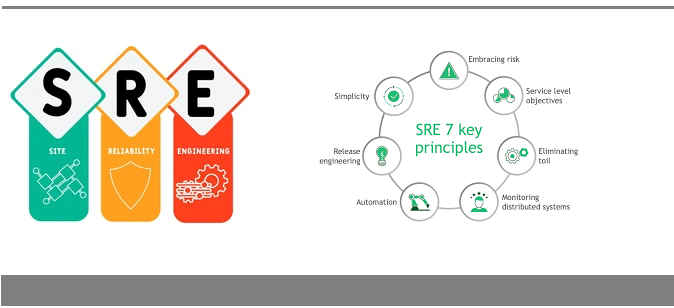
SRE by Google <https://sre.google/>

SRE by IBM <https://www.ibm.com/topics/site-reliability-engineering>

SRE by Amazon <https://aws.amazon.com/what-is/sre/>



### 1. Risk

Evaluating the risk of unexpected failures is critical to improving any system’s reliability.

Achieving the five nines of system reliability is a challenge and systems can fail anytime. To learn how to make services and systems more reliable, SRE teams must understand risk and embrace the challenges to solve them.

Embracing risk requires evaluating the cost of increasing reliability and its impact on customer satisfaction.

### 2. SLOs

In order to meet Service Level Agreement (SLA) teams often set themselves Service Level Objectives (SLOs) to reflect targets of system reliability such as uptime.

Having SLOs mean that SREs and engineering teams have a benchmark to meet when it comes to providing service reliably.

### 3. Eliminating Toil

[Toil](https://reliably.com/blog/eliminating-toil-in-sre/) is a term coined by Google that describes the amount of manual work necessary to keep a service running. SREs aim to eliminate toil - automation is a standard method to do this, enabling SREs to have more free time for more critical tasks.

Sometimes, an SRE is required to perform manual tasks, but not all tasks should be considered toil. Determining which SRE team activities take up the most time is essential.

Teams can reduce toil by adding guidelines and procedures for jobs. An SRE’s goal should be to create features that reduce toil while increasing reliability and productivity.

### 4. Monitoring

Monitoring is examining your system’s valuable data and making decisions based on it.

Monitoring gives SRE teams a historical performance trend and presents insight into what makes up a separate issue versus a more extensive systemic problem.

The [four golden monitoring signals](https://reliably.com/blog/four-golden-signals-of-monitoring-for-sre/), described by Google’s SRE initiative, are as follows:

* **Latency:** is the time it takes for a service to reply to a request
* **Errors:** the frequency with which services fail
* **Traffic:** a system’s load or the amount of users accessing the service
* **Saturation:** the number of system resources available to a specific service

### 5. Automation



Automation is the process of devising methods for carrying out repetitive operations devoid of human involvement.

An SRE’s role is complex by nature, and automating tasks is essential to minimizing the possibility of manual intervention.

When SREs automate tasks, they save time and can focus on achieving the company’s reliability goals.

Tasks that could be automated include:

* Testing
* Software deployment
* Incident response
* Team communication

Manual procedures are more likely to go wrong and often cause problems. Automation speeds up processes and makes mistakes less likely. In addition to getting rid of toil, it is a crucial way to improve reliability and speed.

### 6. Release Engineering

Release engineering is an approach to describing the creation and delivery of software. Defining best practices for creating software services, providing updates, continuous testing, and resolving software issues often falls to release engineers.

At Google, Release Engineers work on product development with Software Engineers (SWEs).

Release engineers work with SREs to define all the steps required to release software, from storing the software in the source code repository to building rules for compilation to how SREs perform testing, packaging, and deployment.

The following are some of the qualities of effective release engineering:

* Process documentation
* Automation
* Rapid deployment
* Configuration management
* Testing

### 7. Simplicity

Simplicity entails creating the least complex system possible whilst ensuring optimal performance.

In real-world applications, this principle emphasizes the notion of designing a system that is only as complex as required.

SREs, try to ensure that a system is not too complicated or hard to manage. However, from the user’s perspective, a service with many features may also offer many benefits.