

FabTime® Newsletter

Volume 26, No. 2

April 2025

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, April 8, 2025 – Vol. 26, No. 2

Keywords: Metrics Standardization; AskJen™; Variability; Line Linearity; Hot Lots

Table of Contents

- Welcome
- Community News/Announcements
- Subscriber Discussion Forum – Relative impact of variability for bottleneck and non-bottleneck tools; balancing turns goals by area; metrics for WIP linearity; operator constraints; and hot lots as a factor affecting cycle time
- **Main Topic – Is a standard definition for moves possible?**
- Current Subscribers

Welcome

Welcome to Vol. 26, No. 2 of the FabTime Cycle Time Newsletter. For our main topic, we explore the variation in fab move definitions and propose a supplemental moves metric called value-added completes that can be used to compare moves across fabs. We also have a plethora of interesting subscriber discussion on various topics as well as announcements about multi-language support in our AskJen AI chat engine and a recap of the latest FOA meeting.

Thanks for reading! – Jennifer

Community News/Announcements

In this month's community announcements, we have a photo-studded recap of the FOA Collaborative Forum and an update about multi-language support in our AskJen fab productivity chat engine. To suggest announcements, please [reach out to Jennifer via this form](#).

Highlights of the February FOA Collaborative Forum

The February SEMI [Fab Owners Alliance](#) Collaborative Forum was a great time! Below: The INFICON team Autumn Watt, Jennifer Robinson, Emilio Martinez, Birender Kahlon, and John Behnke at the Thursday social (missing Joseph Carloni). Also pictured: the winning trivia team the WoMentors from the Women of FOA pre-meeting event, as well as the full group. Many thanks to the SEMI team for an excellent event, and to Wolfspeed for supporting Joe and Autumn in our INFICON case study presentation.



The next FOA meeting will be held April 30-May 1 at Honeywell's fab in Plymouth, Minnesota.

Other Upcoming Events

In other conference news, INFICON's Holland Smith and Gero Grau will be attending the APCm Conference in Prague this week. Holland will be presenting April 10 on "Maximum Happiness or Minimum Disappointment: The Science (and Politics) of Applying Multi-objective Optimization (MOOP) in Semiconductor Factory Scheduling." INFICON will also be sponsoring a technical session

on Smart Manufacturing at the 2025 Advanced Semiconductor Manufacturing Conference (ASMC) in Albany, NY on Wednesday, May 7.

Multi-language support is now available for our AskJen chat engine

In previous newsletters, we introduced AskJen™, our AI-powered chat engine. AskJen harnesses cutting-edge Large Language Model (LLM) technology combined with over 25 years of FabTime cycle time management newsletter issues to empower semiconductor professionals with instant answers to operational questions. Those of you who attended the February FOA may have had a chance to see a demo, or at least to take home one of our beautiful AskJen coasters.

Back at home, we've been continuing to test and improve AskJen, making it ever more responsive. Most recently, we've developed **multi-language support to enhance the usability of AskJen for all of our customers.**

Now, if you ask it a question in French, Japanese, German, or other language of choice, AskJen will provide a response in the same language. In effect, this enhancement makes our 25+ years of newsletter content available to a global audience. For example, here we ask in Chinese about the fundamental drivers of wafer fab cycle time.

晶圆厂周期时间的基本驱动因素是什么？

晶圆厂周期时间的基本驱动因素是工具利用率、变异性和每个工具组的合格工具数量。这三个因素在工具组层面上是影响周期时间的主要因素。其他次要因素如停机时间和产品组合通过影响这三个主要因素来影响周期时间。

参考: FabTimeNewsletter, Volume: 22, Number: 4.

Continue

Alternatively, you can enter your preferred response language as part of your question. On the next page is an example with a more complex question. We've requested a response that includes recommendations from multiple newsletter issues and asked for the results in English and German.

How should I load batch tools in a wafer fab to minimize their impact on cycle time? Please consult multiple sources. Please answer in English and also share the answer in German.

Submit

To minimize the impact of batch tools on cycle time in a wafer fab, it is generally recommended to adopt a "greedy" batching policy rather than waiting for full batches. This approach is particularly effective for tools that are not heavily loaded. Here are some key points from multiple sources:

1. **Greedy Policy:** A greedy policy involves running smaller, more frequent batches rather than waiting for a full batch. This approach helps reduce cycle time, especially at low to moderate utilizations, and usually does not incur a cycle time penalty at high utilizations (Volume 9, Edition 3).
2. **Avoid Full Batch Requirement:** Forcing full batches on tools with lower utilization can significantly increase cycle times. The first lots to arrive may have to wait a long time for a full batch to form, which increases their cycle time (Volume 7, Edition 8).
3. **Smaller Batches:** Running smaller loads through batch tools that are not heavily loaded can have a significant cycle time reduction impact. This approach aligns with lean manufacturing principles and helps reduce variability (Volume 7, Edition 5).
4. **Consider Process and Cost Factors:** While a greedy policy is generally better for cycle time, there may be process or cost reasons to run full batches. It's important to balance these considerations with cycle time objectives (Volume 2, Edition 1).

In summary, adopting a greedy batching policy and avoiding the requirement for full batches can help minimize the impact of batch tools on cycle time in a wafer fab.

