

Escalabilidad de Tableau Server

Guía de implementación técnica para administradores de servidor

Neelesh Kamkolkar

Administrador de productos, equipo de datos y rendimiento

Índice

Resumen ejecutivo	3
VizQL revoluciona un paradigma antiguo	5
Arquitectura de Tableau	7
En memoria y en vivo: arquitectura unificada.....	8
Metodología y método de pruebas.....	9
Metodología.....	9
Características de una carga de trabajo real	11
Pasos para el modelado de prueba	11
Metodología para los procesos en segundo plano	12
Entorno aislado estandarizado	13
Topología de la implementación	14
Mediciones e informes	15
Escenario.....	15
Tiempo de respuesta.....	16
Capacidad de procesamiento de escenarios.....	16
Usuarios activos.....	17
Resultados.....	18
Escalabilidad lineal de Tableau Server 10	18
Resultados de los procesos en segundo plano.....	22
Aislamiento del proceso en segundo plano	26
Consideraciones sobre los procesos en segundo plano.....	28
Prácticas recomendadas: pruebas de escalabilidad personalizadas	28
Prácticas recomendadas para la optimización en el mundo real.....	29
Resumen	30

Resumen ejecutivo

Para muchas organizaciones, Tableau es una herramienta fundamental. A medida que un número creciente de departamentos de estas compañías descubre lo valioso que es extraer información de sus datos, los equipos de TI trabajan con la empresa para ofrecer Tableau como la plataforma de análisis para toda la organización. Como las empresas están comenzando a implementar Tableau de esta manera, es fundamental que los arquitectos empresariales y los líderes de TI comprendan cómo Tableau Server se va adaptando al crecimiento de los datos, el contenido y los usuarios. Además, a fin de dar soporte a las necesidades analíticas actuales y futuras del negocio, también deben conocer cómo se puede implementar e integrar en plataformas de TI empresariales diversas y heterogéneas.

Este informe está dirigido a los arquitectos empresariales y a los líderes técnicos de TI. En él se proporciona un panorama detallado de la arquitectura de Tableau y la manera en que se adapta a medida que aumentan las cargas de trabajo.

Hemos explorado y comprobado la escalabilidad de Tableau Server 10, y hemos visto sus resultados en comparación con versiones anteriores del producto. En respuesta a sus solicitudes, decidimos ampliar el alcance y realizar pruebas de escalabilidad de las cargas de trabajo en segundo plano y de los usuarios.

Ciertos factores del sistema pueden tener un impacto en el rendimiento y la escalabilidad de Tableau Server. Entre esas variables del sistema, se incluyen algunas importantes, como el diseño de los libros de trabajo, la configuración del servidor, el ajuste de la infraestructura, el entorno de datos, la capacidad de procesamiento y las redes. Estos factores varían enormemente entre los diferentes perfiles de uso e implementaciones. Cuando se ajusten o cambien estas variables, los resultados de una prueba determinada cambiarán. Con el propósito de mitigar y aislar estas variables, realizamos las pruebas en equipos físicos en un laboratorio de red cerrada. Nuestro objetivo era minimizar la variabilidad de las mediciones como consecuencia de la influencia de sistemas externos. Con eso bajo control, podríamos medir las métricas de escalabilidad a través del modelado del uso real de Tableau Server.

Con esta finalidad, comenzamos con el análisis de una implementación de Tableau Server en un entorno de producción real a capacidad máxima. Luego, modelamos el uso en pruebas automatizadas. Este método de dos pasos imita una carga de trabajo muy realista. También permite simular divergencias realistas en la manera en que los usuarios reales y las cargas de trabajo en segundo plano (principalmente, las extracciones de datos y las notificaciones de usuarios) pueden poner en ejecución el sistema. En este contexto, las divergencias incluyen el tiempo que los usuarios esperan entre interacciones con la visualización o la cantidad de tareas en segundo plano que se ejecutan según un programa.

Modelamos las divergencias como parte de nuestras pruebas para imitar las condiciones reales de producción. Observamos que el sistema se escala linealmente cuando agregamos más nodos de trabajador a un clúster de Tableau Server. Nuestros experimentos hicieron que el servidor trabajase en condiciones de máxima capacidad, superiores a las que observamos en entornos de producción. El uso en entornos de producción tiende a experimentar picos de capacidad en períodos cortos (ráfagas) pero frecuentes a lo largo del día de trabajo. Las mediciones de carga se expresan en unidades de capacidad de procesamiento. La capacidad de procesamiento es la cantidad de trabajo que un servidor procesa en un período de tiempo dado. Dicho de forma más simple, son las transacciones por segundo. Como se muestra a continuación, en nuestros experimentos observamos que la capacidad de procesamiento aumentó de 4 a 18 transacciones por segundo. Cada barra representa un experimento, realizado según la topología descrita en el encabezado de columna (servidor independiente, servidor principal más un nodo de trabajador, etc.).

