

IREC 2024 Post-Competition DFMS Analysis

Carl Anderson

October 25th, 2024

Purpose

The purpose of this document is to outline an initial effort into the processing and analysis of the data pulled from the 2024 IREC Digital Flight Management System (DFMS), as well as some initial conclusions & possible prescriptions.

Executive Summary

Broadly, the way that teams move through the steps required to launch at IREC (and by extension through the DFMS) can be thought of as a series of events (entering FSR, completing FSR, launching, etc), and queues where teams wait before these events (FSR waitlist, in FSR, pad queue, etc). The DFMS is our view into how teams moved through these steps: where teams spent more or less time, where the queues got congested, and what we can change to more efficiently get teams off the launch rail. Because this analysis is solely based on the data pulled from the DFMS, its quality is limited by the quality of the DFMS data. Among other known inconsistencies, the DFMS only counted 102 rockets launched, as opposed to the 118 counted by the LCO desk. More can be read on the steps behind the analysis in the Methodology section. Despite known data quality issues, this is the only way of this deep of a view into the functions of IREC, and I believe useful conclusions can still be drawn.

First, the stages can be broken roughly into conference day stages (registration & FSR) and flight day stages (FRR, pad queue, & flights). Beginning with registration, we can tell from fig. 1 that FSR was not being bottlenecked by registration, which impressively managed an average 2.4 minutes between teams completing registration. Additionally, registration shouldn't bottleneck FSR in steady state because the mean period between teams was lower for registration, as shown in fig. 2. FSR completions were on average faster than FRR completions according to fig. 2, which again means that in steady-state FSR should never bottleneck FRR. Both registration and FSR also have the advantage of 2 days lead time on all of the following stages, which means that even if these bottlenecked FRR on paper, they probably wouldn't in reality as registration was completed on Monday and 77% of teams had completed FSR by EOD Tuesday.

Launch day operations were where the problems begin to present themselves. Specifically, FRR was bottlenecking pad queue and therefore flights for much of Wednesday and even the beginning of Thursday. This can be seen in fig. 1, as the pad queue and flight burndown lines basically hug the FRR line during this period. This is quantitatively confirmed in fig. 2 as on average a team completed FRR every 18.5 mins, whereas teams were able to be launched every 13.2 mins. This means that FRR needs a 40% speedup in order to keep up with launching cadence. This inefficiency in FRR can be confirmed by looking at fig. 3, which shows that on average teams spent 10.1 minutes in FRR, despite 2 showing that on average a team completed FRR every 18.5 mins. Interestingly, looking at the medians instead of means for FRR durations vs FRR completions is closer to 5.26 vs 6.27 mins, which is only a 1 minute waste per team. I believe that certain factors may have lead to fewer, longer breaks between teams entering FRR, the cause of which I cannot determine from this data but could be theorized to be FSR waitlist being empty, teams taking a while to present at FRR after being paged, lunch breaks, or ESRA inefficiencies. I have the following three prescriptions for how to improve launch cadence for future years:

- **Increase the number of reviewers performing FRR reviews by at least 2x.** I am not sure the specific number of FRR reviewers during the '24 IREC but having a minimum of double and possibly $2x+1$ would ensure that FRR is keeping rate with salvo cadence.
- **Open FRR a minimum of 3 hours before first salvo is planned.** At last years launch cadence rate, assuming that FRR is staffed to match or exceed launch cadence, 3 hours is required to get a full salvo of teams (20) through FRR and into pad queue. This can probably be reduced if FRR is significantly revamped and efficiency is drastically improved.
- **Aiming to have a full pad queue at waiver open every launch day.** This would allow for the entire waiver to be used for launching, and in '24 this would've added 4 hours of launching between Wednesday, Thursday, and Friday mornings. Additionally, this would allow for more runway for the various weather delays and range shutdowns that inevitably occur.

With all three of these changes, assuming waiver open from 7AM—5PM for four days (a relative average of the past three years waiver schedules, a total of 40 hours), and assuming a (conservative) 30% weather hold range downtime, **approximately 125 teams would be able to launch in that 4 day launch window.** Alternatively, with a (very optimistic) 10% weather hold, nearly the same number of teams (123) would be able to launch in just 3 days of launching. This analysis this is based on averages (and past performance is no guarantee of future results), but this means that with good weather luck and smooth operations, a 3-day launch is possible, though its nearly impossible to count on as long as the number of teams launching remains at past levels.

