While there is no doubt the quality of all speakers at the two day conference were exceptional, four of them in particular stood out to me and I wanted to highlight some of their insight here both for my own benefit, and to help consolidate for anyone who may be interested in the "cliffs notes" as they say.

1. Anjuan Simmons, Leadership Lessons from the Agile Manifesto
2. Sam Guckenheimer, Moving 75,000 Engineers To DevOps On The Public Cloud
3. Sarah Hagan, You're So Empathetic, You Probably Know This Talk's Not About You
4. John Allspaw, Taking Human Performance Seriously In Software

Leadership Lessons from the Agile Manifesto

Speaker: Anjuan Simmons

There are two key anecdotes from this talk that summarize it beautifully.

The first is simply that "dignity always outranks the chain of the command" by which Anjuan means that there is no greater or more important thing that a feeling of safety and dignity on a delivery team, or any team for that matter. Everyone should feel safe and empowered to share anything that makes them feel unsafe or otherwise uncomfortable, and enabling that culture on a team is paramount to its long term success.

The second is that "as tech leaders, you're building more than software" and this is fundamental to the leadership experience and a major part of my personal philosophy. As leaders, we need to take the responsibility very seriously that what we are doing is much more than simply delivering a software application or website. We are also building a team, and stronger teams deliver better, faster, and more. It's that simple.

Moving 75,000 Engineers To DevOps On The Public Cloud

Speaker: Sam Guckenheimer

As a technologist and someone whose entire career revolves around delivering software, this talk spoke to me deeply; I encourage *anyone* to review the entire talk. The simplest and most profound element of this talk that I took away was Sam's clarity on the reality of teams driving towards perfection, and *not* towards their SLA. Sam says (and I agree) that teams should never aim for their SLA, that should simply be the point by which you freely refund their money. Teams should drive for perfection, and the tools and processes they use should enable them to get closer to that, rather than inhibit or deter it.

You're So Empathetic, You Probably Know This Talk's Not About You

Speaker: [Sara Hagan](#)

This talk surprised me the most. I had clearly never put *enough* thought into the psychology behind teams and leadership. Do yourself a favor, and check out her [notes and details on the subject](#), and you'll thank yourself later. The emphasis on feedback and how to receive and give feedback with more rigor is something I intend to implement immediately with my teams, because I can see the benefits of this paying off just as quickly. The idea is simple, make sure you ask for and give feedback regularly. It is afterall the most effective way to communicate your desire to improve, and/or to coach. In doing so however, it's fundamentally important to not deliver a feedback sandwich; positive, negative, positive. It is disingenuous and fails to emphasize the importance of the need to correct something with respect to the "negative" feedback. Lastly, as a leader, you benefit most from being sincere in your request for feedback, and actually listening and reflecting when receiving it. Simply asking and then not following up is meaningless. In a leadership position it is very challenging to get your teams to give honest feedback, so achieving that level of trust is something you should take very seriously and be willing to receive it honestly and with an open mind.

However, you could argue the emotional appraisal to be virtually just as important, and as someone who often suffers from a difficulty of separating work and life stresses, I can see how this would also translate to being more effective day-to-day, particularly with larger teams. The simple idea is that if you can get a simple gauge for someone's mood prior to simple ceremonies like your daily standup, you can as a leader identify who would respond

better or worse to the daily "fire" at hand, or the need to do a less than desirable maintenance task, versus someone who is on "Cloud 9" and would be more receptive to spending the day on that legacy system for your lone surviving client who uses it, rather than tearing into some new code with the team for the day.

Taking Human Performance Seriously

Speaker: [John Allspaw](#)

This talk focuses on the human element of software complexity, the reality that RCA (root-cause-analysis) is wasted effort, and understanding how people cope with and resolve complexity within complex systems.

My favorite quip from this entire talk was in response to how RCAs could possible be a waste of effort; "the root cause of a plane crash has never been gravity" John said. After having completed far more RCA than I'd care to discuss, I can whole heartedly agree with this sentiment. John posed a question to the audience at one point asking everyone to raise their hands if their system/app would work as intended without consequence if the entire company decided to do nothing for one day; two days, a week, a month. Each time the number of hands decreased, until the one month at which point John jokingly asked the few remaining hands if he could speak with them after the call implying they were nonsense.

Joking aside the important part of this talk, the conference as a whole is ultimately the realization that DevOps is a culture, and not a process and that no matter what we do, or how well we do it, human beings will always be at the center of our software. The software we deliver ultimately reflects the structure and culture of our organizations. That being said, we as leaders would do well to take special care of our humans!