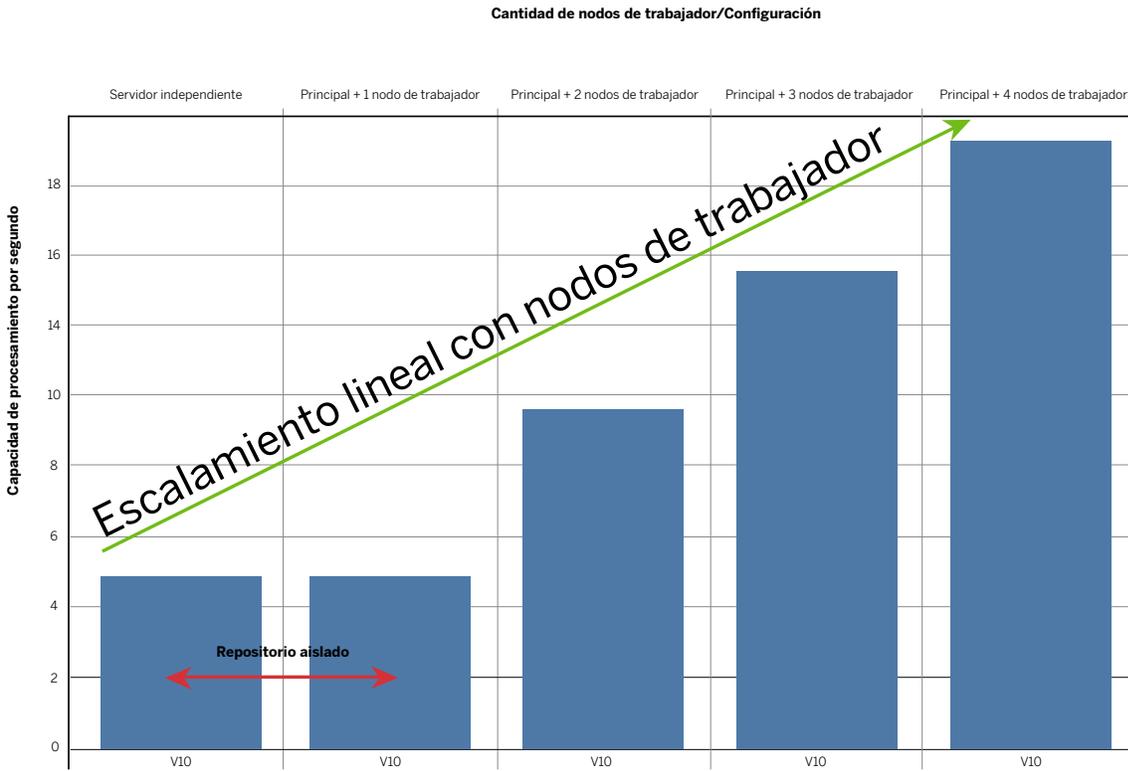


Figura 1: Capacidad de procesamiento de escenarios por segundo

Identificamos la cantidad máxima de usuarios activos que podíamos procesar a través de un servidor único con 8 núcleos físicos. La carga de usuarios activos es un conjunto automatizado de operaciones que se puede ejecutar en una instancia de Tableau Server. En una sección posterior de este informe, describimos en detalle qué operaciones se incluyen en una carga de usuarios activos. Tras medir la cantidad máxima de usuarios activos que puede admitir una instancia de Tableau Server con 8 núcleos físicos, utilizamos esa cantidad de usuarios activos como unidad de escalamiento para comprobar si se puede incrementar linealmente la carga agregando al clúster nodos de trabajador idénticos de 8 núcleos.

Como resultado, observamos que un servidor independiente de 8 núcleos puede admitir hasta 112 usuarios activos durante picos de carga realistas que modelamos de manera continua. Cuando aumentamos ese punto de referencia de manera lineal (agregando cuatro nodos de trabajador idénticos con 8 núcleos cada uno y un nodo de controlador principal con repositorio y una simple instalación base), fuimos capaces de dar soporte a 448 usuarios activos. Son usuarios que realizan tareas de manera simultánea en el sistema, por lo que si se extrapolan esos resultados teniendo en cuenta que no todos los usuarios están usando el sistema todo el tiempo, es posible proporcionar soporte entre 500 y 10 000 usuarios con una configuración de clústeres de entre 8 y 32 núcleos. Como veremos, dicha extrapolación depende de diversos parámetros. Sin modificar el resto de las variables, el tamaño del servidor, en el contexto de la cantidad de usuarios, dependerá del uso que se haga del análisis y de las condiciones de los datos. Lo que nuestros experimentos demostraron es que la arquitectura de Tableau le permitirá continuar aumentando de manera lineal su base de usuarios agregando más nodos de trabajador al clúster de Tableau Server.

En nuestras pruebas, llevamos el servidor al límite para identificar el punto en el que comenzaba a dar errores o dejaba de estar disponible. Estas pruebas de esfuerzo proporcionaron un punto máximo

que define los límites de escalabilidad y rendimiento del servidor. Una vez cuantificados estos límites, vamos más allá de las típicas cargas ráfaga de producción hasta llegar a cargas pico continuas. Por lo tanto, en la práctica, donde el estándar es un estado más estable, la plataforma debería admitir una mayor capacidad de uso que la sugerida en este informe. Puede estar seguro de que la arquitectura de Tableau Server es lo suficientemente escalable como para satisfacer sus necesidades actuales y cubrir su crecimiento futuro en forma lineal.

En la tabla a continuación, se describen cada una de las topologías y casos de uso. En la columna Perfil de riesgo, se incluyen variables como situaciones en las que la topología puede estar expuesta a mayor o menor riesgo según diversos factores, como la conmutación tras error automática, la exposición a fallos en el hardware y la capacidad de aumento disponible durante las ráfagas pico.

Configuración de la implementación	Caso de uso	Perfil de riesgo
Servidor independiente	Implementación simple de servidor único	Alto riesgo para la disponibilidad, alto nivel de disputa por los recursos, sin capacidad disponible durante períodos desafiantes para el rendimiento de carga pico
1 principal + 1 nodo de trabajador	Implementaciones de dos nodos	Alto riesgo para la disponibilidad, menor disputa por los recursos con el repositorio, sin capacidad disponible durante períodos de uso pico
1 principal + 2 nodos de trabajador	Implementaciones de tres nodos	Riesgo moderado para la disponibilidad, menor disputa por los recursos, escalabilidad horizontal mejorada, muy poca capacidad disponible durante períodos de uso pico
1 principal + 3 nodos de trabajador	Implementaciones de cuatro nodos	Riesgo moderado para la disponibilidad, menor disputa por los recursos, escalabilidad mejorada, capacidad disponible moderada durante períodos de uso pico
1 principal + 4 nodos de trabajador	Implementaciones de cinco nodos	Menor riesgo para la disponibilidad, escalabilidad mejorada, mayor capacidad disponible durante períodos de uso pico ¹

Tabla 1: Topologías y usos de caso para implementaciones

¹La capacidad de aumento disponible depende de muchos factores y no se puede garantizar

Es importante comprender estas cifras en el contexto de la metodología y las pruebas. Con este propósito, en el resto de este informe nos concentraremos en describir la arquitectura de Tableau y lo que la hace única en comparación con otras tecnologías de BI tradicionales. Además, expondremos nuestra metodología y el análisis de los resultados de escalabilidad.