German Translation:

Um den Einfluss von Batch-Tools auf die Zykluszeit in einer Wafer-Fabrik zu minimieren, wird allgemein empfohlen, eine "gierige" Batch-Politik zu verfolgen, anstatt auf volle Batches zu warten. Dieser Ansatz ist besonders effektiv für Werkzeuge, die nicht stark ausgelastet sind. Hier sind einige wichtige Punkte aus mehreren Quellen:

1. **Gierige Politik:** Eine gierige Politik beinhaltet das Fahren kleinerer, häufigerer Batches, anstatt auf einen vollen Batch zu warten. Dieser Ansatz hilft, die Zykluszeit zu reduzieren, insbesondere bei niedriger bis mittlerer Auslastung, und führt in der Regel nicht zu einer Zykluszeitstrafe bei hoher Auslastung (Volume 9, Edition 3).
2. **Vermeidung der Anforderung voller Batches:** Das Erzwingen voller Batches bei Werkzeugen mit niedrigerer Auslastung kann die Zykluszeiten erheblich erhöhen. Die ersten Lose, die ankommen, müssen möglicherweise lange warten, bis sich ein voller Batch bildet, was ihre Zykluszeit erhöht (Volume 7, Edition 8).
3. **Kleinere Batches:** Das Fahren kleinerer Lasten durch Batch-Tools, die nicht stark ausgelastet sind, kann einen signifikanten Einfluss auf die Reduzierung der Zykluszeit haben. Dieser Ansatz stimmt mit den Prinzipien der schlanken Produktion überein und hilft, die Variabilität zu reduzieren (Volume 7, Edition 5).
4. **Berücksichtigung von Prozess- und Kostenfaktoren:** Während eine gierige Politik im Allgemeinen besser für die Zykluszeit ist, kann es Prozess- oder Kostenfaktoren geben, die volle Batches erfordern. Es ist wichtig, diese Überlegungen mit den Zykluszeitzielen in Einklang zu bringen (Volume 2, Edition 1).

Zusammenfassend lässt sich sagen, dass die Einführung einer gierigen Batch-Politik und die Vermeidung der Anforderung voller Batches dazu beitragen können, den Einfluss von Batch-Tools auf die Zykluszeit in einer Wafer-Fabrik zu minimieren.

For more detailed information, you can refer to FabTime Newsletter Volume 2, Edition 1; Volume 7, Edition 8; Volume 9, Edition 3; and Volume 7, Edition 5.

The multi-language support in AskJen opens up many excellent use cases for our customers

around the world. AskJen is currently available for testing by FabTime customers. [Reach out to Jennifer](#) for more information.

Interesting Reads

Semiconductor Insights and Productivity Tips

Recent articles shared on Jennifer's LinkedIn include:

- Industrial Engineering geeks like me will enjoy [this WSJ article](#) about how engineers from Chick-fil-A pore through videos, including drone footage, to identify and remove bottlenecks in the drive-thru order process.
- An interesting [article in The Wall Street Journal](#) about disk drive technology. The focus is mainly on Seagate Technology's MN recording head wafer fab, but Western Digital is also mentioned.
- A [news release about SkyWater Technology](#) purchasing Fab 25 in Austin from Infineon Technologies. This move increases US foundry capacity (for 65 to 130 nanometer devices) and "affirms the long-term perspective for nearly 1,000 manufacturing jobs at Fab 25." Historical side-note: Frank Chance and I were onsite at Fab 25 with John Behnke back on 9/11. This fab has seen a lot of changes since then, as have we.

For more industry news, [connect with Jennifer on LinkedIn](#).

Subscriber Discussion Forum

We have subscriber discussion about the relative impact of variability for bottleneck and non-bottleneck tools; balancing turns goals; metrics for WIP linearity; hot lot impact on cycle time; and operator constraints. If there is a topic you've been wondering about, [please let us know](#).

Relative impact of arrival and process time variability for bottlenecks vs. non-bottlenecks

A **new subscriber from Taiwan**, after studying resources from this newsletter and consulting with a colleague from National Taiwan University, observed that the relative impact of arrival variability vs. process time variability will depend on whether a tool group is a bottleneck. He observed that:

1. For bottleneck (high utilization) tools, a high squared coefficient of variation of effective process time will result in a high squared coefficient of variation of the departure process, regardless of whether the coefficient of variation of the time between arrivals to the tool is high or low. This is because the near-constant queue at a bottleneck tool will absorb the arrival variability. When lots arrive at the back of a long queue, the impact of any variation in the arrival pattern of those lots is dampened.
2. For non-bottleneck (low utilization) tools that don't typically have a queue waiting, a high squared coefficient of variation of the arrival process will result in a high squared coefficient of variation of the departure process, whether the coefficient of variation of the effective process time is high or low.

Response from Jennifer: I think these points are intuitively reasonable. I've said many times in my cycle time class that when lots are arriving at the back of a long queue, the impact of the arrival variability is lessened. In the low utilization case, without much of a queue to absorb the arrival

variability, it makes sense that this arrival variability will be passed downstream, potentially augmented by process time variability.

Consider the simpler versions of the queueing formulas for generating operating curves, when we're looking at x-factor at the current tool group. See [our webinar on the Fundamental Drivers of Wafer Fab Cycle Time](#) (video is at the bottom of the linked web page) for details. Here, it doesn't matter very much whether variability is from time between arrivals or from process times, because CV_a and CV_p are weighted equally in the formula (though effective process time variability will also be influenced by downtime distribution when we expand the formula a bit, [as discussed in Issue 25.04](#)).

