedicated to 6-inch InP photonic chips has officially broken ground in Europe." The project "is jointly supported by the Dutch research organization TNO, Eindhoven University of Technology (TU/e), PhotonDelta, SMART Photonics and the Eindhoven High Tech Campus."
- A WSJ article about how TSMC remains optimistic about AI-driven business, raising their 2026 revenue forecast to > 30% growth and downplaying the risk of supply chain disruptions due to the war in Iran. "TSMC's chief executive, C.C. Wei, said the company had double-checked with customers about AI demand and they reassured the chip maker that it was still strong. That led TSMC to accelerate investments such as building more clean rooms, he said."
- A fun WSJ piece about the high demand for AMSL DUV machine Lego sets. On a more serious note, the WSJ also reports on ASML's race to keep up with demand for EUV machines, given the complexity of manufacturing them and TSMC's recent decision not to buy ASML's next-generation tools because they are too expensive.
- Also in the WSJ, a rare mainstream media article about defects in computer chips, and how Apple is able to sell cheaper devices by using chips with slight defects that might otherwise be thrown out.

For more industry news, connect with Jennifer on LinkedIn.

Subscriber Discussion Forum

We have subscriber discussion about maintenance metrics, open lot cycle time, finding data in FabTime, creating line balance charts, applying dynamic x-factor. If there is a topic you've been wondering about, please let us know.

Maintenance Metrics

At one of our customer sites, a **maintenance manager** requested a new metric: **scheduled downtime / (scheduled + unscheduled downtime)**. The idea is to drive this metric to 100%. This metric would be reported as 100% when the denominator is zero.

This yet-to-be-named metric seems similar to **M-Ratio**, which (as used in the semiconductor industry) is the ratio of scheduled downtime to unscheduled downtime. A high M-Ratio (≥ 4.0) is considered desirable, meaning that most downtime is preventative rather than unplanned.

M-Ratio approaches infinity as the quantity of unscheduled downtime gets very small. The scheduled maintenance ratio proposed by our customer seems more intuitive to track on an ongoing basis, since it calls for targeting 100% instead of “some large value, ideally greater than four.”

We are considering adding both the new scheduled downtime / total downtime metric and M-Ratio as columns on the Tool State trend and pareto charts in FabTime. This would enable end users to quickly create charts to track these metrics via the Edit Chart capability in FabTime.

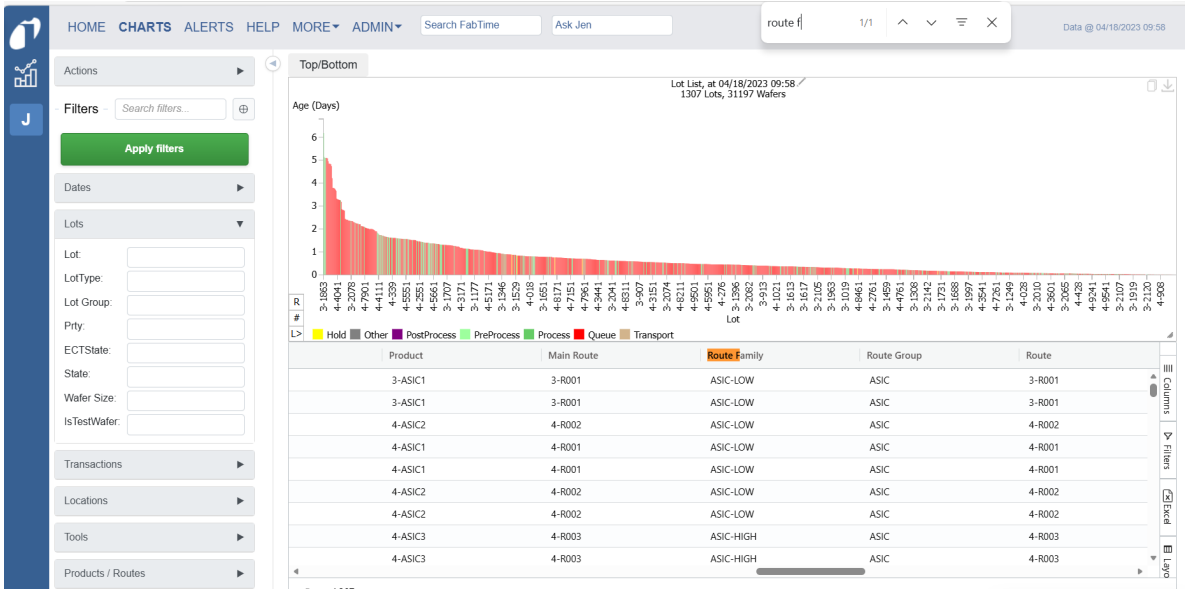
Do you think that this new scheduled to total downtime ratio would be useful and if so, what do you think it should be called? We are considering PM-Ratio. Do you favor using this new PM-Ratio or M-Ratio, or both? Share your thoughts here.

