There are alternatives that can be considered which allow unified monitoring and alerting systems. We can either choose Open source tools like Prometheu or ELK. There are observability platforms such as Dynatrace or Datadog that are able to help with this.

While i have seen teams work well with tools like ELK that they maintained internally, i would opt for 3rd party observability platform like Dynatrace because of reasons below:

### Resources

When it comes to collecting and aggregating logs of resources from AWS and Azure - we are collecting, aggregating, transforming and processing huge amounts of data. This would require huge resources to be able to support and maintain this in-house. SRE teams should focus on ensuring core business reliability.

### Integrating observability into application

One of the most important questions is - how do I integrate observability into my application and how do I add my custom critical metrics into the platform. This is one of the most time consuming tasks as it involves collaboration between SRE and the development team. There would be a task to build a dashboard for different teams.

### Integrations that works

Dynatrace has tools that work and are able to aggregate data from Azure or AWS. While it is not perfect for example trying to aggregate metrics from Azure service bus, some metrics are not available, for example server send latency. There’s new releases on a monthly basis to improve data extraction processes.

But for the most part, logs and traces are available. SRE team would need to install the proper component such as the dynatrace operator in a kubernetes cluster to turn on cluster monitoring.

Operators that are installed sometimes do provide better application instrumentations such as memory footprint or cpu usages by processes.

### Standard UI Access.

It is important to have a single place to go to when you need to view and query logs for all team members that includes developer and SRE team. This helps to avoid asking questions where I need to go to query traces for my payment app deployed to Azure or where to query for my mobile app hosted in AWS. The answer will always be one.

### Costing

Azure Log Analytics can be pretty expensive. Applications that write traces and logs can quickly bump up the costs. Having an observability platform such as Dynatrace can help to reduce these costs.

### Logs retentions

It is important to understand how long the logs will be retained and remain queryable. For example, if I am trying to find out when a user starts having some login issue - it is important to know how far back can i go.

### Learning curve

There will be some learning curve involved as this system uses a custom query language for example in Dynatrace - it uses DQL to query and filter logs. Comparatively, it is less time consuming to learn both AWS and Azure query language.

### Alerting

Creating alerts in Dynatrace involves defining alerting profiles or rules based on metrics and conditions relevant to your environment. By configuring appropriate thresholds, notification channels, and escalation policies, you can ensure timely notifications and effective incident management. Integration to PagerDuty is also possible to ensure support teams are alerted when an incident occurs.

### Summary

The strategy for centralised logging and visualisation from Azure and AWS is to use a 3rd party. The first set of tasks is to set up the integration point which includes access to an AWS or Azure subscription to pull out necessary logs. Then improves observability by rollout monitoring to services like Azure service bus or SQS. The SRE team would work together with development team to onboard application, create key alerts and dashboard in the observability platform.