Analysis

Figure 1 is a burndown chart in which each line represents the number of teams left to complete the major steps in the launch operations process: Registration, FSR, FRR, Pad Queue, and Flight. This chart gives a broad understanding of the flow of teams through the launch operations process and can help identify bottlenecking in the process. You can recognize when one stage is bottlenecking the next stage when their two burndown lines are highly correlated, or even appear to be on top of eachother. This means that the next stage (the bottlenecked) is only able to progress when a team makes it through the previous stage (the bottleneck). It can be seen that pad queue (and therefore also flights) was entirely bottlenecked by FRR on Wednesday and the beginning of Thursday. Registration and FSR didn't seem to bottleneck any further stages.

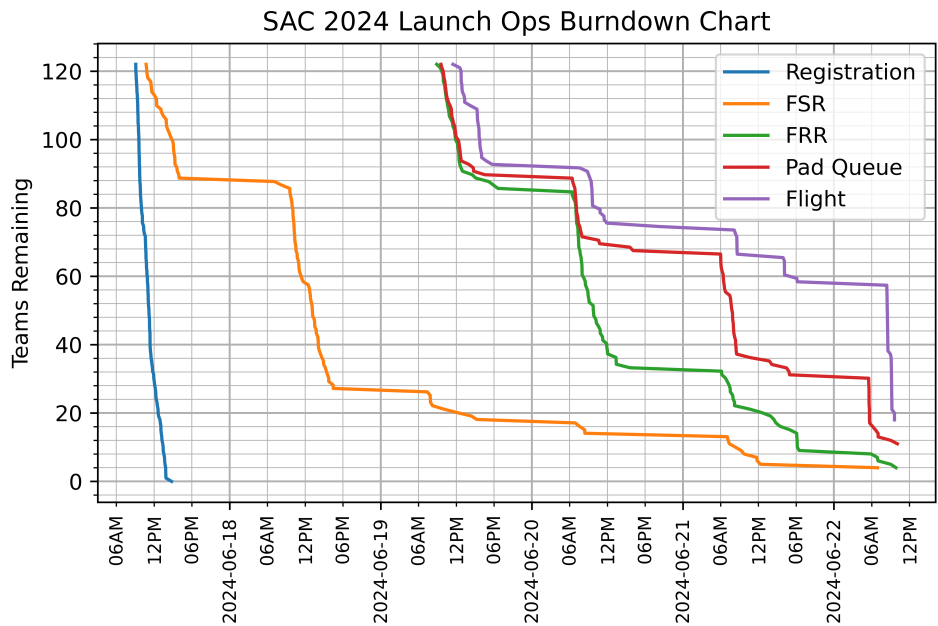


Figure 1: Major Launch Operation Stages Burndown

Figure 2 displays the average number of minutes between consecutive teams completing each of the major steps or stages for the IREC competition. In blue are the stages that typically are completed during conference days (registration and FSR) and in orange are the stages that can only be completed at the launch site (FRR, pad queue, launching). Looking at this data from a queueing theory perspective, imagining that each of the stages are “queues” that student teams have to wait in before moving on to the next one, if all consecutive stages were slower than the previous (FSR slower than registration, FRR slower than FSR, flights slower than FSR) then we would have an “efficient” system as defined to mean that once the first student team made it to the final stage, that final stage would always have teams waiting in queue, until all teams have completed the final stage. However, almost all of the orange or launch day queues are slower (longer period) than launches, which means that there necessarily was inefficient time wasted when more rockets could have been launching but didn't because not enough teams were through FRR into pad queue to send a full salvo out to the solids pads. It should be noted that significant outliers were trimmed ($> \mu + 3\sigma$) to get rid of inter-day waits between flights but retain inter-salvo waits.