FabTime Software Tip: Use Ctrl+F to find columns in the data table

The AG-Grid data table associated with each chart in FabTime offers a robust set of values that can be used to sort or filter the data displayed on the chart. In some cases, especially where customers have extensive custom attributes, these data tables can have dozens of columns. While AG-Grid has considerable native functionality, we wanted to share a quick tip, courtesy of **Elaine Jacobson**, our recently retired and much missed Program Manager of Customer Success, for finding columns of interest. Simply use the Windows Ctrl+F keyboard shortcut, just as you might on any webpage, to search for your target.

For example, suppose you are looking at a WIP Lot List chart, and you would like to sort it by the Route Family. A quick Ctrl+F for “route f” will cause your browser to highlight the “Route Family” column, as shown below. You can then click in the column header to sort by that column.

You can also search for values within the data table and/or the displayed chart axis labels. In the above example, we could search for a Lot ID or for a Hold Code. We hope that FabTime software users find this tip useful.



FabTime software tip: create a line balance chart

A common goal of wafer fab managers is a relatively balanced line. This is not to say that each tool needs to have the same quantity of WIP waiting at all times. There will usually be a few constraint

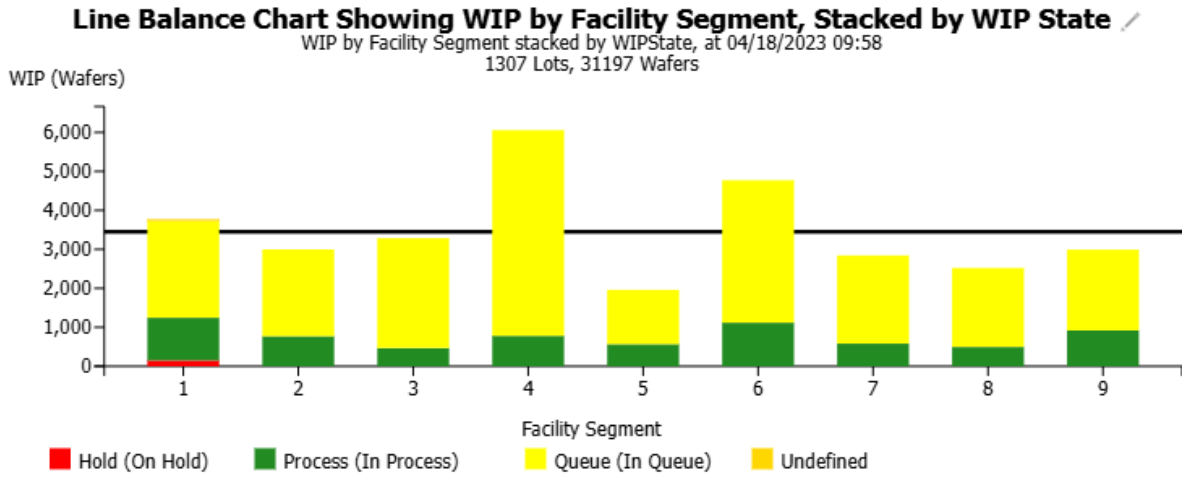
tools that carry higher levels of WIP, on average. Rather, the goal of line balance, in steady state, is to have WIP spread out along each process flow vs. bunched up at steps early or late in the flow, We also want to avoid having all of the WIP at one tool type, of course, but with an acceptance of certain bottlenecks carrying more WIP than other tools.

