As businesses scale, it is vital that they maintain the speed at which they are able to reliably deliver business value to their customers. Businesses which find themselves losing their agility can find themselves disrupted by the competition.

Software allows businesses to test new business ideas in the real-world and ship new functionality to a global user base quicker than ever. However, as businesses scale, it is easy for them to lose this ability to ship quickly. Bureaucracies build, technical debt grows and processes become inefficient.

In high-performing tech companies, Engineering Productivity works alongside other engineers to identify and remove bottlenecks. The team names vary from organisation to organisation; whilst Google has EngProd, Netflix has Developer Productivity (DevProd). However, the results are broadly similar.

Engineering teams which invest in engineering productivity see their teams able to ship more reliable work faster, without having to burnout their teams to do so.

In this report; we review the common trends in the EngProd field over 2021; from benchmarks to trends in software reliability, developer burnout and team culture.
Methodology

This review considers industry reports into developer productivity produced during 2021.

In the production of this report, we’ve considered three key sources of data. Both Puppet [1] and Google Cloud [2] have produced State of DevOps Reports for the year 2021.

As DevOps typically focuses on later stages of the Software Development Lifecycle (from code written to deployment), so it is also important to ensure we have a data source that more broadly covers the area of developer productivity.

Our third form of data is a study conducted by Haystack & Survation into the impact of COVID-19 on developers [3]. This study was conducted during 2021 and focuses more broadly on the entire Software Development Lifecycle.

We have then qualitatively compared the benchmark results between the different reports and have qualitatively identified common themes.

Benchmarks

We start by considering the benchmarks between different engineering teams. Google’s DORA team have performed benchmarks during their 2018 and 2019 State of DevOps reports. They took a hiatus during 2020 and returned with another report in 2021. These benchmarks are assessed on the basis of some key metrics for assessing DevOps performance (Change Lead Time, # of Deployments, Time to Recovery and Change Failure Rate).

<table>
<thead>
<tr>
<th></th>
<th>DORA 2018</th>
<th>DORA 2019</th>
<th>Haystack/Survation 2021</th>
<th>DORA 2021</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elite Performer</td>
<td>7%</td>
<td>20%</td>
<td>31%</td>
<td>26%</td>
</tr>
<tr>
<td>High Performer</td>
<td>48%</td>
<td>23%</td>
<td>47%</td>
<td>40%</td>
</tr>
<tr>
<td>Medium Performer</td>
<td>37%</td>
<td>44%</td>
<td>16%</td>
<td>28%</td>
</tr>
<tr>
<td>Low Performer</td>
<td>15%</td>
<td>12%</td>
<td>4%</td>
<td>7%</td>
</tr>
<tr>
<td>Don’t Know</td>
<td>-</td>
<td>-</td>
<td>2%</td>
<td>-</td>
</tr>
</tbody>
</table>

*Table 1: Performance profiles from DORA State of DevOps reports are calculated on the basis of four dimensions. Haystack/Survation performance profiles are based purely on Cycle Time metrics.*
Haystack/Survation also provided some benchmarking of software delivery, asking respondents about how long it took them to take a feature through development to being in a production environment. This one dimension, Cycle Time, is effectively a more broad version of the Change Lead Time metric used by Google’s DORA team.

As Table 1 shows, the trends from Google’s DORA results in 2018 and 2019 to the results captured in 2021 are consistent. Both Haystack/Survation and DORA show growth in “High” and “Elite” performer groups, whilst those in the “Medium” and “Low” performer groups have fallen.

Table 2 highlights how competitive feature delivery is becoming between software teams. Haystack/Survation data shows that a majority of Software Engineers, 59%, report Cycle Times of less than 3 days at their workplaces. 31% report Cycle Times of less than one day in their workplace.

