Tableau Server Scalability

Introductory Overview to Scaling Tableau Server across Your Enterprise

Neelesh Kamkolkar
Product Manager, Data and Performance

tableau®

Worldwide, organizations of all sizes are delivering true self-service analytics at scale with Tableau. They are strategically transforming their organizations and building an analytical culture that will be critical to their future. As that culture builds, your analytics platform becomes mission-critical to your organization, and downtime becomes a real concern. Proper deployment and scalability planning are essential to minimizing downtime and ensuring your analytics platform can meet the ever-growing needs of the business.

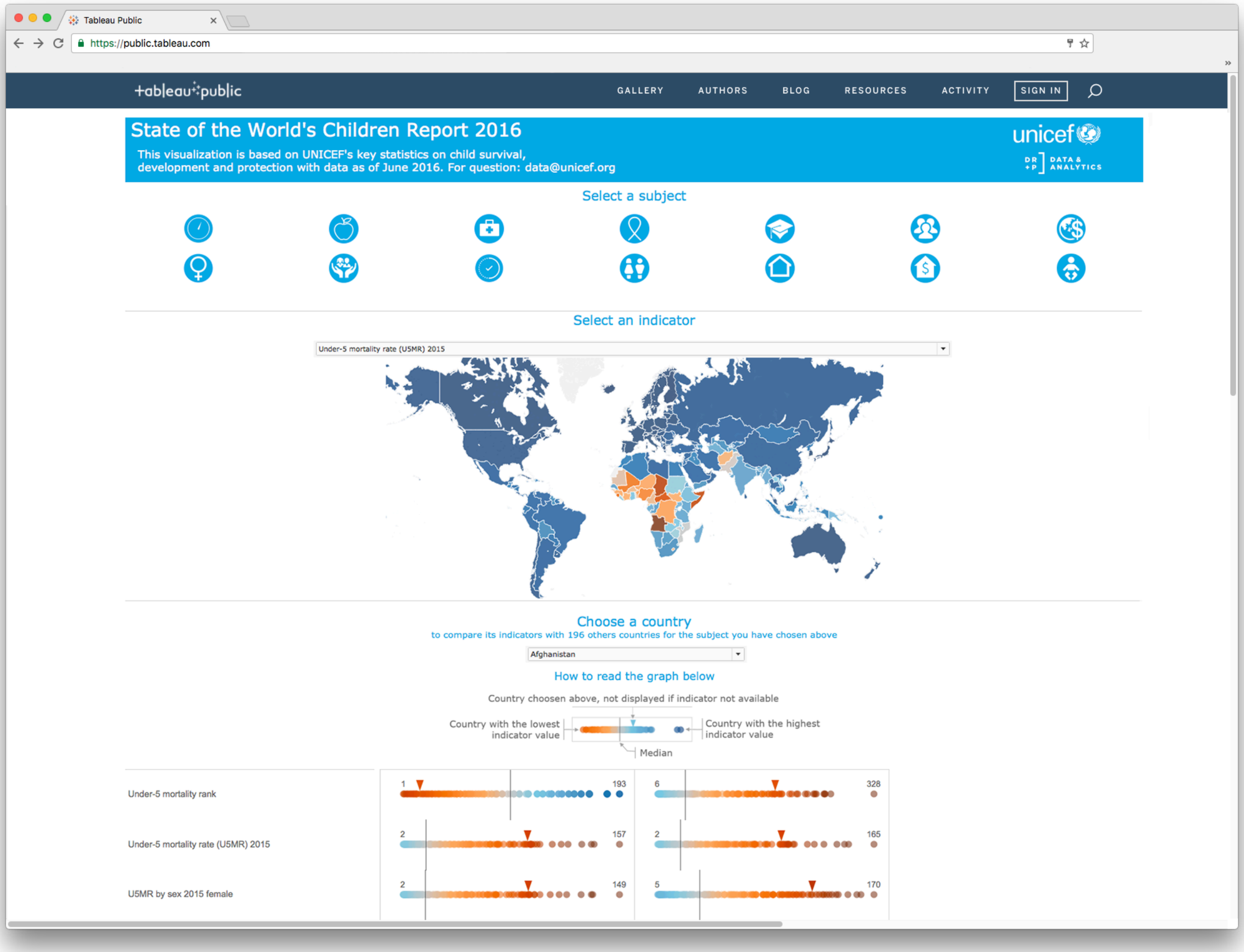
There are many things to consider when setting up an enterprise environment that is built to scale as your business grows. This paper outlines the key things to consider when setting up your Tableau environment and also includes a framework for scalability. It also shares summary results from our scalability labs and provides high-level guidance for planning your deployment and preparing for future growth.

For audiences that want a more in-depth understanding of the experiments, methodology, environments, observations, and results, a deep-dive technical companion to this paper is available HERE.

Table of Contents

Tableau's proven scalability and reliability

You can be confident that Tableau Server can scale to support the needs of your organization. To test and validate that Tableau 10 meets scalability release goals, we deployed Tableau Server 10 to Tableau Public, our free platform for sharing visualizations with the world. The Tableau Public deployment spans a horizontal cluster of many machines supporting over seven million views a week, so it is a great way to test Tableau Server scalability at massive scale.



State of the World's Children Report 2016

This visualization is based on UNICEF's key statistics on child survival, development and protection with data as of June 2016. For question: data@unicef.org

Choose a country

to compare its indicators with 196 others countries for the subject you have chosen above

How to read the graph below

Country choosen above, not displayed if indicator not available

Figure 1 : The Tableau Public gallery showcases many visualizations that are accessed using guest privileges.

Aside from a handful of configurations, the underlying architecture that is powering Tableau Public is exactly the same as Tableau Server that is deployed on-premises in organizations like yours.