To generate a Line Balance chart in FabTime, start with either the WIP Pareto chart or the WIP Stacked Pareto chart. In either case, set the "Slice" variable to whatever attribute your fab uses to break up the process flows into linear groups of steps. This may be called Segment, Facility Segment, Sub-Segment, Route Segment, or something else selected by your site. The idea is that the bucket representing each column consists of roughly the same number of steps. Often, week-long buckets are used as Segments, with optional smaller Sub-Segment buckets. It may be appropriate to filter this chart at your site by process flow, or to exclude certain types of WIP (monitors, engineering lots, etc.).

If you are using the WIP Stacked Pareto chart, you can also select a meaningful "Cross" variable for stacking. WIP state, product family, and module are commonly used.

Optionally, use the "Stripe" option in the Formatting section below the chart to add a line that represents the average WIP across all included segments. This is what the WIP would be by segment if the line was perfectly balanced. An example is shown below, stacked by WIP state. We see that segments 4 and 6 have excess WIP, while segment 5 is starved. The stripe value was calculated by dividing the total WIP (shown in the chart header) by nine (the number of segments).

What buckets do you use for the x-axis when you look at line balance? What do you stack or filter this chart by at your site? Share your thoughts here.



Open Lot Cycle Time definitions

A **subscriber and cycle time class participant** wrote to say that there was debate within her organization about the correct definition for Open Lot Cycle Time. Possible definitions being considered by the site included:

- The time it takes to get from one identified step to a later identified step. For example, the time it takes to get from step 1500.1200 to 3001.1000, which could represent a specific module.
- The number of steps completed in a certain time frame. For example, how many steps were completed from Jan 1, 2026 to Jan 15, 2026.

- In the case of open lot DPML calculation, average the number of mask levels completed for all lots over a period of time.
- And the latest from an AI chat engine is the time it takes for a specific lot to go from start of line to fab out minus all the time spent on hold.

Response from Jennifer: In our FabTime reporting module, Open Lot Cycle Time is defined as follows:

1. Pull all records that match filter and Lot has already started.
2. Compute current factory age = AsOfTime - factory start time.
3. Compute remaining planned cycle time based on RTG_ROUTE_STEPS_PLUS.TARGET_STEP_SEC_TO_EOL
    1. If Lot is currently on hold or rework, find last NonRework record.
    2. If step not found or if step is prior to start time, remove the lot from list
4. Compute Open Lot Cycle Time = (Age from start to last Move) + remaining planned cycle time.

This is essentially current CT + remaining planned CT, calculated for each lot individually and then aggregated as needed. Other metrics for estimating future cycle time based on current performance in FabTime include dynamic cycle time (DCT) and dynamic x-factor (DXF). See Forward-Looking Cycle Time Metrics (Issue 24.03) for definitions of DCT and DXF.

In our FPS Cycle Time Analyzer, open lot (AKA dynamic) cycle time is calculated differently. For each route step, we compute a weighted average cycle time based on moves for the route step combination over the selected time period. For a given route, this is summed across all the steps for that route. A weighted average across routes is computed for the factory. Fill-in logic is used if there have not been enough moves completed at the route-step level.

There are benefits to each of these methods for different purposes. **Readers, how does your fab define Open Lot Cycle Time?** Share your thoughts here.

Is dynamic x-factor useful for small fabs with variable WIP?

While discussing open lot cycle time metrics, the customer cited above added this comment on dynamic x-factor:

"I haven't found dynamic x-factor to be an insightful metric for us because it doesn't correlate well with improved productivity. I believe this is because we have an inconsistent amount of available WIP. When holds are high, DXF is high because there's not much WIP to keep track of. But tool loading is low because there's not much WIP. We've found the raw number of wafers loaded to be a better metric than DXF".