However, a couple of points to keep in mind are:

1. The queueing formulas (as reflected in the operating curve spreadsheet) still show arrival variability having a significant impact on cycle time x-factor at extremely high utilizations. However, in a real fab this effect is dampened by the fact that we don't have infinite WIP. We (hopefully!) never see the highest x-factors that the queueing model predicts from high utilization and variability, because our WIP is capped.
2. Even though the average queue time might be low for non-bottleneck tools (because lots usually arrive when the tool is not busy), we should remember that variability doesn't only impact the current tool. Arrival variability to low utilization tools impacts the variability of the departure process, which becomes the arrival variability for downstream steps.

We appreciate this contribution to the discussion! There's always more to learn and think about in understanding fab behavior.

How should supervisors allocate operators to balance different turns rate goals?

A **long-time subscriber** wrote: "How do you deal with shift supervisors who are held to account for turns? They could be responsible for three areas (CMP, Epi, Implant) each having a massively different turns expectation. Because we are a lean operation, associates are trained to work in all three areas.

Human nature has the supervisors sending personnel to the high turns goal area. This can cause problems for the lower turns rate area (Epi), which cannot catch up if it falls behind. Can you use 'Area Health Charts' that normalize the turns per hour overlaying each area, showing you where to send personnel?"

Response from Jennifer (after consultation with Paul Campbell, our Smart Manufacturing Specialist):

I think that turns are most useful at the factory level, as an indicator of the pace of the line. For a given route, the turns rate, together with an approximate number of steps per flow, can be a forward indicator of cycle time. If your turns rate drops, this is an early warning that future cycle time will be increasing.

The idea behind using turns for areas is that turns account for shortages and excesses of WIP, so we don't penalize the person running that area for not meeting a moves goal when there isn't sufficient WIP. However, turns are not an ideal primary metric for areas because using area-level turns can hurt line balance and increase variability.

A high turns goal in an area means either that we're expecting to do a high number of moves in that area during the shift or that we expect to have low WIP in the area. In the latter case, this encourages areas with low WIP to make moves that may not be ideal from the fab's perspective. Conversely, a low turns goal means that we either expect not to do too many moves or we are holding a lot of WIP in the area. In the latter case, use of turns discourages making moves (and assigning associates), even though keeping that WIP moving would be helpful for cycle time improvement. This sounds like exactly what you are describing in the Epi (low turns) area vs. the others.

I think the fact that turns rate depends on both moves and WIP makes it imperfect for deciding where to send personnel. It's also been my experience that the hourly turns rate in an area can be quite variable. This seems like a question that a scheduler (like [INFICON's Factory Scheduler](#)) should be answering, by setting targets for what specific WIP needs to be moved during the shift. If keeping Epi from falling behind is a priority or constraint, then the scheduler should prioritize having someone do those Epi moves. This avoids the "human nature" part of the decision.

Another way to think about this is to look at which tool groups are the constraint tools for the fab. Those are the tools that we want to ensure don't sit idle with WIP waiting because there's no associate to run the tool. Where supervisors see that happening, that's where they should be sending personnel.

One other idea for improving area performance is to use a metric that we developed with a customer in 2013 called Earned Plan Hours (EPH). EPH is an alternative to moves for tracking production activity. EPH gives credit both when lots are tracked into tools, and as processing occurs on tools. Thus, EPH provides an incentive to keep tools running, even near shift change. See Issue 14.01 for details.

My response here is not to share a better way of using turns to avoid this problem, but to propose that other metrics may be better used for tracking area performance. Sorry if that's not helpful.

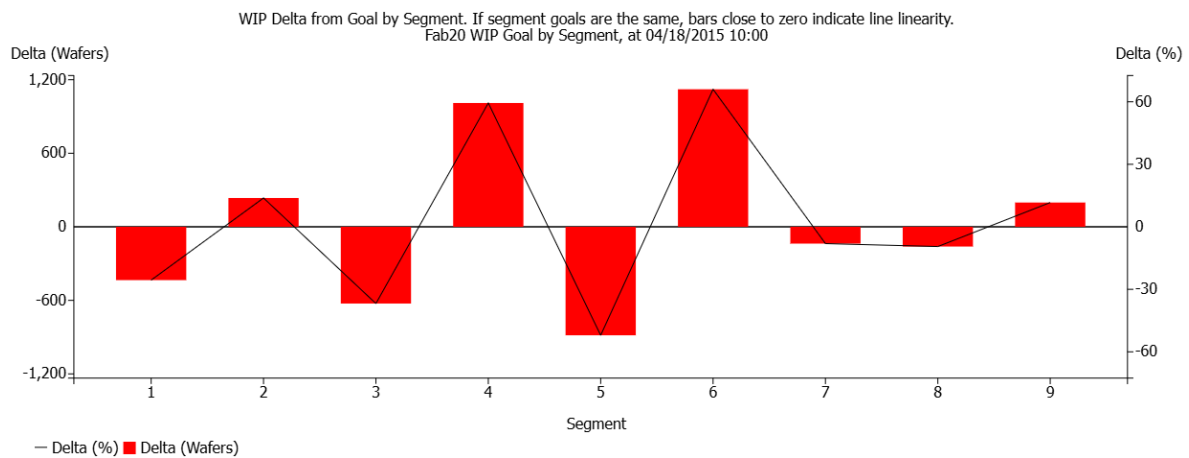
I would love to have input from other subscribers on whether they use turns in this way or have any suggestions. Does anyone have anything to add here?