And Tableau Public scales to support more than 200,000 authors, 600,000 visualizations, and seven million views a week. As part of release process, our entire Tableau Public deployment was running on Tableau 10 during the beta periods before the official release of Tableau Server, truly testing the platform's reliability at an enormous scale.

Framing Tableau scalability

As you build an analytical culture within your organization, users will increasingly rely on data insights to make smarter business decisions. By being empowered to explore trusted data, they will start to see things in the business they couldn't see before. As users recognize the value and ease of answering their own questions, utilization of analytics will increase as will your users' expectation of data freshness. User adoption of analytics and data freshness, paired with your organization's risk tolerance, are the primary areas that must be considered when architecting your deployment strategy and your scalability plan.

Users' impact on scalability planning

Driving user adoption is the obviously one of the main goals of implementing self-service analytics. If you find the right platform and deploy it in a smart and scalable way, that platform will become mission-critical to your users' everyday business. It is for this reason that it is essential to consider usage patterns and user adoption when defining your plan.

The impact of self-service on scalability

While report bursting is common practice in traditional business intelligence deployments, it focuses primarily on report proliferation and consumption. But report bursting fails to empower users in a critical scenario; users are unable to quickly modify the report to ask their own questions of the data. Instead, looking at the data from a different perspective and asking new questions likely requires a round-trip to the analytics team that may take days or weeks, if not months.

Self-service is at the foundation of every organization that has developed a culture of analytics, as it combines its two greatest assets, its people and its data. People who know the business well must be able delve deep into their own data, allowing them to explore it from every angle. Tableau's self-service analytics platform allows everyone to ask questions directly of the data they care about. Tableau's approach to self-service, its proven visual-analytics engine, and patented VizQL technology enable everyone to immediately begin interacting and exploring the data without the shackles of a report development cycle.

The Tableau Way

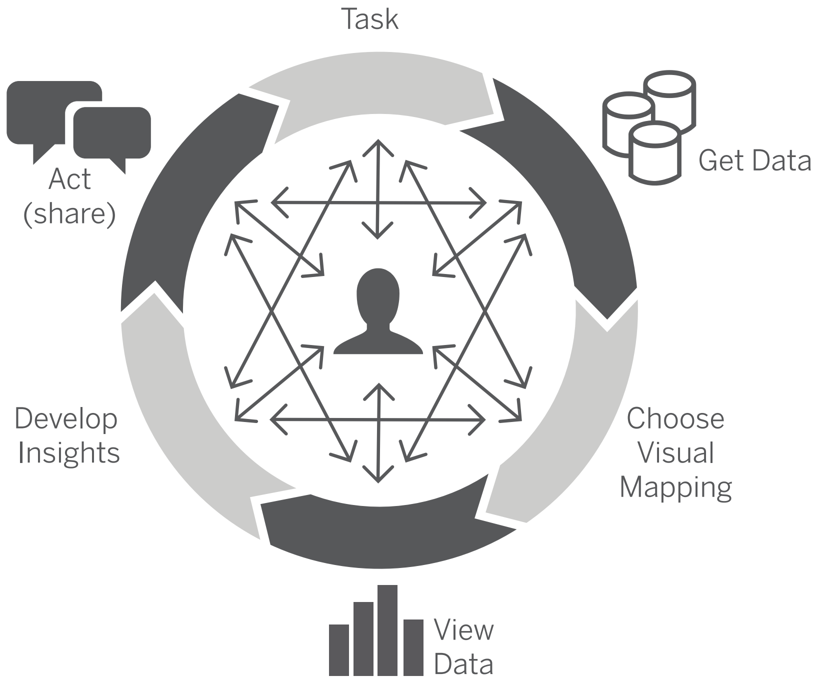


Figure 2: The analytics cycle of visualization

Giving everyone this ability to explore data to their heart's content and the opportunity to unearth hidden insights is highly motivating. As users discover and share their insights with Tableau, adoption increases across the organization. Therefore, capacity and resource planning for Tableau Server must account for the impact of user adoption.

The impact of user adoption on scalability

Tableau provides great analytical rigor, empowering users to ask more and deeper questions of the data. With Tableau Server, users' workbooks and insights can be shared in a secure and governed environment, allowing others to begin exploring existing workbooks so they can quickly jump into their analysis. The more workbooks that are created and shared, the more content is available for more people to jump start their analysis. This type of environment, fostered by Tableau Server, drives engagement and usage of analytics across the organization.

An important planning metric is the frequency with which users are accessing and interacting with their analytics on Tableau Server. Typically, Tableau Server installations encourage a sharp uptick in data analytics in most organizations. But the more content and data you make available to your users, the more they will come to rely on analytics through their decision-making process. Estimating frequency of use is part of planning your initial Tableau deployment. After you have deployed Tableau Server, use our powerful monitoring tools that are included with the platform to measure frequency and usage characteristics. Reviewing usage data will inform how you tune and scale your environment to the continuously-changing needs of the business.

Building a community of practice

To promote and scale a culture of analytics within their organization, many customers are creating active internal user communities, promoting Tableau champions via community programs, hosting weekly lunch-and-learns, incorporating free Tableau training into onboarding programs, and even reaching out to the extensive community of Tableau experts. As more people start using self-service analytics, sharing and collaboration become an important part of allowing the culture to grow. This growth, in turn, impacts the adoption and engagement of your users as well as the workloads running on Tableau Server. To ensure an optimal user experience as well as sufficient capacity to handle the increased workloads, you must plan your infrastructure for this growth.

Data's impact on scalability planning

With Tableau, users can connect to their data regardless of whether it's in the public or private cloud, on-premises, or a mix of all three. With Tableau Server, your users have the ability to connect live to data sources that host time-sensitive data. Or your users can create extracts for data that doesn't change frequently. These extracts can be hosted directly on Tableau Server as a saved data source. You can then refresh those extracts as often as you like, or have them scheduled for refresh automatically. With just a few clicks, your data stewards can switch between extracts and live connections. This way, they can prototype on a subset of the data offline, then reconnect to the entire database when they're back online.