Response from Jennifer: This point arose recently at a customer site that currently has low levels of WIP running. If the quantity of WIP running on tools frequently drops to a very low value, this will drive spikes in the DXF calculation (DXF = Total WIP / WIP currently running on tools). It seems to me that using raw number of wafers loaded instead of using DXF makes sense in this situation. Raw number of wafers loaded won't give you a forward look at cycle time, but it will tell you if your fab is doing a good job keeping tools busy. This metric will, of course, drop down at shift change if your fab isn't doing a good job keeping tools running at that time. **Do any other subscribers have real-world experiences to share with using DXF or other forward-looking cycle time metrics?**

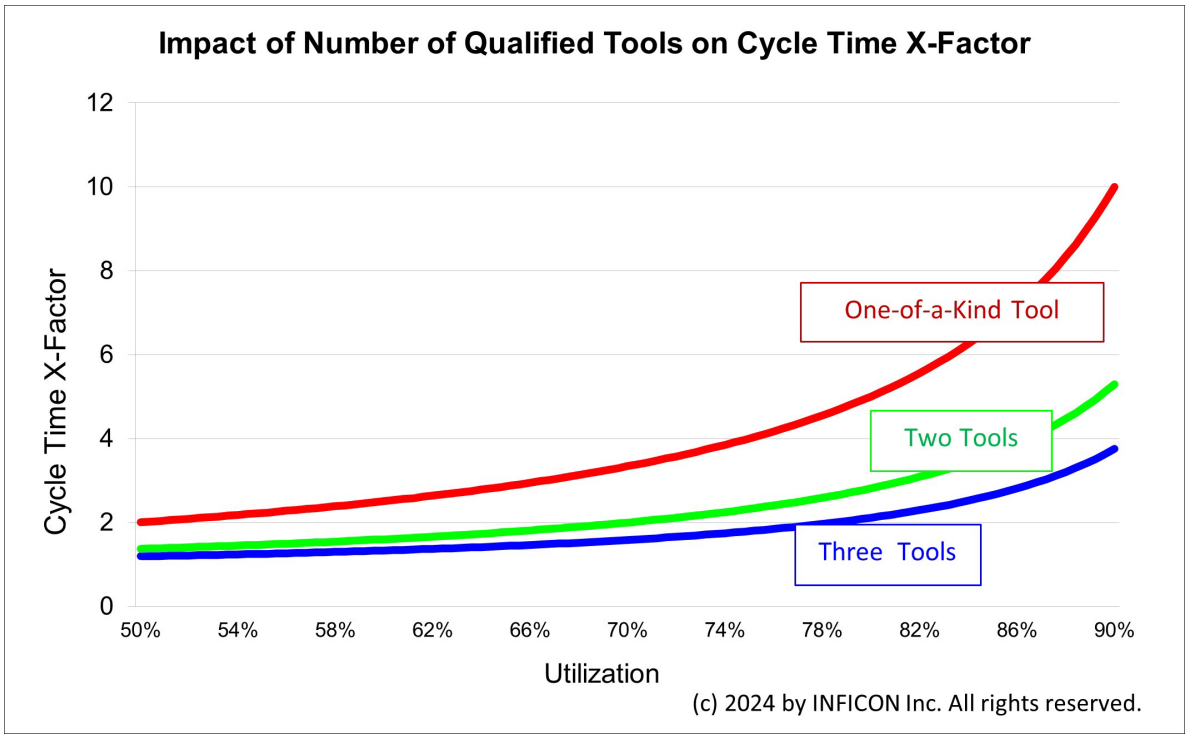
Main Article: Setup time, tool qualification, and cycle time

By Jennifer Robinson

One of the top recommendations we've made in this newsletter (see Issue 20.05, for example), and in our cycle time management course, is for fabs to move from one qualified tool to two wherever possible. We've demonstrated that **qualifying a second tool decreases average cycle time per visit by approximately 50%** (with smaller decreases as we go to three qualified tools, four qualified tools, etc.). The intuitive explanation for this is that whenever lots are waiting in queue for a single path operation, they are subject to all variability associated with the tool in question, including long unplanned downtimes. When there is a backup tool, the odds of both tools having a long adverse event that holds up the lot are much smaller.