VizQL revoluciona un paradigma antiguo

Si está acostumbrado a las soluciones tradicionales de inteligencia de negocios (BI) o Tableau es nuevo

para usted, puede serle de ayuda comprender algunas diferencias básicas del funcionamiento de Tableau en comparación con la BI tradicional. Derrocamos la premisa fundamental del método de BI tradicional, en el que se resuelve la consulta primero para después generar la visualización. En su lugar, creemos que los datos desvelan conocimientos a medida que se los explora y se realizan consultas en tiempo real. Durante más de una década, Tableau viene trabajando con una tecnología patentada, denominada VizQL™, que combina las consultas y la visualización en una sola plataforma. Esta eficaz unión de funcionalidades permite a los usuarios finales hacer preguntas ilimitadas a sus datos. Pueden hacer consultas, filtrar y analizar, todo de manera simultánea a la visualización de sus datos. VizQL™ es un lenguaje fundacional de Tableau que asigna expresiones a las acciones y preguntas de los usuarios, y las traduce en consultas que pueden ejecutarse en cualquier conjunto de datos tanto en las instalaciones físicas de la empresa como en la nube. Durante más de una década de inversión en ingeniería, esta tecnología maduró y continúa evolucionando con el fin de adaptarse a la próxima generación de fuentes de datos y requisitos de análisis.

A diferencia de los informes de inteligencia de negocios tradicionales, que se diseñan y desarrollan con un conjunto estático y predefinido de requisitos, las visualizaciones de Tableau están diseñadas para la interacción y la colaboración. Los usuarios pueden hacer preguntas a sus datos sin necesidad de programar consultas SQL ni uniones complejas. Para conseguir las respuestas a sus preguntas, no es necesario que los usuarios esperen hasta el próximo ciclo del proceso de desarrollo de software tradicional. En lugar de eso, pueden iterar en las visualizaciones existentes y continuar sus análisis para responder preguntas acerca de sus negocios o proyectos.

Con la BI tradicional, es posible que esté acostumbrado a realizar pruebas de carga de informes estáticos que cumplen con un acuerdo de nivel de servicios (SLA) específico, en los que las consultas se diseñaron para ejecutarse en sistemas particulares con optimizaciones pregeneradas. Un informe estático tiene un alcance fijo formado por un conjunto fijo de consultas optimizadas por un desarrollador, una por una, a lo largo de varias semanas. Aunque es relativamente sencillo definir y configurar un SLA para un informe estático, las modificaciones en el informe reinician el ciclo de desarrollo completo. Esta limitación habitualmente no está contemplada en el SLA.

Las visualizaciones de Tableau, por otra parte, regeneran y envían nuevas consultas en nombre de las acciones exploratorias del usuario. El almacenamiento en caché del navegador local y las optimizaciones en VizQL, que permiten la rápida recuperación de datos, pueden ayudar al usuario a permanecer en el flujo de análisis en vez de esperar los resultados de una consulta.

El lenguaje de VizQL está incluido en nuestro proceso de servidor VizQL y funciona en consonancia con muchos otros procesos de servidor distribuidos, a fin de proporcionar una plataforma de análisis de negocios escalable, disponible, segura y confiable.

Arquitectura de Tableau

La arquitectura flexible de Tableau está diseñada para escalar en vertical y en horizontal. Permite ejecutar Tableau como una plataforma empresarial estándar o como una que impulsa el análisis en la nube. Tableau Server es capaz de dar soporte tanto a los requisitos de infraestructura de producción empresarial más complejos, como a implementaciones sencillas en un departamento o grupo de trabajo.

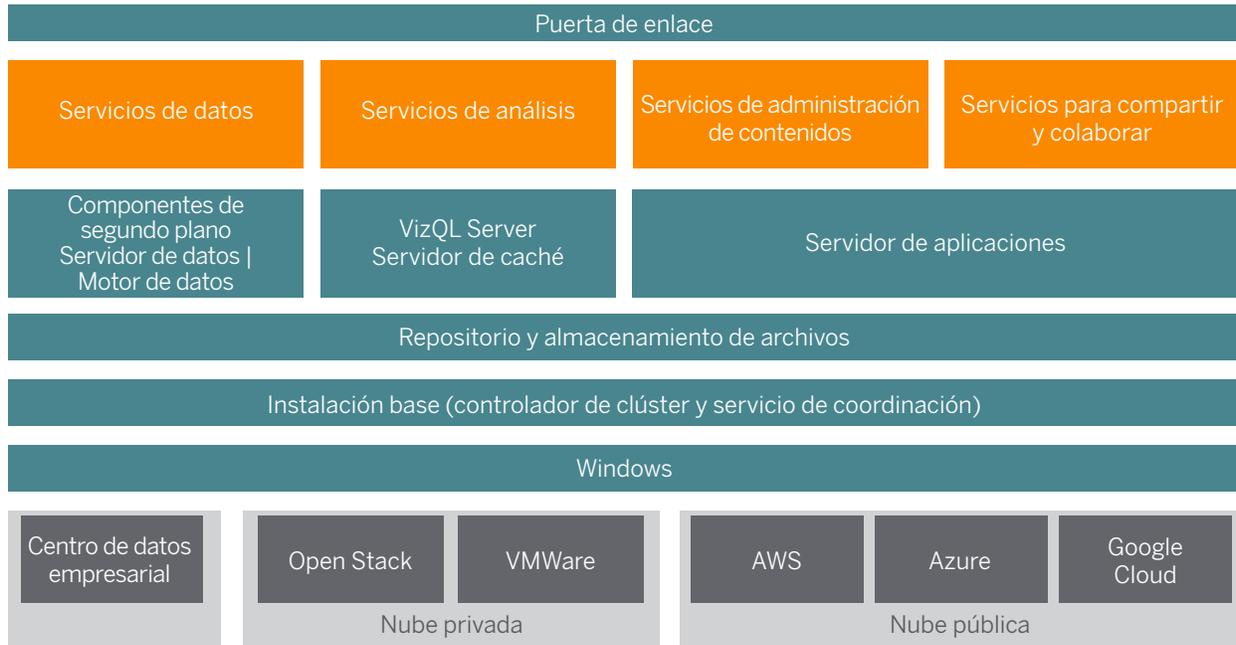


Figura 2: Arquitectura de Tableau Server

Tableau Server cuenta con un proceso de instalación y configuración simple. Una vez instalado, numerosos procesos de servidor (en azul en la Figura 2) trabajan conjuntamente para proporcionar servicios en diferentes niveles.