Tableau gives your users the flexibility to optimize access to your data in a way that makes the most sense for your business. Empowering your users to make business decisions means giving them access to the data they need when they need it. Moving the business user closer to their data drives flexibility and agility, but also makes factors like data freshness, data size, and location important to consider in ensuring your environment scales well.

Data freshness

The right business decisions are made on fresh data. However, in some cases, users don't always need up-to-the-minute data. For example, consider a daily bug triage meeting. In this scenario, the data only has to be refreshed once a day because the bug triage meeting is a recap of the last 24 hours, not up to the minute.

On the other hand, consider a support queue where your goal may be to actively help and respond to customers on the phone. In this case, data may need to be refreshed every few minutes during the workday, and not just once every 24 hours.

Refreshing data can be a strain on your infrastructure. As you plan your Tableau deployment, carefully consider your refresh requirements based on the data itself and the needs of the business so that you don't place an unnecessary refresh burden on your environment. For example, using Tableau Server's content analytics, identify workbooks that are not being used at all yet have data extracts that are being refreshed. Partner with and iterate with IT and the business to identify the most appropriate data-refresh rates given content analytics and usage of the workbooks. These simple tricks may help you deliver a better quality of service overall to your end users.

Data size and location

More than ever, the size of data is growing while the frequency of access is accelerating and the variety of data source types is expanding. Running and analyzing all of that data in-memory is not always feasible. And while data may be spread across a number of databases on-premises, in the cloud, or even in flat files, users require the ability to see, interact, and visualize it, regardless of where it is located.

Users must have the flexibility to use in-memory analytics for fast performance where needed. They must also be able to leverage existing investments in database infrastructure to run their analytics against live databases.

In the past, business users depended on someone with SQL expertise to build a report from data across different relational database management systems and flat files. With Tableau, cross-database joins and Unions bring the power of data, regardless of size or location, directly to the business users—without the need for expert SQL skills.

For databases that are frequently updated, users can connect with a live connection for access to the latest data. However, if performance is a concern and the business scenario does not require real-time data access, then you can easily extract that same data to leverage Tableau's in-memory technologies. You can then tune your refresh schedule for that data extract according to your business requirements.

How often you choose to refresh data to make accurate business decisions will be important as you plan your initial capacity and consider future data needs across your organization.

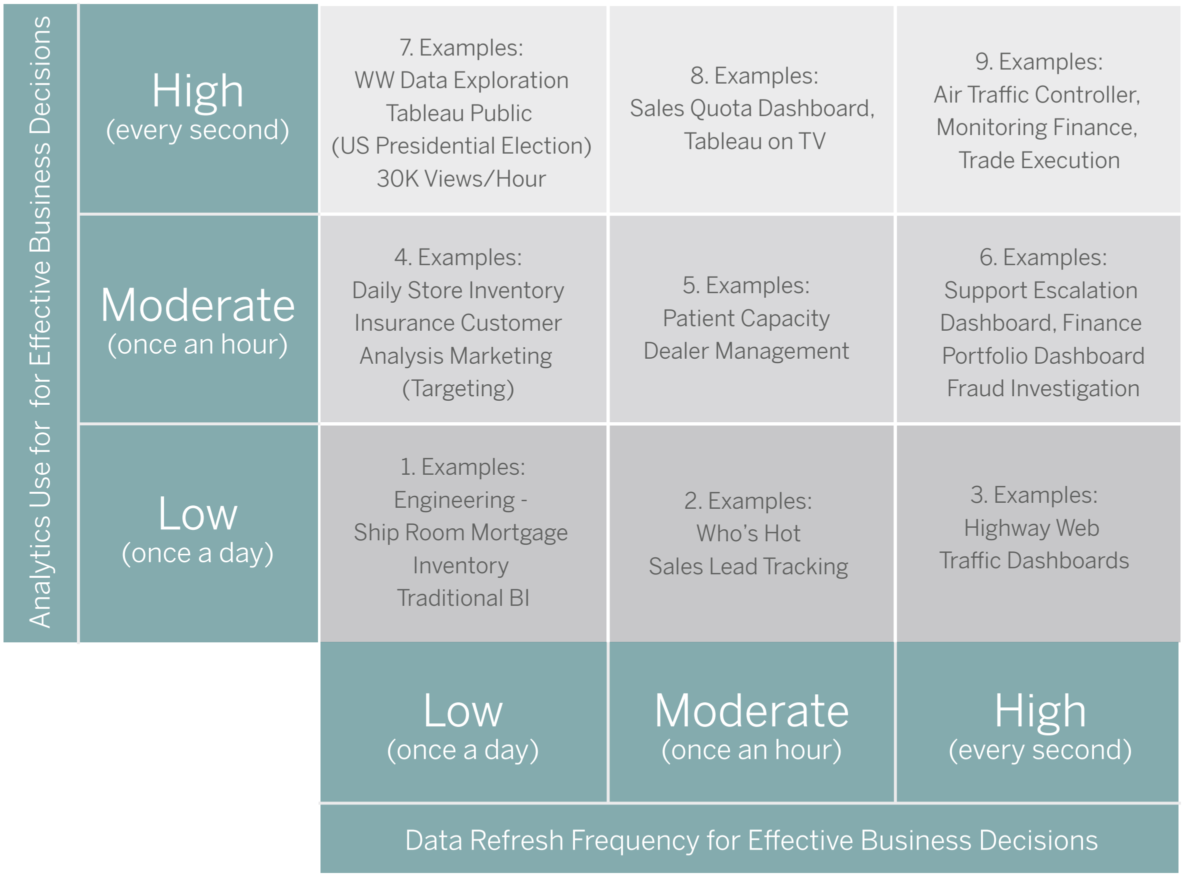
A framework for planning your tableau scalability

While there are many variables that inform scalability of a deployment, those that we've discussed are important factors to estimate as you start planning:

- User impact - self-service usage and user adoption: How many users will be using analytics? How often will users employ analytics to make an informed decision? How complex are visualizations that users are creating?