However, nearly every time I present this result, someone in the audience says something like: "Sure, but what about the extra setup time if we have two cross-qualified tools vs. two dedicated tools?"

Here I usually point to the graph below and note that even if we lose some utilization to having to do extra setups, we're still better off being on the green curve at a higher utilization than. being on the red curve. If we wanted to constrain the cycle time per visit to 4X, we'd have to keep the utilization to ~72% on the red (fully dedicated) curve. But if we could move to the green curve, we could increase utilization to ~84% and achieve the same 4X cycle time. We could do quite a few extra quals within that 12% difference.

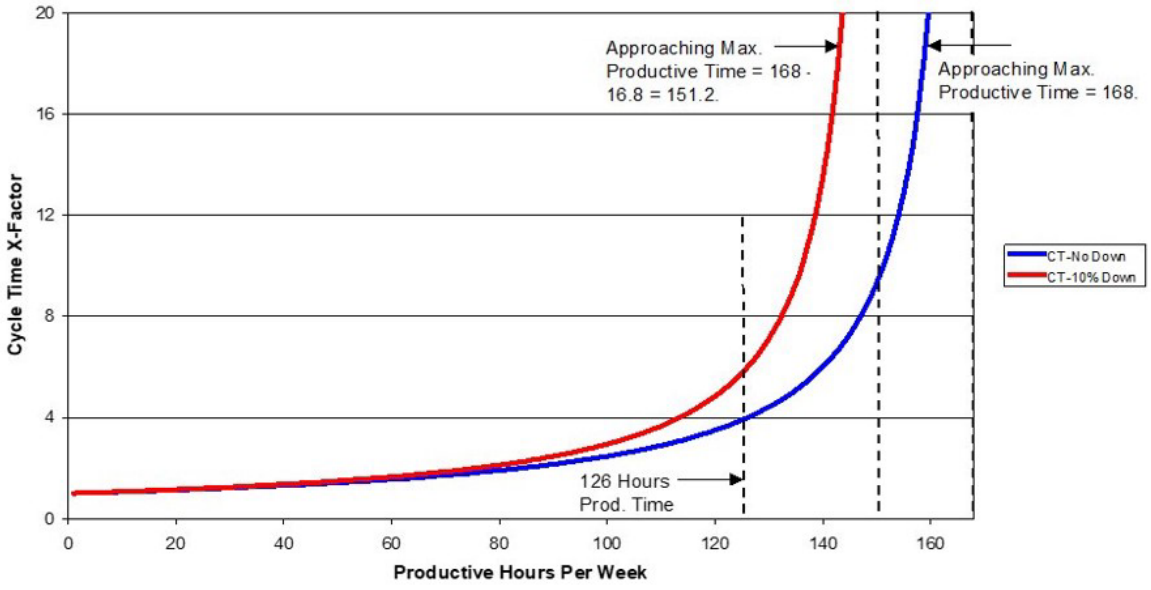


This explanation makes intuitive sense, but it's a bit tricky to visualize when looking at the graph. I decided it was high time I created a graphic to show **the positive impact of qualifying a second tool even if that cross-qualification requires some additional setup time**. What I discovered was that the outcome is sensitive to the quantity of additional setup time (of course) and where we are on the utilization curve to begin with.

How the operating curve spreadsheet works, and why we can't just add some extra downtime to one scenario to represent extra setup time

The graph above was created using the FabTime Operating Curve Generator, available for download from INFICON's website. Users can enter values for up to three scenarios. The spreadsheet uses queueing models to generate the resulting operating curves. Inputs include number of tools, percent downtime, and arrival, processing and downtime variability measures. One input that doesn't make sense to vary across scenarios is percent downtime. This is because utilization is defined relative to available time. Changing percent downtime between scenarios would require the different scenarios to be on different x-axis scales, with different definitions of 100% utilization.