El proceso de puerta de enlace es el componente que redirecciona el tráfico proveniente de todos los clientes de Tableau hacia los nodos de servidor disponibles en un clúster.

Los servicios de datos son una agrupación lógica de servicios que mantienen los datos actualizados, administran los metadatos de manera compartida, y controlan las fuentes de datos y los datos en memoria. Los procesos subyacentes que forman parte de los servicios de datos son los procesos en segundo plano, de servidor de datos y de motor de datos.

Los servicios de análisis, compuestos por los procesos de VizQL y de servidor de caché, proporcionan las visualizaciones orientadas al usuario y los servicios de análisis y de almacenamiento en caché.

El proceso de servidor de aplicaciones controla la administración de contenido y el servicio para compartir y colaborar. Este proceso también proporciona las funcionalidades principales de Tableau Server, como el inicio de sesión de los usuarios, la administración del contenido (proyectos, sitios, permisos de seguridad, etc.) y las actividades de administración.

Todos los servicios mencionados anteriormente utilizan el proceso de repositorio y dependen de él. Este proceso contiene datos relacionales estructurados, como permisos, libros de trabajo, extracciones de datos, información de usuarios y metadatos. El proceso de almacenamiento de archivos permite generar redundancias en los archivos de extracciones de datos en todo el clúster y garantiza que las extracciones estén disponibles a nivel local en todos los nodos del clúster. Durante los períodos de cargas altas, los archivos de extracción están disponibles localmente para todo el clúster a fin de asegurar mayor velocidad de procesamiento y representación.

La arquitectura de Tableau es flexible, lo que le permite ejecutar la plataforma en cualquier lugar. Puede instalar Tableau Server en las instalaciones físicas, tanto en una nube privada como en un centro de datos, en Amazon EC2™, en Google Cloud Platform™ o en Microsoft Azure™. La plataforma de análisis de Tableau también se puede ejecutar en plataformas de visualización, como VMware ESXi™ o Microsoft Hyper-V™. A fin de garantizar el máximo rendimiento de Tableau Server, le recomendamos seguir las prácticas recomendadas para cada plataforma de visualización.

Para obtener detalles sobre procesos de servidor individuales, revise la [guía de administración](#) de Tableau Server.

En memoria y en vivo: arquitectura unificada

Tableau es compatible con una heterogénea variedad de plataformas, lo que le permite conectarse a más de 50 de las fuentes de datos de mayor popularidad, provistas por decenas de proveedores. La arquitectura cuenta con la flexibilidad necesaria como para que pueda elegir si desea realizar análisis rápidos en memoria o impulsar el análisis a través de una conexión directa con almacenes de datos en vivo. Independientemente de lo que elija, también es sencillo pasar de un análisis con conexión en tiempo real a uno en memoria, para satisfacer las diversas necesidades del negocio o adaptarse a distintas fuentes de datos.

Tableau admite el análisis en memoria a través de la extracción de datos y su posterior acopio en un almacén de columnas. Este tipo de archivo de propiedad exclusiva denominado Extracción de datos de Tableau (TDE) se carga en archivos asignados en memoria para un acceso rápido. Las extracciones se pueden crear mediante una instancia de Tableau Desktop o a través de procesos de negocios externos que utilicen las API de Tableau. La arquitectura de Tableau Server proporciona soporte incorporado para garantizar que las extracciones creadas por sus usuarios se actualicen según un programa optimizado que usted puede controlar.

Independientemente de cómo se conecte a los datos y los incorpore a Tableau Server, debe asegurarse de tener los recursos de memoria suficientes para los análisis de datos, el almacenamiento en caché, las actualizaciones de extracciones y otras operaciones relacionadas.

A diferencia de las herramientas que solo trabajan en memoria, la superficie de memoria de Tableau abarca el clúster completo y no se acumula en un solo servidor. Por su parte, el uso de la memoria se distribuye entre todo el clúster a través de servidores de caché, para el almacenamiento en caché de consultas externas, y de otros procesos que almacenan en caché en el nivel de la sesión. De esta manera, el impacto en memoria del aumento de cargas se distribuye entre todo el clúster, de acuerdo con sus cargas de trabajo.

Metodología y método de pruebas

A diferencia de la mayoría de los proyectos tradicionales de pruebas de carga, en los que la aplicación evaluada se considera una caja negra, las pruebas de cargas de Tableau Server se deben realizar con un entendimiento suficiente de su arquitectura. Desarrollamos Tableau Server para que se pueda ejecutar en cualquier lado: en las instalaciones físicas o en la nube. Está diseñado para funcionar en equipos y organizaciones de todos los tamaños. Por lo tanto, la instalación predeterminada de Tableau Server debería cubrir las necesidades de la mayoría de las implementaciones estándar. Cuando desee comenzar a escalar e implementar Tableau Server en toda la empresa, debe comprender cómo se procesan las diferentes cargas de trabajo y de qué manera unos pocos y simples ajustes de configuración pueden mejorar los resultados para su caso de implementación.

Para fijar los objetivos de escalabilidad de la versión 10, decidimos contestar las siguientes preguntas:

1. ¿Tableau Server 10 escala de manera lineal cuando se agrega más hardware en las siguientes dos situaciones comunes?
 - Escalabilidad de la carga de trabajo de usuarios finales
 - Escalabilidad de la carga de trabajo en segundo plano
2. ¿Cuándo es el momento correcto para pasar los procesos en segundo plano a un nodo de trabajador dedicado?
3. ¿Cómo es el rendimiento de Tableau Server 10 en comparación con las versiones anteriores?

En las versiones anteriores de este informe, nos centramos específicamente en la escalabilidad de usuarios finales, pero muchos clientes nos preguntaron acerca de la escalabilidad de la carga de trabajo en segundo plano. Las cargas de trabajo en segundo plano controlan que los datos estén actualizados (actualizaciones de extracciones) y el consumo de análisis (notificaciones de suscripciones).