<table>
<thead>
<tr>
<th>Cycle Time</th>
<th>% Responses</th>
<th>Cumulative %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Within a few hours</td>
<td>9%</td>
<td>9%</td>
</tr>
<tr>
<td>On the same day</td>
<td>22%</td>
<td>31%</td>
</tr>
<tr>
<td>Within two to three days</td>
<td>28%</td>
<td>59%</td>
</tr>
<tr>
<td>Within four to six days</td>
<td>10%</td>
<td>69%</td>
</tr>
<tr>
<td>In a week’s time</td>
<td>9%</td>
<td>78%</td>
</tr>
<tr>
<td>Within two weeks</td>
<td>8%</td>
<td>86%</td>
</tr>
<tr>
<td>Within three weeks</td>
<td>2%</td>
<td>88%</td>
</tr>
<tr>
<td>In a month’s time</td>
<td>6%</td>
<td>94%</td>
</tr>
<tr>
<td>In over a month</td>
<td>4%</td>
<td>98%</td>
</tr>
<tr>
<td>Don’t know</td>
<td>2%</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 2: 59% of Software Engineers report Cycle Times of less than 3 days at their workplaces according to Haystack/Survation data [3].

Benchmarking was performed slightly differently in Puppet’s State of DevOps report. Historically they have only subdivided performance into three groups; “High”, “Mid” and “Low”.

Consistent with Google’s DORA research and the Haystack/Survation research, Puppet found that their “Low” tier had reduced. However, finding that their “Mid” tier had remained stagnant, this year they have begun to split that tier out into “high-mid”, “middle”, and “low-mid”.
Across the board, research this year has shown that percentage of low performing teams has dropped. At the same time, research indicates that “High” and “Elite” performing groups have grown. This indicates that companies are getting better at adopting DevOps and EngProd practices and that the landscape is now more competitive than ever before.

Reliability

The Haystack/Survation study asked developers to what extent they agreed with the phrase: "Software reliability at my workplace concerns me". 57% said they agreed “to a great extent” or “to a moderate extent”.

Whilst 20% of respondents reported that software reliability concerned them to a great extent, just 16% of respondents reported that software reliability at their workplace was not a concern for them at all.

Whilst software reliability concerns often take the form of concern for customers and wider society, Haystack/Survation research has also indicated it can be a cause for burnout amongst developers themselves. Of software developers reporting being burnt out from work, 20% reported that unreliable software was a contributing factor. This can be due to phenomena like “alert fatigue” for software engineers who are on-call for operational reasons and are being paged constantly.
This is complimentary to Google’s DORA research, which found that teams which employed modern Site Reliability Engineering (SRE) practices are less prone to burnout.

Google’s DORA research had an increasing focus on reliability this year. Whilst historically they have used Four Key Metrics to assess DevOps performance (Change Lead Time, # of Deployments, Time to Recovery and Change Failure Rate), this year they added a fifth dimension.

This year, the Google team also asked respondents about their ability to meet or exceed their reliability targets. This is significant as historically DORA reports have measured availability as their reliability dimension (using metrics like Time to Recovery and Change Failure Rate) - this year they expanded that to cover software reliability more broadly.

In the report, the Google DORA team claim they "saw major benefits across multiple outcomes for teams that prioritized meeting or exceeding their reliability targets".

The research did also find that even “elite” performers have room for growth, with only 10% in that performance group fully implementing every Site Reliability Engineering practice that the report investigated.
Elite performers were 2.1 times as likely to report using modern Site Reliability Engineering practices than their low-performing counterparts. **Teams which excel at modern operational practices were also 1.8 times more likely to report better business outcomes.**

**Burnout**

Burnout was a central theme of the Haystack/Survation study and was also studied in the Google report. The Haystack/Survation study found that 83% of software developers suffer from workplace burnout. The top reasons cited for burnout included high workload (47%), inefficient process (31%) alongside unclear goals and targets (29%).

![Q1. To what extent, if at all, do the following statements apply to you?](image)

*Figure 3: 50% of Software Engineers report Cycle Times of 1-3 days at their workplaces. In only 4% of cases do Cycle Times exceed a month.*

81% of developers reporting increased burnout due to the pandemic. The top reason cited by developers for greater feelings of burnout during the pandemic was increased workload.

Google's report found that 89% of respondents worked from home due to the pandemic despite only 20% reporting having ever worked from home prior to the pandemic.

Google's report also found that teams with a generative culture, featuring belonging and inclusion, were half as likely to experience burnout during the
pandemic. Performance-oriented cultures, where employees are encouraged to bring their whole self to work and where their unique experiences are valued, are more likely to see lower levels of burnout than workplaces with less positive organisational cultures. Generative cultures are performance-oriented and encourage psychological safety (with shared risks, blameless postmortems and novelty encouraged).

Psychological Safety

For all the discussion of the impact of individual practices on metrics, there is one central point of convergence between all the data - the critical importance of psychological safety.