Metrics for WIP linearity

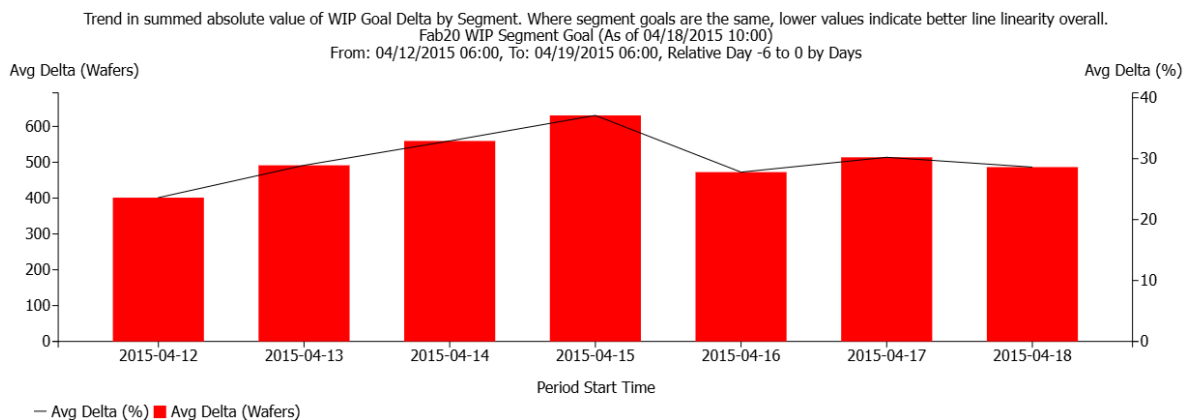
A **new subscriber** asked: "Is there an established metric to quantify and visualize WIP balance in a manufacturing line that can be used to show a trend? We are looking for something that highlights missing moves in parts of the flow as well as 'too many' moves in other parts of the flow. Maybe you can point me to corresponding papers."

Response from Jennifer: One metric that we have in the FabTime reporting software that could capture this is WIP Goal Delta. Data for this chart is initially calculated at the object level for whatever the customer chooses to slice by and then can be aggregated over time to look at trends. To look at line linearity, what you could do if your fab is in a relatively steady state in terms of starts is look by segment or sub-segment of the line (usually a segment is about a week of process flow, while sub-segments are smaller) and set a WIP goal that is the same for each segment. Then you can look at the absolute delta between that goal and the actual WIP by segment, summed across the segments, and reported over time. You'd be looking for an absolute delta as close to zero as possible. Note that if the WIP levels in the fab change, and/or the start rate changes significantly, it might be necessary to adjust the targets.

Here's an example of a snapshot of this WIP Goal Delta by Segment:



We can see that the early and late segments do well meeting the WIP target, but the segments in the middle are more variable. And here is a rolled-up trend version:



The height of the right-most bar on the bottom chart is the sum of the absolute values displayed on the top chart. Where segment goals are the same, lower values on this chart indicate better linearity. I like this metric because it's flexible (you can define the segments or sub-segments as you like).

The INFICON [Scheduler](#) and [Factory Dashboard](#) also use a line balance metric for ensuring WIP Linearity. The calculations are similarly based on the absolute value of the largest effective delta to target WIP or the largest WIP target in the facility, whichever is greater.

Do any other subscribers have other metrics to suggest here for maintaining WIP linearity?

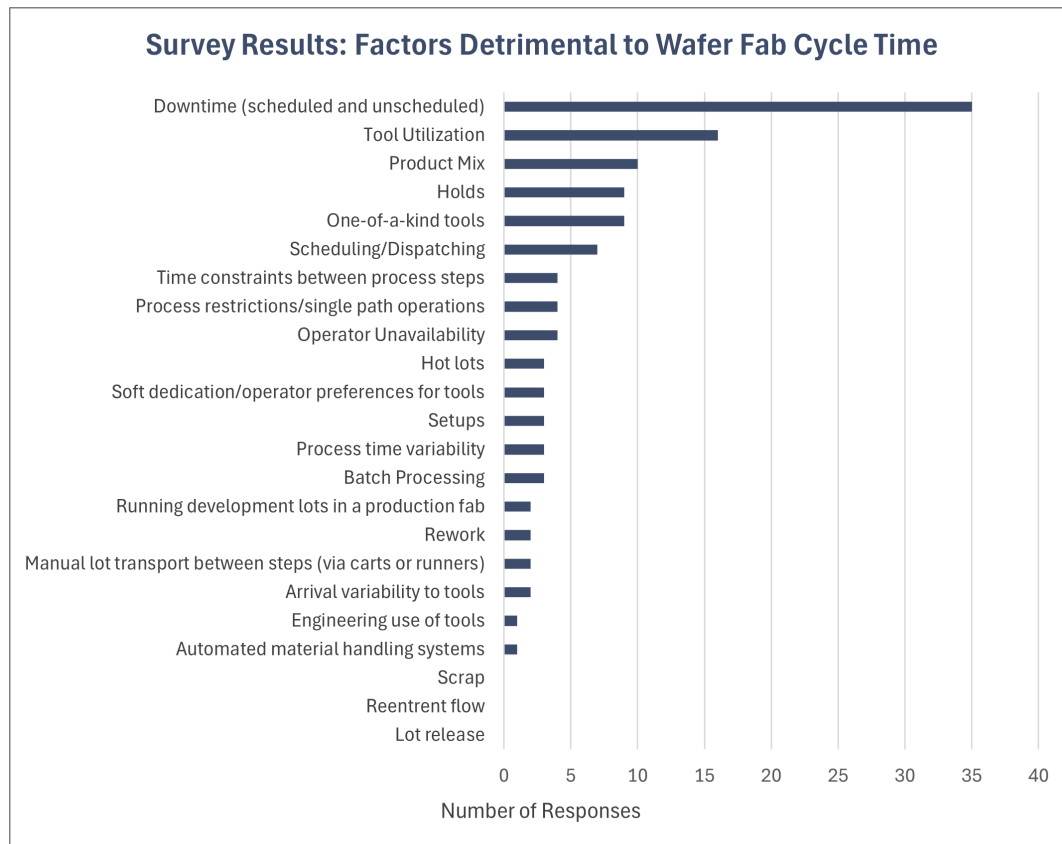
Why weren't hot lots on the list of factors influencing cycle time in the previous issue?