- Data impact – freshness, size, and location: How big is your data? Where is the data located? How fresh does the data need to be to inform business decisions accurately?

As you plan your Tableau deployment, you should attempt to quantify these vectors. This section will discuss how to think about scaling Tableau in the context of these factors.



Analytics Use for for Effective Business Decisions				
	High (every second)	7. Examples: WW Data Exploration Tableau Public (US Presidential Election) 30K Views/Hour	8. Examples: Sales Quota Dashboard, Tableau on TV	9. Examples: Air Traffic Controller, Monitoring Finance, Trade Execution
	Moderate (once an hour)	4. Examples: Daily Store Inventory Insurance Customer Analysis Marketing (Targeting)	5. Examples: Patient Capacity Dealer Management	6. Examples: Support Escalation Dashboard, Finance Portfolio Dashboard Fraud Investigation
	Low (once a day)	1. Examples: Engineering - Ship Room Mortgage Inventory Traditional BI	2. Examples: Who's Hot Sales Lead Tracking	3. Examples: Highway Web Traffic Dashboards
		Low (once a day)	Moderate (once an hour)	High (every second)
		Data Refresh Frequency for Effective Business Decisions		

Figure 3: A simple business framework for scalability

Understanding Tableau Server processes and how they support various server-load functions is essential for tuning and scaling your Tableau Server deployments. See Tableau Server Processes for more information.

In the matrix above, the vertical axis represents frequency of active use of analytics on Tableau Server. There are a handful of processes associated with this functionality that are considered "user-facing." The horizontal axis represents the frequency of data refreshes necessary to provide the latest data to your organization to make business decisions. The primary process that supports data refresh is the Backgrounder process. Increasing process instances according to these primary use cases is the first step in scaling Tableau Server.

First, plot your organization's use profile on the matrix above to help determine your starting place for the initial Tableau deployment and further tuning. Tableau Server scales effectively both horizontally and vertically. Scaling your deployment is primarily a function of adding more process instances or additional cluster nodes (workers) to the corresponding use profile.

Other important factors that informs both performance and scale are how users author Tableau visualizations and how complex their visualizations turn out to be. Anyone can author Tableau visualizations easily with Tableau Desktop or with Web Authoring on Tableau Server, but when they create a workbook that will see very high usage, they should follow additional best practices focusing on performance, scalability, and reliability. For more information, see Designing Efficient Workbooks.

Example scenarios

The following scenarios provide some examples of different ways to scale Tableau Server. These scenarios are based on real-world modeling of workload from a production environment and tests run and simulated in our scalability labs to inform scalability.

Small single-server deployment

For this scenario, where your risk profile allows you to tolerate some downtime, consider a single 8-16 core server with 64 GB–128 GB RAM. Ensure that there is sufficient disk IO bandwidth (>500 MBPS) available for Tableau Server. Our minimum RAM and disk recommendations will work great for installation purposes and trial deployments, but we generally recommend a minimum of 8 GB of RAM per each core allocated to Tableau Server for production deployments.

We have engineered the default "out-of-the box" Tableau Server configuration to optimize for smaller deployments and departmental servers. But the Tableau Server architecture gives you flexibility in how you scale to support your use profile.

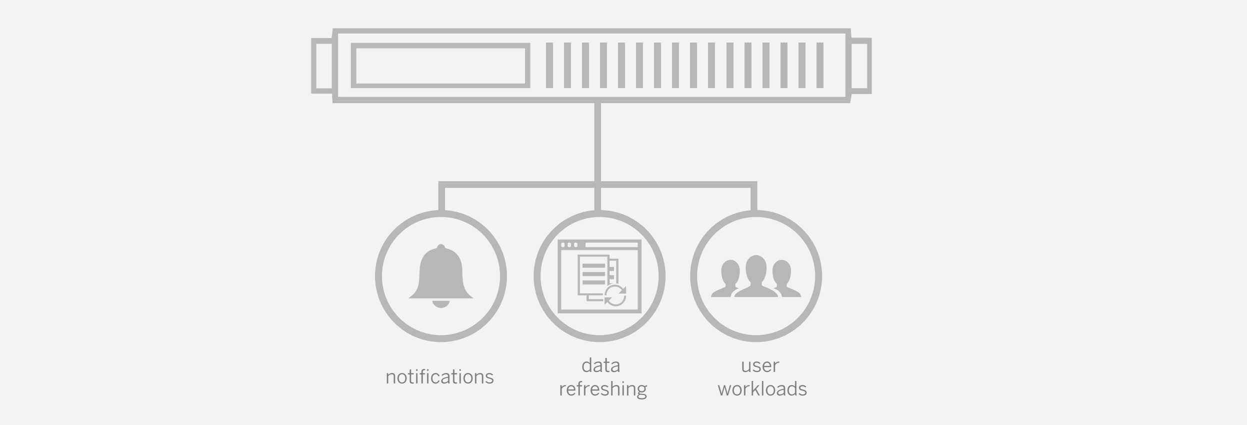


Figure 4: Deployment for a single machine

A single-server deployment is easy to set up, configure, maintain, and manage for smaller teams with minimal IT support. See Tableau Server: Everybody Install Guide for end-to-end guidance on planning for, installing, and maintaining a single server deployment.