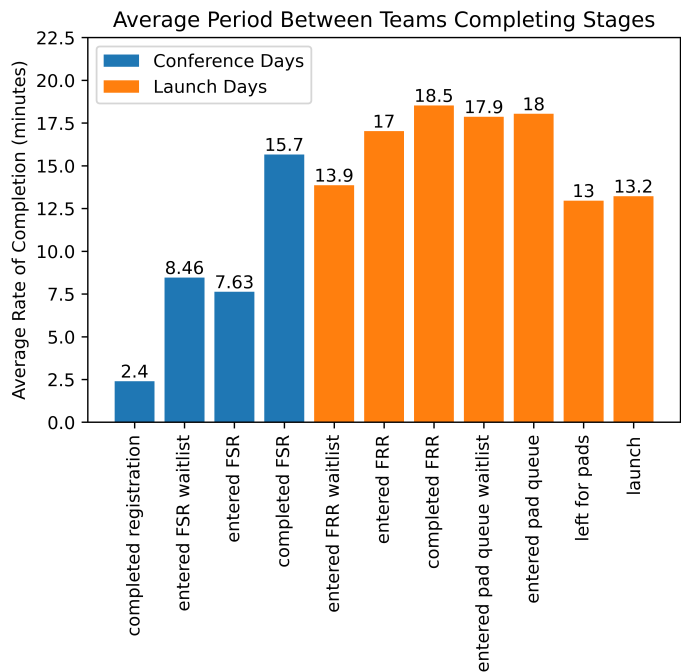


Figure 2: Average Time Between Teams Completing Stages

Figure 3 gives a view into how long it took students to get through the FSR/FRR queues, the actual FSR/FRR reviews, and how long remediations took to resolve. The height of each bar represents the number of teams that took the corresponding range of minutes to get through the respective stage. Outliers were trimmed only for the review durations (for both FSR & FRR) and was capped at 2 hours, as reviews lasting longer than 2 hours were assumed to be data entry errors.

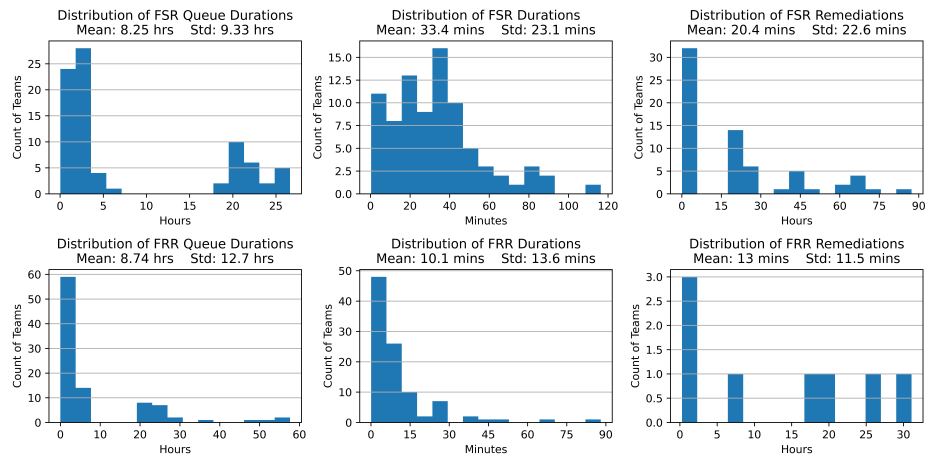


Figure 3: FSR & FRR Statistics

Figure 4 is similar to the previous and shows the distributions that teams waited after FRR to get into pad queue (Pad Queue Waitlist) and then the distribution of how long teams waited in pad queue to get out to the pads. The bottom two histograms are simply zoomed in on the large primary hump of the first one. The top plots have many outliers which extend into even the 2+ days range for some, I am not sure if this is due to poor data entry, scrubbed launches, or other issues at the pad. The zoomed plots were generated by trimming the top two plots data to < 1,4 hours respectively.