For example, the graph below (from Issue 25.04) shows an operating curve with no downtime (the blue graph) compared to one with 10% downtime (the red graph). We can't put "Utilization" on the x-axis, because these scenarios have different utilization values at the same number of productive hours per week. The shorter dashed line is shown at 126 hours of productive time. For the blue graph, this is 126/168 = .75, or 75% utilization. For the red graph, where the maximum productive time is only 151.2 hours, the utilization at 126 hours is 126/151.2 = 83.3%.



This means that I couldn't just compare, say, 1 dedicated tool at 20% downtime with 2 cross-qualified tools at 25% downtime, with the extra 5% representing setup time. I had to do more of a work-around.

What I did instead to put two different scenarios (one with two dedicated tools and one with two cross-qualified tools and 5% of total time spent doing extra quals) on one graph

Here's what I did:

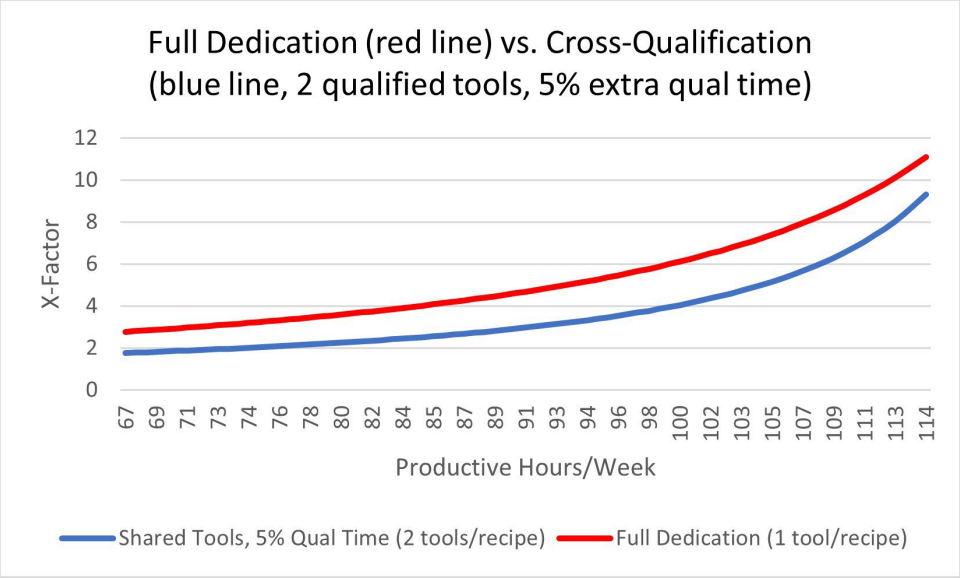
- Used the operating curve spreadsheet to generate the operating curve for a single tool with 20% downtime (33.6 hours/week) and moderate variability. This is the base scenario for full dedication, when there is only one tool per recipe.

- Converted the utilization values for that operating curve into productive hours/week for each point on the curve.
- Calculated a revised standby time at each point if 8.4 hours/week (5%) qual time was taken out of standby time. (168 hours– 33.6 hours of downtime – 8.4 hours of qual time – productive hours)
- Calculated a revised utilization % based on the original productive hours and the revised standby hours. (Productive / (Productive + Revised Standby)
- Filled in the revised utilization numbers in the operating curve calculations for a scenario with two tools (cross-qualified case), using the same 20% downtime and other variability values.
- Graphed productive hours vs. cycle time x-factor for the two scenarios onto the same graph in Excel.