Generally speaking, this deployment is sufficient for servicing a small organization where frequency of data refresh is low and frequency of use is low to moderate. You should not impose high frequency of data refresh on this deployment because the server processes responsible for data refresh and analytics use will compete for system resources on the same machine. We measured about 10% degradation on end-user quality of service when data refresh workloads are brought onto a single-machine deployment.

It's important to consider that a single-server deployment may compromise performance in load bursts that exceed your use-profile planning. Obviously, a single server also impacts availability in the case of failure. While Tableau Server provides out-of-the-box, process-level high availability (processes restart automatically if they fail), in this scenario, a hardware failure will result in downtime.

Dual-server deployment

You can moderate the risk of downtime while improving performance with a two-server deployment. While a two-server deployment does not give you the same fault tolerance as a multiple-server distributed cluster, running a two-server deployment is a cost-effective way to provide scale at a lower risk than a single-server deployment.

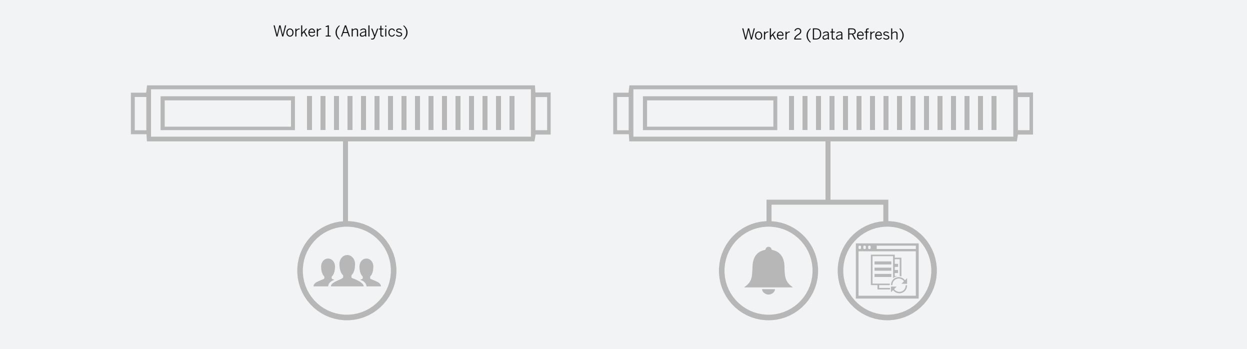


Figure 5: Deployment for two machines

With Tableau Server's flexible architecture, you can split workloads across servers. You might have one server servicing the end-user analytical workload and dedicate the second server to Backgrounder jobs that refresh data and deliver notifications to end users. Compared to a single-server deployment, the dual-server dedicated deployment will allow more frequent refreshes of data without impacting user-facing performance.

While your results may vary, based on our testing, for the workloads we used we pushed Tableau Server to sustained loads with 80% CPU utilization, and we found a properly configured two-node server cluster could support anywhere between 1,000 and 5,000 users.

For example, a Tableau Server deployment of up to 16 cores can support an organization of 1,000 people where about 10% of them are active for an hour (peak usage time) and their business requires data to be fresh every 24 hours. Twelve of the cores would be allocated to analytics use and the remaining four cores for data workloads and keeping data fresh. Depending on how you spread your analytics usage and data workloads as well as your focus on ensuring performant design for workbooks, you may be able to push more users on this system or require an additional 8 cores for running very complex and slow-running workbooks.

While the above information is based on our test results, your results will likely vary depending on your use cases. You may want to consider conducting an architecture review to better understand or inform your sizing needs over time. A review can help ensure you are not just planning for a point in time, but also planning for your projected Tableau usage growth, business requirements, and risk mitigation.

Adding single-purpose cluster nodes

You can also scale up specific components of Tableau functionality to support your business requirements. To this end, you may add single-purpose nodes to a cluster.

Supporting fresh data

Data freshness in the context of Tableau Server is determined by how you manage extract refreshes. Specifically, be sure to have sufficient capacity to ensure that extracts are refreshed in a timely manner for use by business users.

For this scenario, the scale unit that matters the most is the Backgrounder process. Among other improvements, adding Backgrounder processes accelerates extract refreshes and ensures subscriptions to your visualizations go out in a timely manner.

A common deployment configuration for organizations with requirements on real-time data or frequent data extraction schedules is to add a node to the Tableau Server cluster specifically to run the Backgrounder process. Running this the Backgrounder offloads the CPU and processing from the other nodes, thereby freeing up more cycles for servicing front-end requests, such as visualizations and authoring tasks.

Mission-critical deployment with high-availability requirements

If your business requirement prioritizes uptime (availability), then deploying at least three nodes (starting in Tableau 10) is a requirement to ensure that you offset the risk of hardware and software failures. Describing how high availability works is outside the scope of this document. Please refer to our in-depth technical whitepaper, Tableau Server High Availability: Delivering mission-critical analytics at scale.

Multiple-server, extract-heavy deployment

If your business has a very low-risk tolerance for downtime, consider deploying a multiple-server distributed cluster that gives redundancy of deployment. To increase availability and scale, you can add more servers to the cluster as needed over time. Tableau Server can be configured to scale linearly as you add server nodes to a cluster. The number of users that you can support will vary depending on your workloads and specific situations. However, in our testing, we found that Tableau Server was able to support over 5,000 users with 414 active users on a 32-core server cluster. Each of the workers was 8 cores each with the primary deployed with repository only.

If an organization of 3,000 people has about 5% of them active for an hour (peak usage time) and the business requires data to be refreshed every eight hours instead of every 24 hours as in the above scenario, it can be supported with a Tableau Server deployment of 28 cores with well-designed, fast-performing workbooks on the low end. Sixteen of these cores would be sufficient for the analytics use cases and the data freshness workloads (Backgrounders) would benefit from having the 12 remaining cores. On the height end of the range, this may need up to 40 cores supporting a mix of very fast-loading workbooks (<4 seconds) and moderately-slow workbooks (between 4-10 seconds, given the complexity of the dashboards).