The Puppet report found that low performing teams were 2.2 times more likely to have a culture that discourages risk than their high performing counterparts.

The report goes on to state: “The result of all this is that those organizations that claim to be discouraging risk are actually following practices that increase risk, and many of their existing practices around risk management of infrequent deployments are simply risk management theater”.

Over multiple years, Google’s DORA team have studied the importance of psychological safety on developer teams. As the Google team studied Site Reliability Engineering this year, they noted: “SRE is a learning discipline that prioritizes cross-functional communication and psychological safety, the same values that are at the core of the performance-oriented generative culture typical of elite DevOps teams.”

As the Google report makes clear: “High-performing organizations are more likely to have a culture that encourages employees to take calculated and moderate risks without fear of negative consequences.”

This matches earlier findings by Google in Project Aristotle [4]. The researchers studied a total of 180 teams, including 115 project teams in engineering, using double-blind interviews and survey data. When assessing the dynamics which made teams successful, Google reported: "Psychological safety was far and away the most important of the five dynamics we found".

This reaches the fundamental dogma of what is at the heart of much of software reliability. For companies to achieve commercial success, they must be able to take calculated risks. The inability to do this leads to poor software reliability and poor business outcomes. No risk, no reward.
Good professional engineering judgement is vital to being able to mitigate serious risks before they become measurable consequences, but often minor bugs can be tolerated within thresholds. When this is the case, it’s often most important for customers that minor bugs are resolved promptly once reported (hence why Full Resolution Time for Bugs is an important metric).

Engineering Council UK have produced excellent Guidance on Risk [5] which spans much of the professional engineering skills that are required to be effective at managing risk.

Psychological safety is embedded within this guidance too, for example:

- “Establish strong, honest and effective two-way communication within and beyond their organisation”

- “Express clearly the balance of risk and benefit”

- “Encourage an ‘open reporting’ approach, and a spirit of questioning and learning from others”

- “Avoid a ‘good news only’ or closed culture”

- “Be prepared to engage in public debate on the perceived risks and benefits”

- “Be honest and clear about uncertainties, and be prepared to challenge misrepresentations”

Conclusion

In this report we assessed the state of engineering productivity in 2021. To do this, we have considered 2021 State of DevOps reports from Puppet [1] and Google’s DORA team [2] alongside a report on the impact of COVID-19 on software engineers produced by Haystack & Survation [3].

The software development industry is increasingly competitive.

Benchmarks have consistently shown improvements in performance amongst software engineering teams, with the proportion of low performing team declining whilst high and elite performing teams have grown.

Haystack/Survation data shows that burnout is a clear problem, with the overwhelming majority of software developers suffering from it. Google’s report found that teams adopting a generative culture (performance-oriented, shared risks, high cooperation) were less likely to suffer from burnout.
Research has increasingly focused on software reliability during 2021. The Haystack/Survation report found that 57% of software developers said they agreed “to a great extent” or “to a moderate extent” with the phrase: “Software reliability at my workplace concerns me”.

Just 16% of respondents reported that software reliability was not a concern for them at all.

Google’s DORA team also found there was room for improvement in software reliability. Even amongst “elite” performing teams, only 10% had fully implemented every Site Reliability Engineering practice that the report investigated. At the same time, the Google team found clear business advantages for doing so, finding that teams which excel at modern operational practices were also 1.8 times more likely to report better business outcomes.

Shared amongst all the research was the critical importance of psychological safety.

No risk, no reward. For businesses to grow, there must be a willingness to take calculated risks and learn collectively. It is therefore critically important that engineers are equipped to balance risk and reward.


The Puppet report found that low performing teams were 2.2x more likely to have a culture that discourages risk than their high performing counterparts.

The Google report states that: “High-performing organizations are more likely to have a culture that encourages employees to take calculated and moderate risks without fear of negative consequences.”

The key to improving software reliability is rests in ensuring software engineers are well equipped to make professional judgements on the balance of risk and reward. To do so effectively, psychological safety is essential.
References

2. DORA 2021 Accelerate State of DevOps (Google Cloud)
3. Study to understand the impact of COVID-19 on Software Engineers - Full Report (Haystack/Survation)
4. The five keys to a successful Google team (Google People Operations)
5. Guidance on Risk (Engineering Council UK)