A **longtime subscriber** wrote in response to the February issue: "I am wondering that priority corridors (e.g. hot lots) did not at all appear in the list of detrimental factors to wafer fab cycle time, since it complicates on-time-delivery and increases variance of fab cycle time for the regular lots, I would think."

Response from Jennifer: Thank you for noticing that hot lots weren't on the list in the graphic. That was due to an editing error on my part. I checked the spreadsheet I used to compile the results, which I had reformatted since publishing the earlier issue on this topic and discovered that

I had combined hot lots with holds. They are separated properly in the figure below (nine responses for holds and three for hot lots).

This survey question, which was on the FabTime website for a time, received 123 total responses. As people could only select one factor, I'm not surprised that hot lots weren't selected frequently. I do think that downtime, tool utilization, and product mix have a greater impact for most fabs, though I agree that hot lots increase variability (and sometimes cause capacity losses), and thus do impact fab cycle time.



For more about the impact of hot lots on cycle time, see Issues 19.03 and [25.01](#).

How can I model operator constraints?

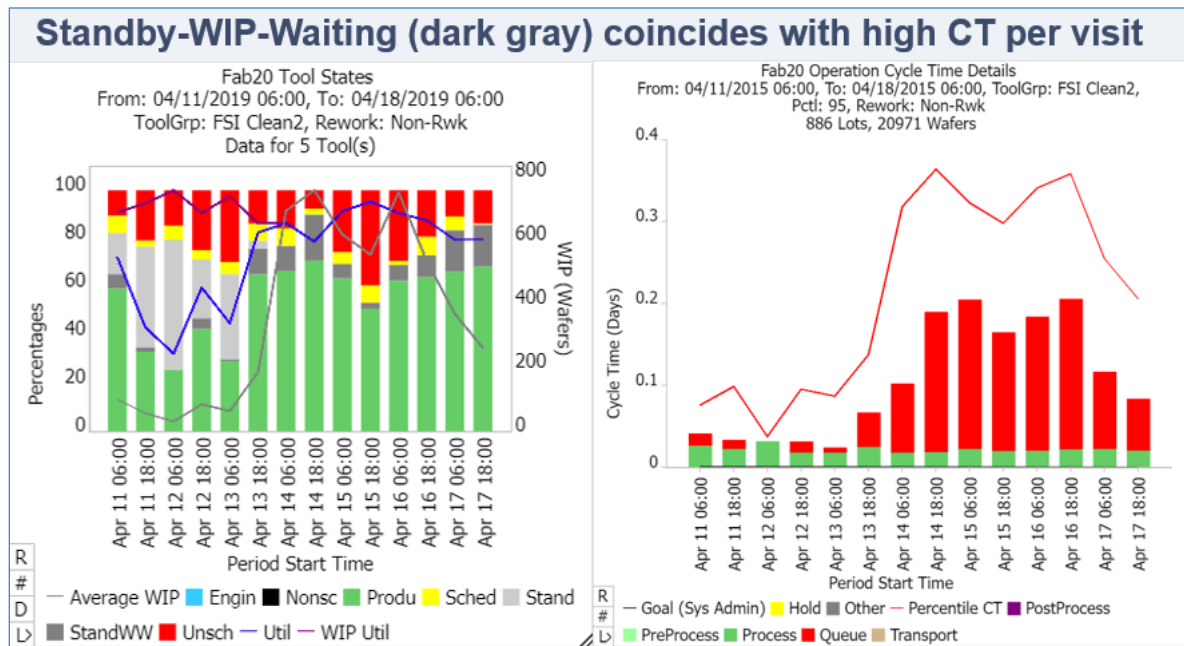
A new subscriber wrote: "One of the areas that I am interested in is not just equipment utilization, but people utilization. In most areas I am more people constrained than equipment constrained. I did find your newsletter on that subject but also would be interested in how to effectively model that constraint with the other three cycle time drivers."

Response from Jennifer: We don't have a lot of content on the impact of operators beyond what's included in our past newsletters on that topic, though we do also discuss this in [our cycle time class](#).

In general, you can think about how operators affect the three fundamental drivers of cycle time.

1. Operators effectively increase utilization because you end up having either:
 - a. Inflated process time because no operator is available to unload the tool (which you can sometimes measure as post-process time); or

- b. Wasted capacity due to tools sitting idle with WIP in front of them because there's no operator to load the tool (which you can measure as standby-WIP-waiting time, as shown below).
2. Operators increase variability. This can be hard to quantify, however.
3. Operators can reduce the number of available qualified tools. This happens when you can't run tools because you don't have enough operators, or when operators just don't run as many tools as you expect, resulting in soft dedication.