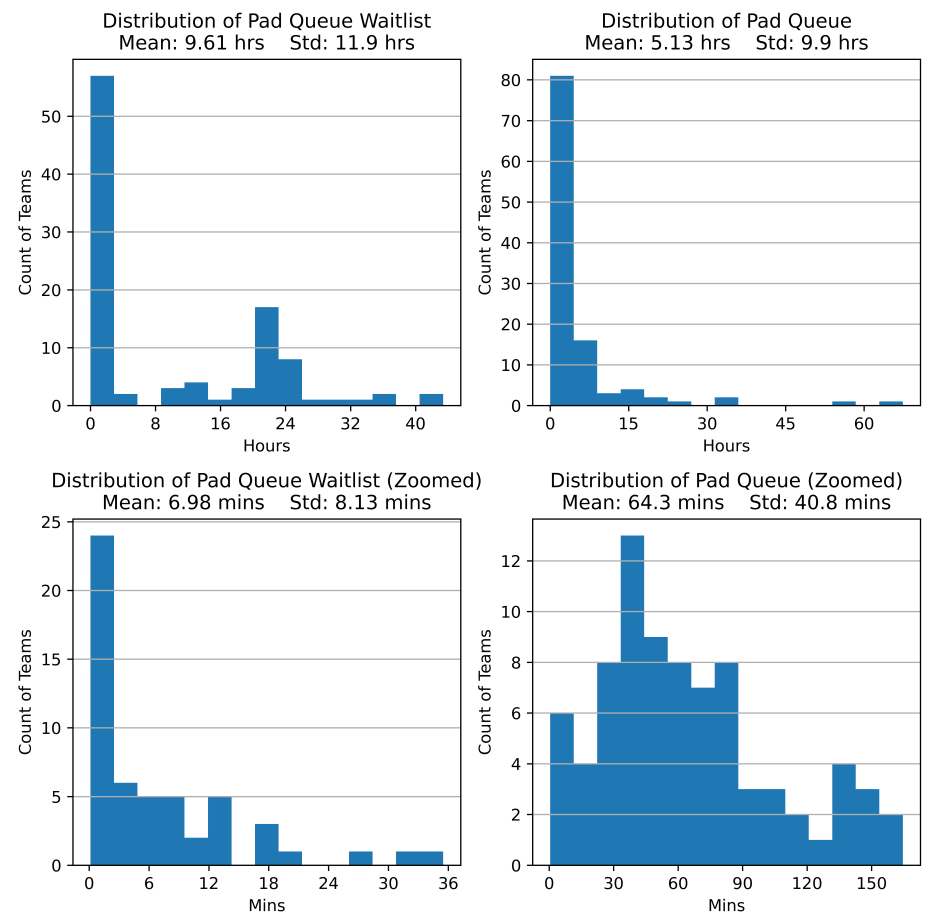


Figure 4: Pad Queue Statistics

Figure 5 is one of the more information rich being presented. It is similar to the burndown chart (Figure 1) except that it shows the actual state of each team over time instead of just counting down the number of teams left to reach some event. It also shows all of the states listed in the table of states above, whereas the burndown chart only had major steps in the process. Interesting anomalies are present, such as the 8 teams that were supposedly in FSR over Monday night, however overall this gives a very rich qualitative look at the state of the teams throughout the entire event.

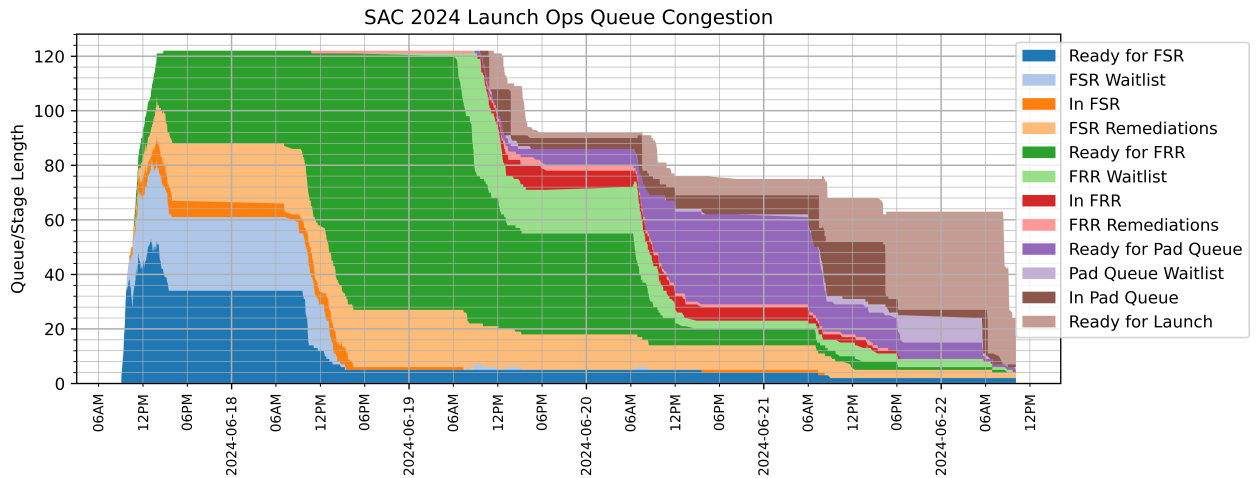


Figure 5: Queue Congestion

Finally, figure 6 is a flow diagram of the teams moving through key gates of the competition, which has the added benefit of showing which teams made it to FSR, and the overlap between FSR and FRR remediations. It should be noted that from the LCOs desk 118 launches were recorded. This clearly shows an issue in either the DFMS user quality or the quality of my extraction, there are probably issues with both. It should also be noted that this was manually generated to better account for inconsistencies with the data generated from the DFMS.