Metodología

Los objetivos principales de la versión 10 de Tableau Server fueron mejorar el rendimiento y la escalabilidad. Como consecuencia, un requisito fundamental era, también, desarrollar una metodología de pruebas de clase empresarial que estuviera al nivel de la producción. Para empezar, ejecutamos la metodología descrita en esta sección en las versiones previas al lanzamiento de Tableau Server 10. Durante el proceso iterativo del desarrollo ágil, y gracias a la metodología, descubrimos y solucionamos casi dos docenas de errores específicamente relacionados con la escalabilidad y el rendimiento. Tras el lanzamiento de la versión 10, continuamos haciendo pruebas y solucionamos los errores con la versión de mantenimiento 10.0.1. En todo este informe, nos referimos en términos generales a la versión 10 de Tableau Server. Sin embargo, los resultados de las pruebas se basan en la versión de mantenimiento 10.0.1. Para obtener los mejores beneficios de escalabilidad y rendimiento de la versión 10, ejecute en su organización la versión de mantenimiento 10.0.1 o posterior.

Modificamos continuamente nuestras prácticas de escalabilidad con la incorporación de los resultados de las pruebas de cargas de trabajo que imiten situaciones realistas de nuestros clientes. Aunque hay muchas variables que afectan la escalabilidad de una implementación, los factores importantes que se deben tener en cuenta mientras planea la suya son los siguientes:

- Impacto de los usuarios (uso de autoservicio y adopción de usuarios): ¿Cuántos usuarios realizarán análisis? ¿Con qué frecuencia los realizarán para tomar decisiones informadas? ¿Qué complejidad tienen las visualizaciones que crean los usuarios?

- Impacto de los datos (nivel de actualización, tamaño y ubicación): ¿Qué tamaño tienen los datos? ¿Dónde se encuentran? ¿Qué grado de actualización deben tener para propiciar la toma de decisiones precisas?

Uso de análisis para la toma eficaz de decisiones de negocios	Alto (cada segundo)	7. Ejemplos: Exploración de datos a nivel mundial en Tableau Public (Elecciones presidenciales de los Estados Unidos) 30K vistas por hora	8. Ejemplos: Dashboard de cuotas de ventas, Tableau en TV	9. Ejemplos: Controlador de tráfico aéreo, Control de finanzas, Ejecución de operaciones
	Moderado (una vez por hora)	4. Ejemplos: Marketing de análisis de clientes de aseguradoras Inventario diario de una tienda (fijación de objetivos)	5. Ejemplos: Administración de capacidad de pacientes	6. Ejemplos: Dashboard de soporte para escalado, Dashboard de cartera de finanzas Investigación de fraudes
	Bajo (una vez al día)	1. Ejemplos: Ingeniería - Reunión de lanzamiento, Inventario de hipotecas, BI tradicional	2. Ejemplos: Registro de oportunidades de venta para negocios potenciales	3. Ejemplos: Dashboards de tráfico en sitio web de autopistas
		Bajo (una vez al día)	Moderado (una vez por hora)	Alto (cada segundo)
Frecuencia de actualización de datos para la toma eficaz de decisiones de negocios				

Figura 3: Matriz de uso de análisis y frecuencia de actualización de datos

Para realizar pruebas teniendo en cuenta ambas consideraciones, debimos incorporar pruebas en las que un servidor se cargaba de manera incremental con más usuarios finales al mismo tiempo que se aumentaban las cargas de trabajo en segundo plano. Esta metodología nos permitió estudiar el impacto de las cargas de trabajo en segundo plano en la calidad de servicio a los usuarios finales. A fin de modelar usos de análisis bajos, moderados y altos, y el impacto de aislar las cargas de trabajo en segundo plano, ejecutamos más de 400 iteraciones de pruebas en nuestros laboratorios con diferentes cargas de trabajo. Estudiamos la escalabilidad del sistema y solucionamos errores que solo se manifestaron durante las cargas altas.

Con el propósito de que estas pruebas fueran pertinentes en el mundo real, primero tuvimos que identificar conjuntos de cargas de trabajo adecuados para ejecutar las pruebas.

Características de una carga de trabajo real

Para la versión 10, analizamos y caracterizamos la forma en que se utilizaba una instancia de Tableau Server en un entorno de producción real durante períodos de utilización máxima.

A fin de determinar qué libros de trabajo debíamos modelar y cuáles eran las características de la carga de trabajo para nuestras pruebas, revisamos los archivos de registro de Tableau Server provenientes de un entorno de producción de 3000 usuarios. Identificamos las visualizaciones y los libros de trabajo que se utilizaban con mayor frecuencia. Calculamos la distribución de uso entre estos libros de trabajo y, luego, analizamos las características de uso. Por ejemplo, investigamos el intervalo de tiempo entre solicitudes (también conocido como “tiempo de reflexión de usuario”). A continuación, se indican los pasos específicos que seguimos para modelar la carga de trabajo a partir del servidor de producción.

Pasos para el modelado de prueba

1. Obtener los registros del servidor de producción, generados durante los períodos de uso pico.
2. Identificar las N vistas principales de los libros de trabajo según el tiempo promedio ponderado registrado en el servidor.
3. Promedio ponderado = Tiempo de respuesta promedio * Número de solicitudes
4. Calcular las ponderaciones relativas entre las N vistas principales de los libros de trabajo.
5. Para cada vista de libro de trabajo seleccionada:
 - a. Calcular el porcentaje de cargas de visualización con: actualización = y (esta era una manera de averiguar cuántos usuarios actualizaban activamente los datos).
 - b. Calcular las N interacciones principales por ponderación.
 - c. Calcular el promedio de tiempo de reflexión de usuario entre interacciones.
6. Para la verificación del modelo:
 - a. Calcular el tiempo promedio entre arranques (períodos entre pruebas) para la vista del libro de trabajo.
 - b. Identificar las N tareas principales en segundo plano por tiempo promedio ponderado registrado en el servidor.
7. Para la caracterización del tráfico en segundo plano según las diversas tareas (por ejemplo, suscripciones y actualizaciones de extracciones):
 - a. Calcular el tiempo promedio entre las principales tareas en segundo plano e incluirlo en el modelo.
 - b. Identificar la cantidad de suscripciones en los diferentes programas.
 - c. Identificar el tamaño y tipo de las actualizaciones de extracciones (extracciones publicadas en el servidor de datos, extracciones de libros de trabajo, etc.).