What the results look like at low to moderate utilization

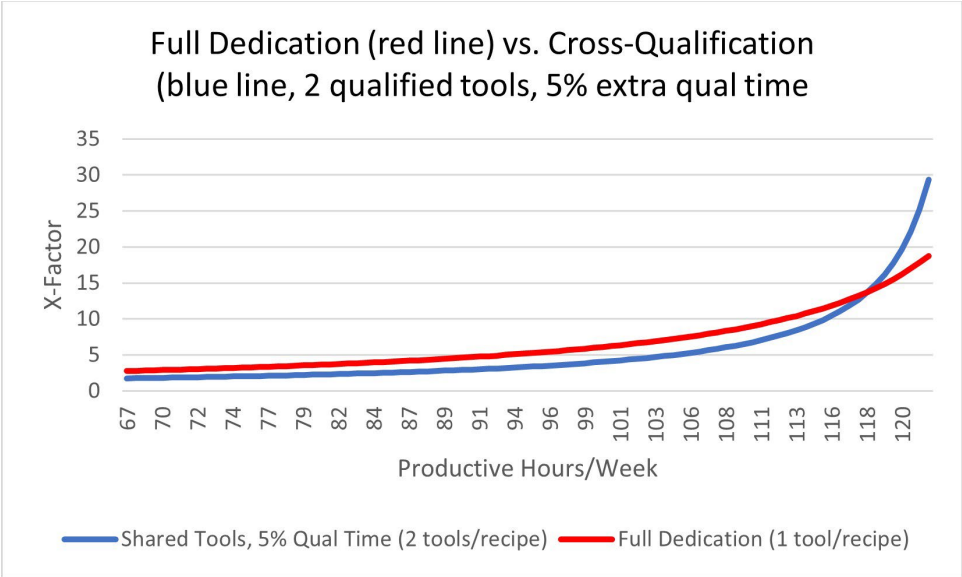
Here's the resulting graph, up to the equivalent of 85% utilization for the dedicated case. Note there is 20% downtime in both cases (MTBF = 24 hours, C_r = 1.0). The maximum productive hours = 168 – 33.6 = 134.4. The right-most point shown on the graph is 114/134.4 ~= 85% for the red curve.

On this graph, the average cycle time reduction from the red curve (full dedication) to the blue curve (cross-qualification with 2 tools, 5% qual time) is ~33%. This reduction ranges from about 37% at low utilization down to ~12% at 85% utilization.



What happens when we push the utilization higher, closer to 100%?

Here's what happens when we push the utilization a bit higher, to 122 productive hours/week. This works out to 90.8% utilization for the red curve. For the blue curve, with 8.4 fewer available hours per week, the utilization at 122 productive hours = 122/(134.4 – 8.4) = 122/126 = 96.8%. Because the blue curve has a higher utilization for the same number of productive hours, we reach the steep part of the operating curve sooner and more dramatically. The curves cross, as shown.



Comparing these results to the prior "50% reduction from qualifying a second tool" framework, with some sensitivity analysis

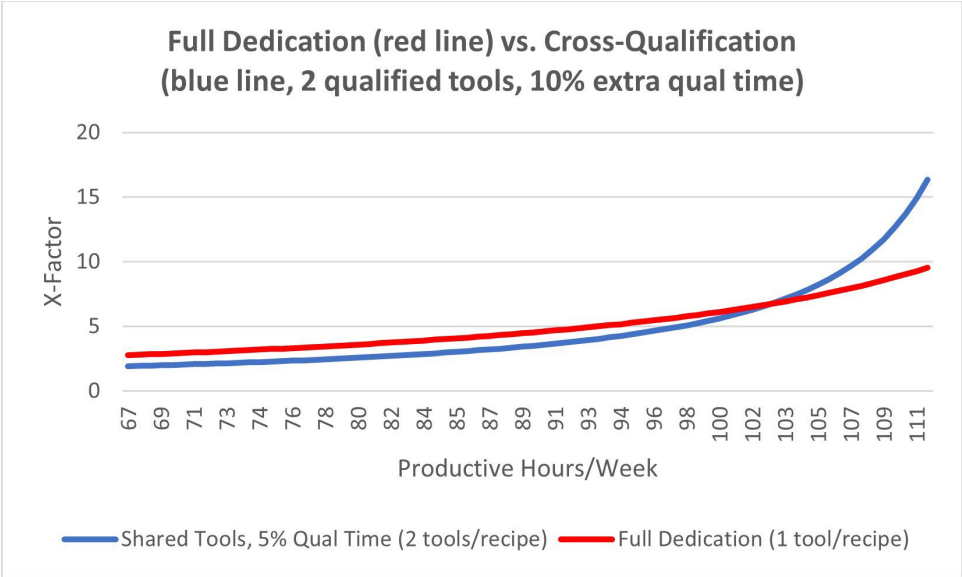
With no extra setup/qual time, the cycle time x-factor for the cross-qualified case (with a backup tool) is approximately 50% lower than for the dedicated case (only one tool per recipe). When we include 5% setup time (8.4 hours/week), the cycle time improvement from adding a second tool drops to ~33%. This improvement only applies up to ~85% utilization for the dedicated (red) curve. Above that, the extra setup time takes the blue (cross-qualified) graph into the steep part of the operating curve. We see worse performance for the cross-qualified system due to the dominant effect of utilization.