As you scale up to users, the architecture supports this easily by adding more Tableau Servers to your existing deployment cluster. For example, for an organization with 5,000 people with about 5% of them active with a business goal for data freshness of less than 24 hours could be supported with up to 44 cores for the analytics services and an additional 12 cores for the data freshness services (Backgrounder) for a total of 56 cores.

While adding nodes to a cluster and tuning process instances to respond to use profiles are straightforward management tasks, complex deployments require professional IT staff. Tableau Server requires communication with data sources that may be internal to the network or in the cloud. Users may be connecting to those data sources through Tableau Server from the internet or internally on a variety of platforms. User authentication may be integrated with Active Directory or any third-party enterprise identity management suites via SAML. Therefore, planning a complex deployment of Tableau Server should include input from all senior members of your IT department.

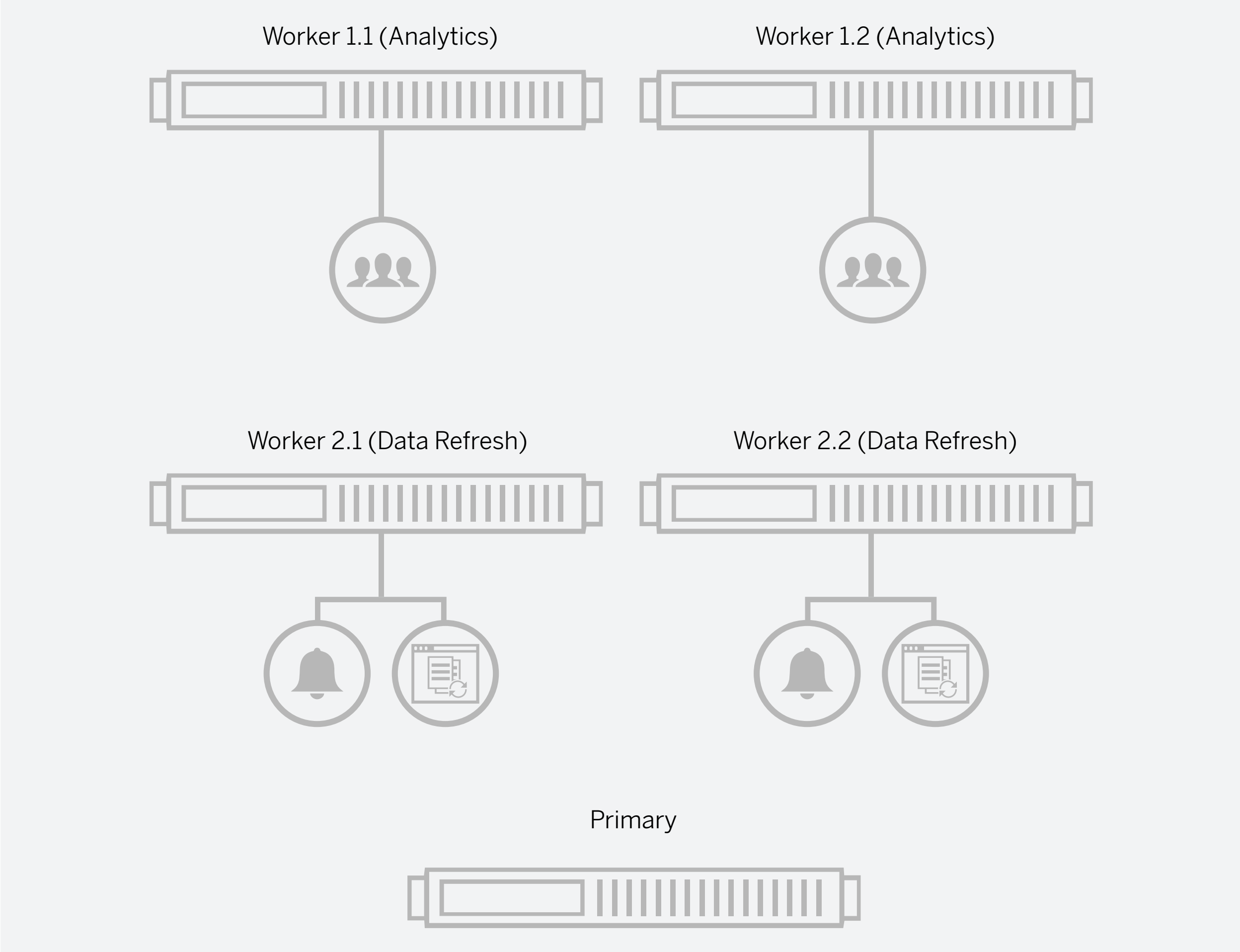


Figure 6: Multi-machine deployment

Below is an illustration of an architecture where multiple Tableau Servers can service users both internal and external to the enterprise.