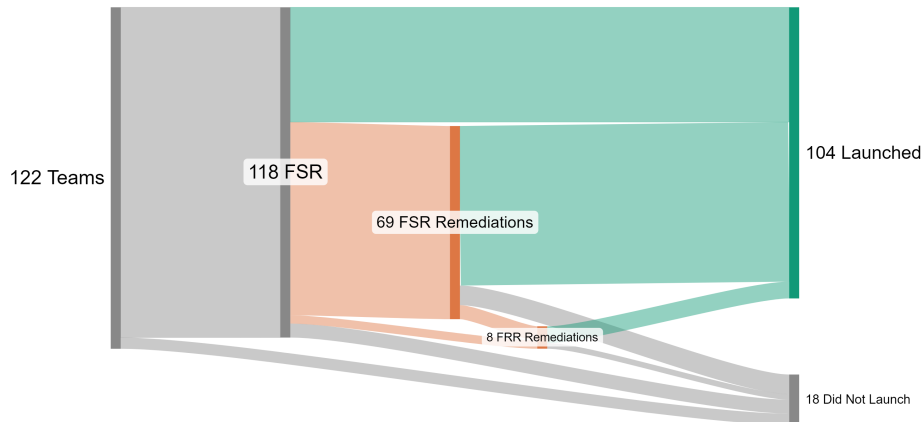


Figure 6: Team Flow Through Remediations

Methodology

The DFMS uses a web application called Trello for tracking the progress of student teams through the competition. Trello allows you to export a lot of data from various boards including but not limited to the board current state, data about the board configuration and lists, and a full action history. The action history includes every action that’s ever occurred on the board including but not limited to cards entering the board, exiting the board, moving between lists within the board, and card data being modified.

This analysis was done using the action history pulled from the DFMS boards. Actions tracked included team cards moving between lists within a single board as well as between boards. This allows for a macroscopic view of the timeline of each individual team moving through the competition through the pinhole of the DFMS.

A major caveat with this analysis is that the quality of the data presented can only be as good as the users of the DFMS. For instance, if a user accidentally moved a team to the wrong section, or forgot to move a team to the next section for a couple hours, these anomalies will be seen in the data analyzed here. This is unavoidable and for this reason the analysis presented here is best interpreted as broad trends (where teams were held up the most) and not microscoping issues in the process (why did team X spend 3 hours in FSR).

The key measurement that was taken on a per-team basis from the data was when various state changes happened: when a team left the registration board, when a team moved into the Undergoing Safety Review list, etc. These events as well as a qualitative description of them are listed in table 1. The rightmost two columns are only applicable for the Moved Lists action type.

Table 1: DFMS Events			
Description	DFMS Board	Action Type	List Name (if applicable)
Completed Registration	Registration	Left Board	N/A
Entered FSR Waitlist	Safety Review	Entered List	Safety Review Wait List
Entered FSR	Safety Review	Entered List	Undergoing Safety Review
Entered FSR Remediation	Safety Review	Entered List	Remediating Safety Review Issues
Completed FSR	Safety Review	Left Board	N/A
Entered FRR Waitlist	Final Safety Inspection	Entered List	Final Safety Wait List
Entered FRR	Final Safety Inspection	Entered List	Undergoing Final Safety Inspection
Entered FRR Remediation	Final Safety Inspection	Entered List	Remediating Final Safety Inspection Issues
Completed FRR	Final Safety Inspection	Left Board	N/A
Entered Pad Queue Waitlist	Pad Queues	Entered List	Pads Wait List
Entered Pad Queue	Pad Queues	Left List	Pads Wait List
Left For Pads	Pad Queues	Left Board	N/A
Launch	LCO	Left Board	N/A

From these events, states can be created between them. Generally, the order in which the states occur are supposed to be linear and monotonic, meaning that teams are only moving down the table through the states, never revisit a previous state, and will visit every state in the table. The exceptions to these rules are the FSR and FRR Remediation states which may be skipped by teams who are well prepared for the competition rigor. This is to say that the *In FSR* state ends either when the team reaches the *Completed FSR* event OR the *Entered FSR Remediations* event, in which case they enter the *FSR Remediations* state. Teams can only be in a single state at once. table 2 shows the states that have been collected, as well as the events that deem the beginning and end of each state.

Table 2: DFMS States		
State Name	Initiating Event(s)	Concluding Event(s)
Ready for FSR	Completed Registration	Entered FSR Waitlist
FSR Waitlist	Entered FSR Waitlist	Entered FSR
In FSR	Entered FSR	Entered FSR Remediations, Completed FSR
FSR Remediations	Entered FSR Remediations	Completed FSR
Ready for FRR	Completed FSR	Entered FRR Waitlist
FRR Waitlist	Entered FRR Waitlist	Entered FRR
In FRR	Entered FRR	Entered FRR Remediations, Completed FRR
FRR Remediations	Entered FRR Remediations	Completed FRR
Ready for Pad Queue	Completed FRR	Entered Pad Queue Waitlist
Pad Queue Waitlist	Entered Pad Queue Waitlist	Entered Pad Queue
In Pad Queue	Entered Pad Queue	Left for Pads
Ready for Launch	Left for Pads	Launch