What I would focus on most would be #1. Try to measure where operators are driving up utilization on tools, particularly where this is happening on your bottleneck tools. And then try to reallocate operators where your cycle time is the highest. These are things that INFICON can help you pull out of the data with [FabTime](#) and [Factory Dashboard](#), but you might also find that your supervisors know where this is happening already. Do other subscribers have anything else to suggest here?

We welcome the opportunity to publish subscriber discussion questions and responses. [Submit your responses here.](#)

Main Article: Is a Standard Definition for Moves Possible?

By Jennifer Robinson

Although every fab seems to define moves differently, we believe that a consistent industry definition offers value. While fabs are unlikely to change existing definitions, we propose a clear, supplemental metric: value-added completes.

Over the 30+ years that I've been working with fabs, one thing that has become clear is that even the most seemingly simple topic is likely to become complex once you dig into it. A notable example of this is moves. Every fab tracks moves as a core manufacturing metric. The very first chart included in FabTime was a Moves Trend chart. We all know what moves are.

But do we?

When we talk about moves, are we talking about the same thing? And if we're not talking about the same thing, how can we make comparisons across fabs?

What is a move?

"Move" is presumably short for "move out," which is a transaction logged to an MES when a lot has finished processing on one tool and is ready to be transported to the next tool. This sounds straightforward. However, when we dig in, various questions arise.

- Does a move have to include a tool, or should we include things like visual inspections, where only an operator is needed?
- Does a move have to be a value-added step? (And what does value-added mean?)
- What about steps like inspections that are only done on some lots? You can skip them. It's more efficient to skip them. But there are reasons to do them. Should you count the ones you do as moves?
- What about rework? Are rework operations moves?
- What about fabs that track moves at a higher level, for a group of operations (like stage moves in Promis)? Are those moves?
- What transaction should we use to designate the move? Typically, there is a track out transaction that is automatically recorded by the tool in more automated fabs or manually logged as a move out by an operator in less automated fabs. Some fabs, just to keep things interesting, have a mix of more and less automated tools.
 - What if the lot is still stuck on the port of the previous tool? It completed the operation and may have been automatically logged as track out even though it's still sitting on the port of the previous tool because no one has physically moved it yet. Was *that* a move?

As we work to integrate the FabTime reporting module into the INFICON Smart Manufacturing suite, we've had occasion to validate the move numbers displayed for the same demonstration data in the FabTime system and the Factory Dashboard (formerly called FPS Dashboard). We also have a history of matching move numbers between FabTime and Factory Dashboard for joint customers that pre-dates INFICON's acquisition of FabTime. Matching these numbers isn't as straightforward as we might prefer, because of the above questions. Let's explore this further.

Why is it important to have a good definition for moves?

We need a good definition for moves because:

- People use them to compare across fabs. A move isn't necessarily the same thing in different places. This renders comparisons invalid.
- No matter how many other metrics are put in place, operators on the floor usually pay close attention to move targets. This is because moves give instant recognition of activity (vs. cycle time and outs, which are lagging indicators).

- How we define moves affects how we measure cycle times at the step level, which in turn drives how much we're able to improve. The more granular the move, the more information we have about queue time vs. process time at individual tools.

Our definition should comprehend whether a move is value-added because:

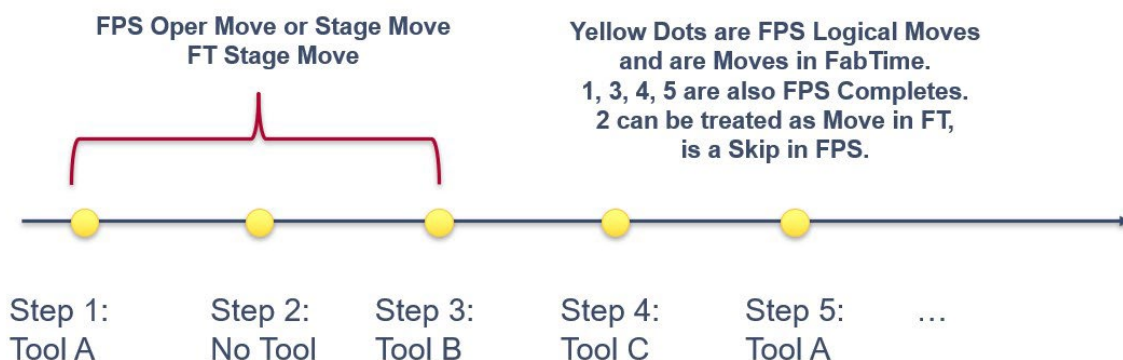
- If people in the fab are focused on total moves and we successfully reduce non-value-added activities, the total number of moves will decrease. This may make people uncomfortable, unless they have access to a move-related number that *increases* due to these efforts.
- If we're not using a good definition for moves, operators can end up incentivized to make poor choices in the fab. For example, if the non-value-added moves are the easier moves, and they count, why wouldn't operators focus on those? This is less of an issue for more automated fabs, of course, where the scheduling system sets the plan.

For these reasons, we as an industry need a clear and consistent definition structure for moves that can be used within and across fabs to facilitate tracking and improving fab performance. We've been working with a team of INFICON customers and engineers to provide that structure, and we are sharing our recommendations here.

How have Factory Dashboard and FabTime historically defined moves?

The Factory Dashboard (AKA FPS) team always had a structured definition for moves, as shown below, while FabTime has been a bit more flexible depending on the needs of the customer.