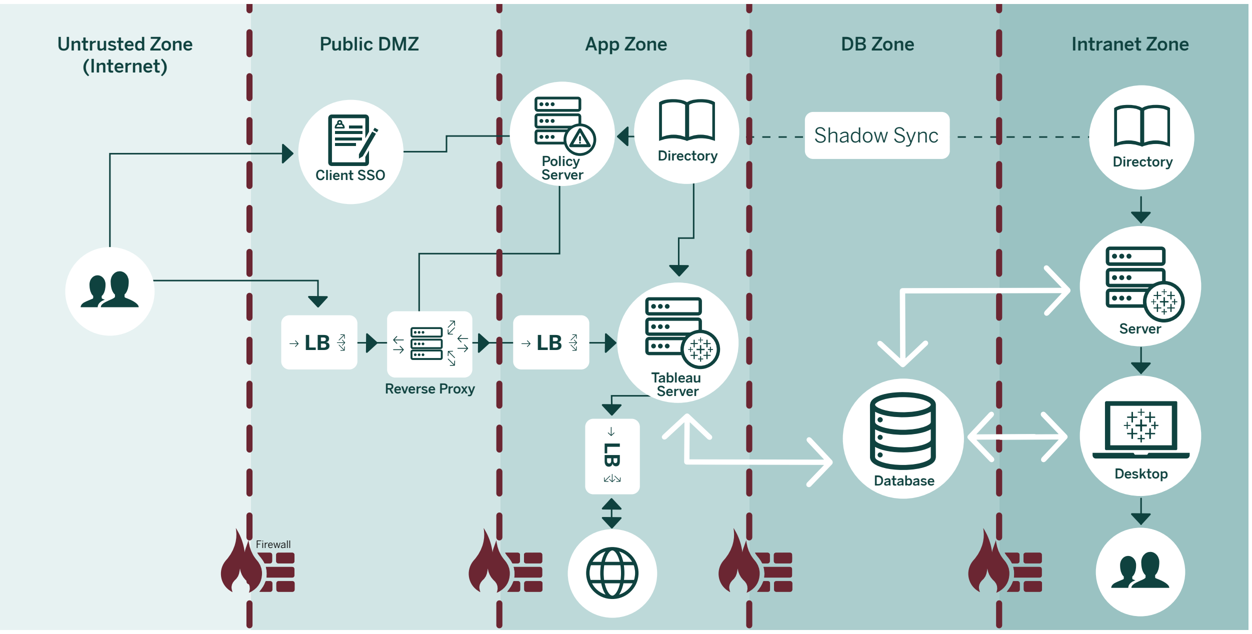


Figure 7: An overview enterprise-class deployment

Considerations

Our sizing guidance makes certain technical assumptions that may or may not reflect your own circumstances. We use a base-level assumption that about 10% of total users are typically active on a server. In this scenario, Tableau Server can support up to 1,120 total users in an organization with an 8-core single-server deployment and up to about 4,480 total users on a 32-core multi-node deployment. Based on varying workloads described in the previous section, a single 8-core server can support a range of between 50 to 112 active users. Using the same 10% concurrency rate, this means a range of 500 to 1120 total users on a single 8-core system, reflected in the first column of Figure 9.

sůňŊſƃĐķůňŝůŖɣ~ķſŝķŁŝŰŝĐɳ

We often say that in your real world performance, you may see variations from what we publish based on our tests. There are many factors that influence the variations you may see due to the nature of a self-service analytics environment.

Take a very simple view of a Tableau Server system that is only loading views on behalf of end users, and there is no other workload on the system: if a single Tableau view takes 1 second to load fully, then on an 8 core system, you can expect up to 8 active users who start simultaneously to get a 1 second response time. If 9 users load the view at the same instant, the 9th user on that system will get a 2 second response time because at least one of the first 8 users needs to complete their work first. Realistically, most users rarely arrive at the same exact instant and often there are delays in human interaction with the system; think time, caching layers, query optimizations, parallelism, network latency, and other variables will have a meaningful impact on the exact number of users you can support on your system.

Furthermore, the complexity of workbooks can have a large effect on the system throughput. In our scalability testing, we used a mix of workbooks based on real production usage of the system during peak usage. Simple workbooks took between 1.5 and 4 seconds to load, while moderately complex workbooks took between 4-10 seconds to load and more complex workbooks took between 10-20 seconds.

QŝłŊůƃŝůŖ

For complex workloads that lead to single-user load times between 10-20 seconds, you can expect Tableau Server to support approximately 500 user per 8 cores without meaningful performance degradation. In our experiments, we pushed the CPU to 80% capacity. Assuming consistent workload, you can expect end user response times to increase with additional users. With lighter workloads, the system can handle more users. For planning purposes, we recommend a starting assumption of 500 users per 8 cores. You can then adjust based on observations of user workloads.

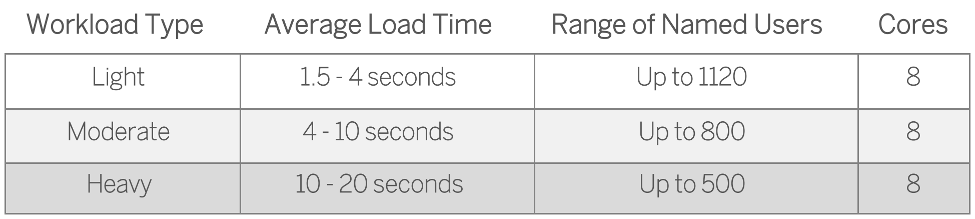
For enabling increased loads, Tableau architecture offers flexibility to isolate workloads and/or run the cluster in a highly available topology to mitigate business risks. An example of workload isolation is captured in the second column of Figure 9, where the repository was isolated to its own hardware. While this is optional, in some cases you may see improved scalability by having various Tableau workloads isolated to their own hardware.

Results summary

Our sizing guidance makes certain technical assumptions that may or may not reflect your own circumstances. We use a base-level assumption that about 10% of total users are typically active on a server. In this scenario, Tableau Server can support up to 1,120 total users in an organization with an 8-core single-server deployment and up to about 4,480 total users on a 32-core multi-node deployment. Based on varying workloads described in the previous section, a single 8-core server can support a range of between 50 to 112 active users. Using the same 10% concurrency rate, this means a range of 500 to 1120 total users on a single 8-core system, reflected in the first column of Figure 9.

For complex workloads that lead to single-user load times between 10-20 seconds, you can expect Tableau Server to support approximately 500 user per 8 cores without meaningful performance degradation. In our experiments, we pushed the CPU to 80% capacity. Assuming consistent workload, you can expect end user response times to increase with additional users. With lighter workloads, the system can handle more users. For planning purposes, we recommend a starting assumption of 500 users per 8 cores. You can then adjust based on observations of user workloads.