Bottom Line: Going from one qualified tool to two is still helpful for reducing cycle time, even in the presence of additional setup time, at least at low to moderate utilizations. Once we get to the steep part of the operating curve, the impact of the extra qual time on the utilization outweighs the benefits of the redundancy (the curves cross).

Exact results will, of course, depend on how much variability is present. I looked at adding more downtime variability (changing the coefficient of variation of repair time to a higher value). In that situation, the red and blue curves moved a bit further apart at the lower end, showing closer to a 40% reduction in cycle time going from the dedicated to the cross-qualified case. But the curves (not shown here) still crossed at a similar utilization.

I also looked at a scenario with 10% setup time instead of 5%. The red curve (dedicated case) was still higher than the blue curve by about 33% (e.g. 4x vs. 3x at the same number of productive hours) at lower utilizations. However, the curves crossed sooner (at about 103 vs. 119 productive hours). That graph is shown below.

I also considered modeling setup time that varies according to the utilization. However, it wasn't clear to me what form that variation should take. Nor did this seem likely to have a significant impact on the results.



One other point about the dedicated case is that this analysis implicitly assumes that flows are dedicated equally to tools. That is, we are effectively considering two equal volume recipes. In the dedicated case, we assign each recipe to a different tool. In the cross-qualified case either recipe can be run on either tool (hence the need for additional setup/qual time). In practice, it's unlikely that actual production volumes will be so equally balanced. Instead of two tools each at 85% utilization, you might end up with one tool at 75% and one at 95%. Cycle time will be significantly worse in the latter case, because of the nonlinear operating curve. This is another argument that cross-qualification is likely to be better than dedication for cycle time performance.

Conclusions

This newsletter and accompanying cycle time improvement course have stated many times that a top recommendation for improving fab cycle time is, where possible, to qualify a second tool. Where additional setups are not an issue, qualifying a second tool results in a roughly 50% reduction in cycle time per visit. Fabs that are cycle time focused (e.g. foundries) tend to have policies that require at least two qualified tools per recipe before a new flow is released to production.

This article presents a more nuanced view of the benefits of cross-qualification in the presence of setups. Even when there are setups, it's usually still worthwhile to qualify that second tool, especially when your tools have high availability or process time variability. There may be cases where the setup time is very long and/or the tools in question are already at a high utilization. In those cases, tools may not be able to absorb those extra setups without paying a significant cycle time penalty. Where this occurs, dedication may still be necessary.

In most cases, the benefits of qualifying a second tool remain strong. Having a backup tool provides an alternative path that keeps lots moving in the presence of variability (from down tools, long process times, etc.) and lowers average cycle time as well as cycle time variability.

Closing Questions for Subscribers

Does your fab require a second qualified tool for each recipe when releasing new flows to production? Have you found real-world scenarios where cross-qualification hurts cycle time, by depleting your standby time buffer too much?

Further Resources

All past FabTime newsletters are available in PDF format from the FabTime Newsletter Archive. Please reach out to me for the link or look in the most recent email issue of the newsletter. You can download individual issues or download a zip file containing all past issues. Some articles have been re-published on the INFICON website. Those are linked above where mentioned.

For a more in-depth discussion of how these choices apply to your site, consider hosting a session of our four-hour web-based cycle time management course.