Move Definitions in Factory Dashboard (FPS) and FabTime (FT)



In Factory Dashboard:

- "Step completes" are recorded for each completed step that includes a tool visit.
- "Skips" are recorded for completed steps that do not have a tool associated with them (like visual inspections).
- "Logical Moves" are the sum of step completes and skips but are not prominently used as a metric in Factory Dashboard. The lot is moving logically from one step to the next.
- "Oper moves" or "Stage moves" are recorded when a small grouping of contiguous steps is completed. This group of steps can be called an operation (Workstream) or a stage (PROMIS) or whatever that site prefers.

In FabTime:

- Moves are the same as Factory Dashboard step completes (recorded at the level of a single step, not grouped), except that skips can also be treated as moves in FabTime if the customer chooses (this is site-configurable).
- Stage moves in FabTime are essentially the same as Factory Dashboard oper moves. The customer can have a flag e.g., FabTime.StageOut, which is set to “Y” when a FabTime move is also a stage move. This lets the customer report those higher-level stage moves as well as the more detailed step-level moves.
- Operation in FabTime is always a single step, though some customers only report stage moves. Customers can also include a flag denoting that a step is value-added. In this case, non-value-added moves can be excluded.

Each of these frameworks is reasonable. Each has been used in dozens of fabs over the years. But they are indisputably different.

What did our customers say about this?

Faced with definitional differences, we did what we always do. We asked our customers what they think. While we will of course maintain confidentiality of individual customer responses, here are some things we learned:

- One company uses “move” for a group of steps, defined as a value-added operation (series of sub-steps). They don’t count any metrology steps as moves, because they don’t want to incentivize over-inspection.
- A couple of companies use both completes (which must be done on a tool) and logical moves (which don’t require a tool), as defined in Factory Dashboard. But for one of them, completes must be value-added. The other just requires that completes be on a tool.
- Another company uses completes as defined in Factory Dashboard, but they call that moves. This metric can include non-value-added steps, but they also use a stage move that must be value-added.
- Another company just says that if a step is completed on a tool (including metrology tools), then it’s a move.
- One site counts everything except rework, even staging operations, as moves, but then designates a separate financial move that accounts for added value.
- Another mostly uses stage moves but sometimes uses the moves as defined in FabTime, with no differentiation for value-add.

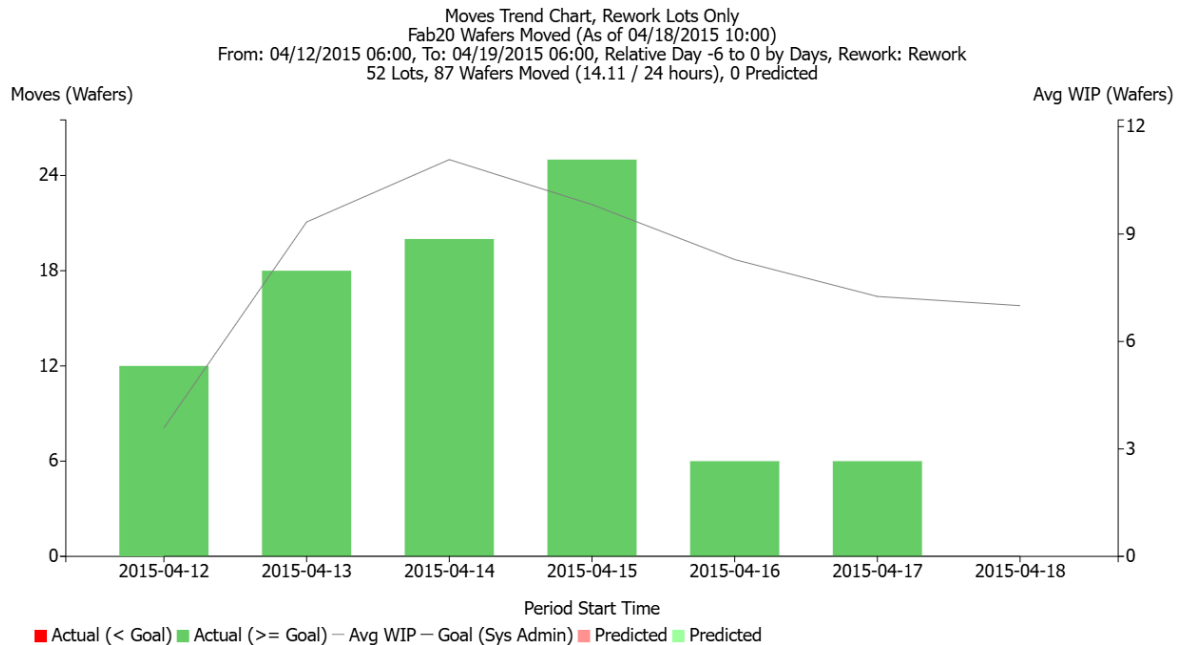
In summary: it’s complicated. Some companies use step moves, and some use stage moves. Some require a tool to be included and some don’t. Some look at whether steps are value-added, and some don’t. Some use multiple definitions of things related to moves for different purposes (moves vs. completes vs. stage moves vs. financial moves, etc.).

About all we can reliably conclude is that:

- A stage move is generally a group of steps; and
- A complete usually requires a tool.

What about rework?