For a single 8-core system the following figure shows the observations from our test data in terms of the number of users the system could support.



Workload Type	Average Load Time	Range of Named Users	Cores
Light	1.5 - 4 seconds	Up to 1120	8
Moderate	4 - 10 seconds	Up to 800	8
Heavy	10 - 20 seconds	Up to 500	8

Figure 8: Scalability of an 8-core Tableau Server based on workload type

For enabling increased loads, Tableau architecture offers flexibility to isolate workloads and/or run the cluster in a highly available topology to mitigate business risks. An example of workload isolation is captured in the second column of Figure 9, where the repository was isolated to its own hardware. While this is optional, in some cases you may see improved scalability by having various Tableau workloads isolated to their own hardware.

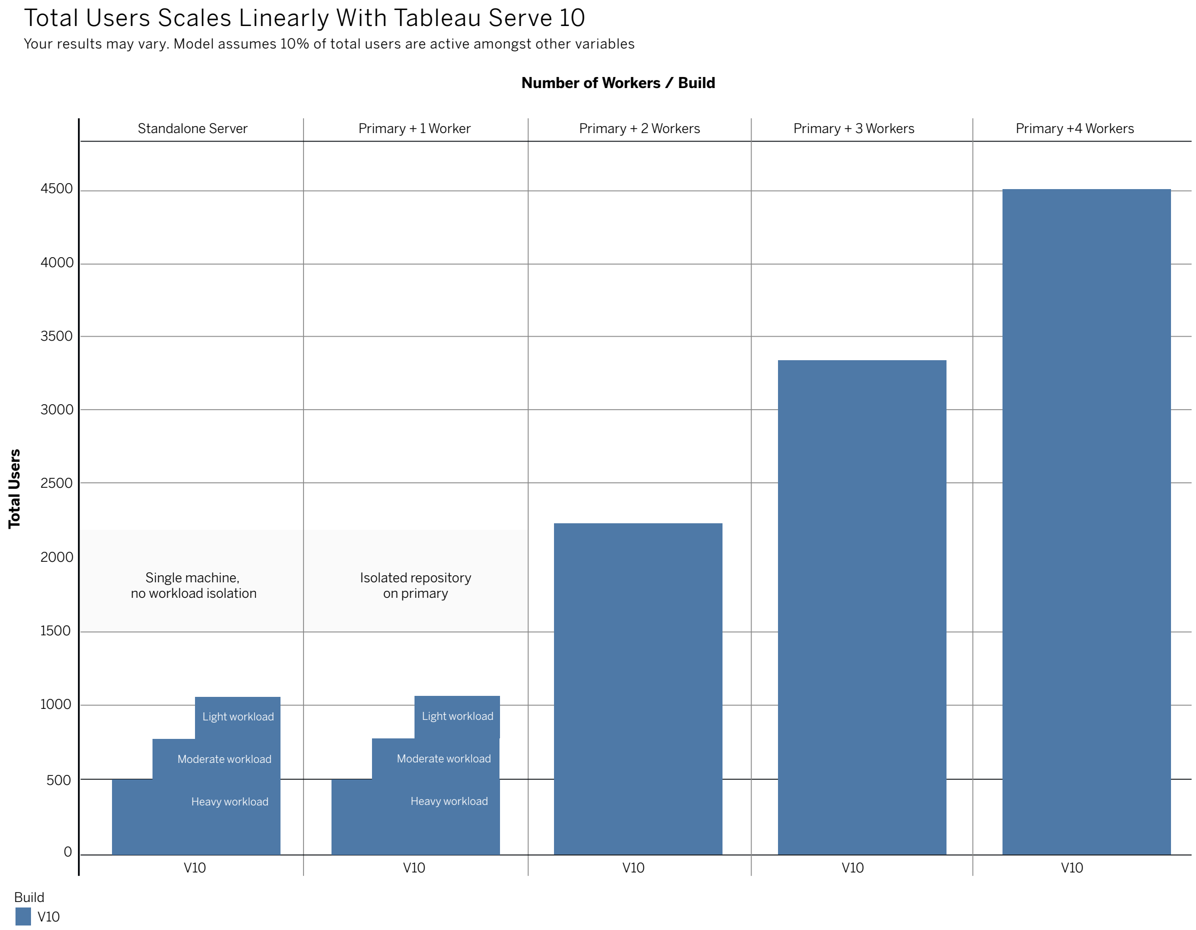


Figure 9: Tableau Server scalability (total users) scales linearly with more workers.

Conclusion

A true culture of analytics can only be achieved when self-service and governance are scaled across the organization. Tableau Server has the flexibility to grow with your business needs, but it is important to properly assess business requirements - such as user adoption, data location, data freshness, and tolerance for downtime - in order to set up a reliable environment for your users.

This paper examines factors that affect scalability of Tableau Server and reviews results based on our internal testing. Often times, enterprise deployments can vary in size and complexity. When deployed properly, Tableau Server meets enterprise demands for scalability, availability, and manageability on your infrastructure of choice.

About Tableau

Tableau helps people transform data into actionable insights that make an impact. Easily connect to data stored anywhere, in any format. Quickly perform ad hoc analyses that reveal hidden opportunities. Drag and drop to create interactive dashboards with advanced visual analytics. Then share across your organization and empower teammates to explore their perspective on data. From global enterprises to early-stage startups and small businesses, people everywhere use Tableau's analytics platform to see and understand their data.

Resources

Tableau Server Scalability: A Technical Deployment Guide for Server Administrators

Tableau for the Enterprise: An IT overview

How to Build a Culture of Analytics

Tableau on Amazon Web Services