Utilizamos todos los datos antes mencionados para modelar una combinación realista de cargas de trabajo que imitara el uso real que los usuarios le dan al servidor de producción durante períodos de uso pico. Por último, generamos un modelo basado en cargas de trabajo para las suscripciones y actualizaciones de extracciones en segundo plano a partir del análisis de los registros de producción.

Las combinaciones de cargas de trabajo se detallan en la siguiente tabla.

Nombre de la carga de trabajo	Descripción	Cómo hacer la comparación con versiones anteriores
Entorno de producción real Carga de trabajo del servidor	Esta carga de trabajo se basó en el análisis y caracterización del uso de una instancia de Tableau Server en producción, que proporcionaba soporte a 3000 usuarios en una organización como aplicación esencial administrada por TI.	Results data from only this whitepaper are comparable between 9.3 and 10.0 as specified here. The results are not comparable to previous whitepapers we've published—including the 9.0 whitepaper—since testing methodology was significantly different.
Carga de trabajo de una instancia de Tableau Server en entorno de producción real + Nuevas funcionalidades	Esta carga de trabajo combina la carga anterior con libros de trabajo que ejecutan las nuevas funcionalidades de la versión 10.	No permite la comparación entre las versiones 9.3 y 10, porque la versión 9.3 no cuenta con las funcionalidades de la 10. El mejor uso que se le puede dar a este informe es como fuente de información sobre las características de escalabilidad de la versión 10.
Combinación de componentes de segundo plano	Esta carga de trabajo se basa en el análisis de cargas de trabajo de producción y modela libros de trabajo reales y programas que simulan el entorno de producción.	La nueva combinación introducida en el informe de pruebas de escalabilidad de la versión 10. No se puede comparar con los resultados de ningún informe anterior.

Tabla 2: Descripciones de las combinaciones de cargas de trabajo

A continuación, ejecutamos cada una de estas combinaciones de manera independiente en un laboratorio aislado de escalabilidad, en equipos físicos, con cargas de usuarios finales y de procesos en segundo plano cada vez mayores. Una vez saturada la capacidad máxima del clúster, continuamos aumentando la carga después de agregar un nodo de trabajador a la vez. Observamos el comportamiento del sistema durante cada uno de estos experimentos. En cada prueba, registramos los indicadores clave de rendimiento, como tiempo de respuesta, capacidad de procesamiento y tasas de errores. También llevamos un registro de las métricas de sistema y de servidor de aplicaciones con JMX. En caja ejecución correlacionamos los datos y analizamos el comportamiento del sistema ante el incremento de las cargas de trabajo. Durante este proceso, también encontramos errores de escalabilidad y los corregimos como parte de nuestro ágil proceso de desarrollo.

Metodología para los procesos en segundo plano

El servidor de procesos en segundo plano procesa las tareas en segundo plano en el nivel de sistema y usuario. El proceso en segundo plano realiza tareas en el nivel de sistema, como el mantenimiento de rutina del repositorio. Las tareas en el nivel de usuario son aquellas que un usuario puede enviar para que el sistema las ejecute en su nombre. Por ejemplo, los usuarios pueden publicar extracciones en el servidor y, luego, configurar una actualización de datos periódica para una extracción según un programa. Este conjunto de operaciones genera una tarea de actualización. El proceso en segundo plano es el proceso que revisará las listas de tareas y las ejecutará en nombre del usuario. Esta operación es fundamental para un autoservicio eficaz, porque el usuario no debe esperar que un departamento administrativo actualice los datos. Sin embargo, si el equipo de administración que supervisa Tableau Server no prevé la capacidad suficiente para este tipo de carga, se producirán errores en la calidad del servicio que afectarán al usuario

final. Tener en cuenta el grado en que deberán optimizarse los servicios de procesos en segundo plano es fundamental a la hora de planear y determinar el tamaño de servidor. Debe evaluar la posibilidad de separar los servicios en segundo plano en otro equipo a fin de aislar las cargas de trabajo.

Hay varias prácticas recomendadas simples, detalladas al final de este informe, que le permitirán separar el procesamiento de las cargas de trabajo según el momento del día. Sin embargo, si el proceso en segundo plano se ejecuta en el mismo equipo que los servicios de análisis, es posible que los usuarios finales perciban un deterioro de la calidad del servicio durante cargas altas, debido a límites en la capacidad de compartir recursos en el servidor.

Por esta razón, nos propusimos estudiar cómo afecta el proceso en segundo plano a la escalabilidad de usuarios finales cuando se ejecuta en el mismo equipo que los servicios de análisis. Además, queríamos cuantificar la escalabilidad del proceso en segundo plano con cargas en aumento cuando se lo aislaba en su propio hardware.

Para estudiarlo, utilizamos un equipo con cuatro núcleos físicos para ejecutar el proceso en segundo plano de manera aislada. No se ejecutó ningún otro proceso de Tableau Server en el mismo equipo. Probamos las mismas cargas de trabajo, modeladas según la producción, en las versiones 9.3 y 10.0 de Tableau Server. En la carga de trabajo se incluyeron actualizaciones de extracciones y suscripciones para que nos pudiésemos centrar en las tareas en el nivel de usuario. La carga de trabajo contenía 400 suscripciones provenientes de 8 programas. Estudiamos el éxito de las notificaciones de suscripción, además de la cantidad de tiempo que Tableau tardó en completar todas las suscripciones.

Entorno aislado estandarizado

Realizamos las pruebas de escalabilidad en nuestro laboratorio de rendimiento en máquinas físicas con las siguientes especificaciones.

Tipo de servidor	Dell PowerEdge R620
Sistema operativo	Microsoft Windows Server 2012 R2 Standard de 64 bits
CPU	2,6 GHz, 1x8 núcleos físicos, Hyper-Threading habilitado (16 núcleos lógicos)
Memoria	64 GB

Tabla 3: Especificaciones de hardware para cada nodo del entorno de prueba

Aunque esta tabla enumera núcleos físicos, le recomendamos que no deshabilite la tecnología Hyper-Threading. Para lograr homogeneidad, con la excepción de esta mención, en este informe nos referimos a cantidad de núcleos físicos y siempre asumimos que la tecnología Hyper-Threading está habilitada en los equipos físicos.