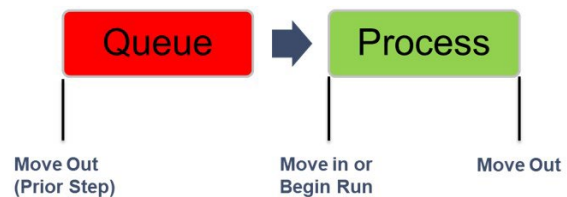
Rework adds another layer of complexity to move definitions. Some people treat rework steps as moves because the operators are doing the work. Some don't, because these steps aren't value-added. Sometimes it depends on whether the rework occurs during the same shift as the original move. In FabTime, all moves that take place within a rework loop have a rework flag. Any moves chart can be filtered to display all moves, non-rework moves only, or rework moves only. For example, the chart below, from our demo server, is filtered to only show rework moves.



How should we record moves if we care about improving cycle time?

Tracking moves as completes (one step completed on a tool) is best for analyzing operation cycle time. Stage moves can be useful for comparing fabs (if the stages are defined consistently), but they don't tell us which tools are accruing queue time. For that, we need to track the move in and move out at each tool. (Even then there are possible further levels of detail, which we will address another time.)

To understand cycle time, we need to track moves at each tool



Tracking logical moves that don't include processing at a tool could be helpful for highly manual fabs. For understanding cycle time, it's better to track these as additional steps rather than lumping the time in with, say, queue time for a tool. For instance, if there's a visual inspection prior to a step that takes place on a tool, and there's a wait for the technician who can do the visual inspection, it's more informative to capture that separately from the subsequent queue time for the tool itself. Maybe the capacity of the tool is fine, but production is gated by a lack of technicians to do the visual inspection. The more information we can gather about the causes of delay, the better positioned we'll be to make improvements.

On the other hand, when it comes to metrology steps that can be skipped and aren't done on every wafer, we agree with our customer cited above that including these as moves can lead to

poor incentives. If you can do a quick track in and track out of a metrology step, and get credit for an easy move, that can incentivize doing too many inspections. For cycle time we need the granularity of the individual step moves. However, including them for non-value-added steps can lead to artificial inflation of move numbers.

So how can we define moves in a way that lets us compare across fabs, but also gives different fabs the flexibility that they need?

After working with many fabs over the years and discussing moves definitions recently within INFICON and with INFICON customers, our metrics alignment team has concluded that very few fabs are going to be willing to redefine moves for their site. Changing the definition of something so fundamental to fab culture is a non-starter.

That said, we believe it's worth adding a supplemental move metric that is defined the same way for all fabs and thus allows for benchmarking across fabs. We propose:

Value-Added Complete: A lot moves logically from one step to the next AND is processed on a tool AND value is added. Since value is not added by doing rework (we would like to disincentivize doing extra rework), rework steps are not value-added completes.

For an existing FabTime or Factory Dashboard customer to display value-added completes, they would need to add a "value-added" yes or no flag to each step (if not already included). This could be done programmatically via a set of rules. The rules could be something like "everything that isn't a rework step or an inspection step that requires a tool is a value-added step." With such a flag in place, it would be possible to filter existing FabTime moves charts or Factory Dashboard completes to display value-added completes. Value-added completes could then be used to benchmark across fabs and across companies, and to measure improvement progress.

Conclusions

In our experience, the most widely used metric in most wafer fabs is moves. Moves tell us whether a fab is on pace to meet overall throughput goals, and whether individual areas, operators (sometimes), and shifts are ahead or behind. Despite the ubiquity of "moves" as a metric, however, we've observed that all moves are not created equal. Differences exist between fabs regarding whether moves must be done on a tool, must be value-added, can include rework steps, and can (or must) include groups of steps. These differences make it impossible to compare performance across fabs with any degree of accuracy.

Because fabs are accustomed to their own move definitions, we do not believe it feasible to ask for broad change. We do, however, believe that a new move-related metric, one that is clearly defined and consistent across companies, will be a useful addition to the fab metrics toolkit. We thus propose that in addition to measuring moves the way they always have, fabs should also start measuring value-added completes. Value-added completes are transactions in which a lot moves logically from one step to the next, is processed on a tool, and has value added.

Acknowledgements

Many thanks to the INFICON internal metrics team as well as the customer representatives who have been generous with their time and opinions as we've worked through questions about moves and other metrics. Special thanks to my co-leader on the metrics team, **Paul Campbell**, whose consulting experience informs this article.

Subscriber List

Total number of subscribers: 2,792

Top 20 subscribing companies:

- onsemi (122)
- Micron (113)
- Intel (110)
- Infineon (107)
- Analog Devices (97)
- Microchip (96)
- Skyworks (91)
- GlobalFoundries (78)
- NXP (67)
- STMicroelectronics (66)
- Texas Instruments (66)
- Seagate (57)
- X-FAB (56)
- Western Digital (55)
- Wolfspeed (52)
- Tower Semiconductor (37)
- Qualcomm (36)
- Applied Materials (35)
- ASML (33)
- SkyWater (30)

Top 4 subscribing universities:

- Arizona State University (7)
- Ben Gurion University of the Negev (5)
- Ecole des Mines de St. Etienne (EMSE) (3)
- Nanyang Technological University (3)

Note: Inclusion in the subscriber profile for this newsletter indicates an interest, on the part of individual subscribers, in cycle time management. It does not imply any endorsement of INFICON or its products by any individual or his or her company.

There is no charge to subscribe to the newsletter. Past issues of the newsletter are now available in PDF for download by newsletter subscribers [from the INFICON website](#). To subscribe to the newsletter, please [visit the FabTime registration form](#). INFICON will not, under any circumstances, give your e-mail address or other contact information to anyone outside of INFICON without your permission.