Topología de la implementación

Los clústeres se componen de uno o más nodos principales (controladores) y uno o más nodos de trabajador. En nuestras pruebas, los nodos de trabajador compartían el mismo perfil de configuración de procesos:

Proceso	Principal tsperf-212.perf.dev.tsi.lan	Nodo de trabajador 1 tsperf-213.perf.dev.tsi.lan	Nodo de trabajador 2 tsperf-215.perf.dev.tsi.lan	Nodo de trabajador 3 tsperf-216.perf.dev.tsi.lan	Nodo de trabajador 4 tsperf-217.perf.dev.tsi.lan
Controlador de clúster	✓	✓	✓	✓	✓
Puerta de enlace	✓	✓	✓	✓	✓
Servidor de aplicaciones		✓	✓	✓	✓
VizQL Server		✓ ✓	✓ ✓	✓ ✓	✓ ✓
Servidor de caché		✓ ✓	✓ ✓	✓ ✓	✓ ✓
Buscar y examinar		✓	✓	✓	✓
Componente de segundo plano		✓	✓	✓	✓
Servidor de datos		✓ ✓	✓ ✓	✓ ✓	✓ ✓
Motor de datos		✓	✓	✓	✓
Almacenamiento de archivos		 Sincronizando	 Sincronizando	 Sincronizando	 Sincronizando
Repositorio	✓				
<input type="button" value="Estado de actualización"/>  Activo  Ocupado  Pasivo  Sin licencia  Fuera de servicio <input type="checkbox"/> Estado no disponible 					

Figura 4: Configuración de procesos

La configuración de procesos de nodos de trabajador que se muestra aquí es la predeterminada. Es posible que la escalabilidad varíe según la cantidad y los tipos de procesos que configure para su entorno y situación de uso.

Los nodos principales se configuran con una instalación base, que incluye los procesos de controlador de clúster, puerta de enlace y repositorio. Es importante tener en cuenta que, cuando se implementa en un esquema de licencias basadas en núcleos, este tipo de configuración de nodo principal no se incluye en el conteo de núcleos.

Aumentamos la carga de trabajo mediante generadores de carga (TabJolt) para simular la carga de trabajo de usuarios descrita anteriormente. **TabJolt** es una herramienta para pruebas de rendimiento y carga diseñada para trabajar fácilmente con Tableau Server 9.0 o posterior. En la siguiente figura se muestra una vista lógica de la ejecución de la prueba.

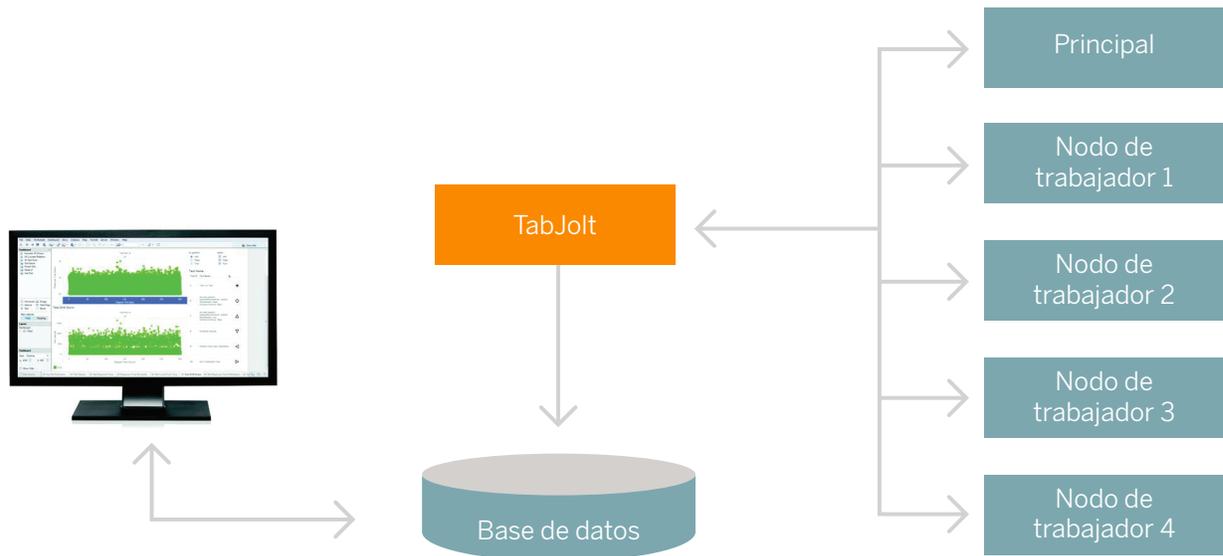


Figura 5: La vista lógica del entorno de prueba

Recopilamos datos de cada iteración de prueba para analizarlos. Pero, antes de revisar los resultados, es recomendable entender primero algunas de las métricas y sus definiciones.

Mediciones e informes

A fin de comprender el rendimiento y la escalabilidad del hardware, llevamos un registro de una serie de métricas, entre las que se incluyen métricas de sistema para CPU, memoria y disco. También medimos el rendimiento y la escalabilidad a través de métricas como los tiempos de respuesta, la capacidad de procesamiento, las tasas de errores y la duración de ejecución, entre otras.

Para comprender los datos que se analizan en este informe, revisemos rápidamente algunas definiciones.

Escenario

El escenario es la actividad del usuario de nivel superior en el servidor. En las versiones anteriores de este informe, nos centramos en la carga de visualización y los tiempos de interacción cuando el servidor tenía el acceso de invitado habilitado. En esta versión, decidimos expandir las cargas de trabajo a fin de incluir los servicios de servidor de aplicaciones (como el inicio de sesión) y otros servicios. Los usuarios finales siguen una serie de pasos basados en el modelado de las cargas de trabajo de producción, que nosotros simulamos mediante una versión personalizada de TabJolt.

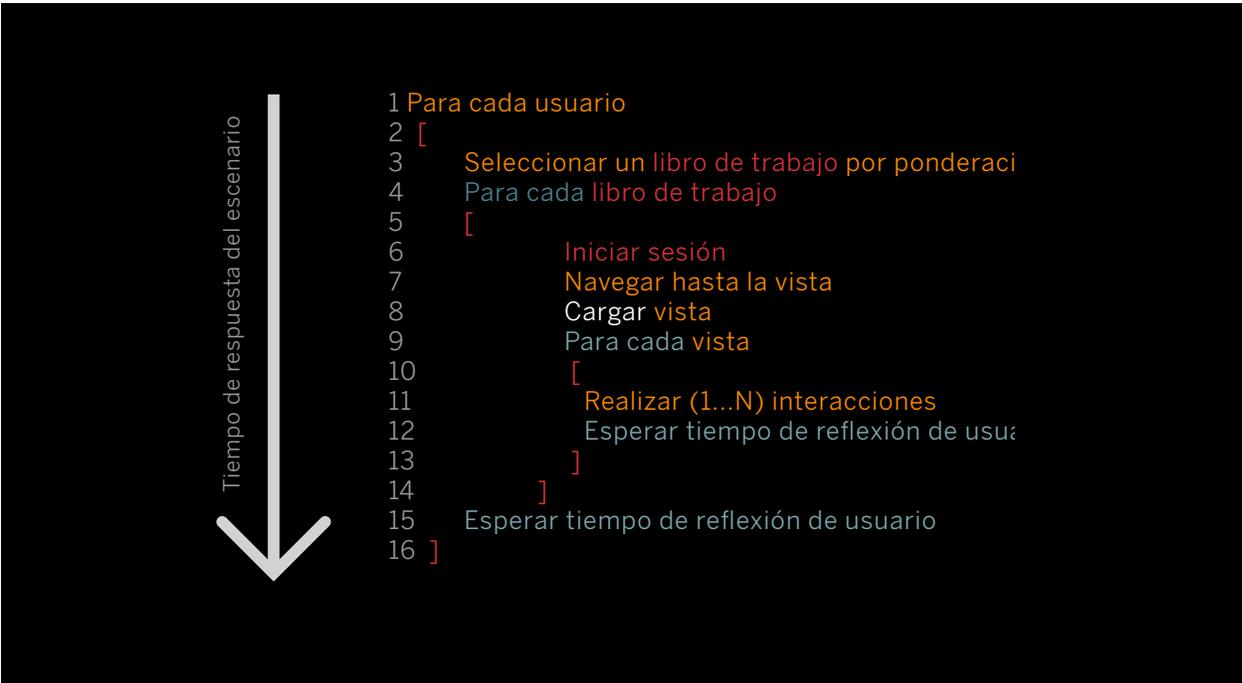


Figura 6: Algoritmo para pruebas de escenario

Los tiempos de respuesta registrados en este informe son mayores que los de los modelos de carga e interacción más simples, porque nuestros modelos de pruebas incluyen el tiempo necesario para completar el escenario. El tiempo de respuesta se mide y se registra por transacción desde el punto de vista del cliente. Esto significa que el escenario describe la perspectiva real del usuario, incluidas las variables de red del entorno, como la latencia.

Tiempo de respuesta

El tiempo de respuesta se mide como la cantidad de segundos que el servidor necesita para responder la solicitud del usuario final. Imagine esta situación: un usuario inicia sesión en el servidor, navega hasta una visualización, modifica un filtro en dicha visualización, espera a que se actualice y se muestre, y luego la analiza (tiempo de reflexión de usuario). El tiempo total “de reloj”, desde que el usuario inicia sesión en el servidor hasta el final del tiempo de reflexión de usuario, se registra como el tiempo de respuesta para esa interacción del escenario.

Capacidad de procesamiento de escenarios

La capacidad de procesamiento de escenarios es la cantidad de escenarios completados correctamente por segundo. Para calcular la capacidad de procesamiento de escenarios por segundo, ejecutamos escenarios durante una hora con una topología determinada al límite de la potencia de cálculo del sistema. La cantidad total de escenarios dividida por la cantidad de segundos en una hora (3600) da como resultado la capacidad de procesamiento de escenarios por segundo.

Por ejemplo, en el tiempo que tardó la prueba de ejecución de nuestros escenarios en una instancia de Tableau Server 10 de 8 núcleos, se logró completar 16 372 escenarios. Esto da como resultado una capacidad de procesamiento de escenarios por segundo de aproximadamente 4,5 (16 371/3600). Durante estos experimentos, forzamos el servidor con cargas mucho más altas que las observadas en producción, donde la capacidad de procesamiento de escenarios por segundo era menor que 1.

El aumento significativo que se observó en los datos experimentales se debe a que llevamos el servidor hasta sus límites, para poder conocer la máxima capacidad de rendimiento para el hardware y las unidades de escala elegidos.

Sin embargo, las implementaciones de producción experimentan ráfagas de cargas durante los períodos de uso pico. Además, los usuarios, por lo general, son mucho más lentos que lo simulado automáticamente en la ejecución de prueba. Salvando las diferencias, Tableau Public, por ejemplo, proporciona escalas masivas de aproximadamente 12 cargas de visualización (denominadas “impresiones”) por segundo, que se traducen en más de 7 millones de impresiones de visualizaciones de Tableau por semana.

Usuarios activos

La métrica de usuarios activos mide la cantidad de usuarios que utilizan Tableau Server de manera simultánea en el momento pico de una ventana de tiempo de una hora. En esta instancia, los escenarios incluyen inicio de sesión, carga de visualización, interacciones del usuario, búsqueda y otras acciones. Los usuarios activos se definen como usuarios que realizan cualquier actividad de este tipo en el momento pico de una ventana de tiempo de una hora.

Para averiguar la cantidad de usuarios activos que admite nuestra instalación de servidor, comenzamos por determinar la cantidad de usuarios que un solo servidor podía admitir sin perjudicar el tiempo de respuesta, sin superar una tasa de errores del 2 % (errores HTTP de Tableau Server) ni sobrepasar el 80 % de utilización de CPU.

Teniendo en cuenta las características de la carga de trabajo de producción, asignamos ponderaciones a libros de trabajo específicos. A continuación, generamos un subproceso (un usuario virtual) que seleccionara un libro de trabajo de manera aleatoria sobre la base de las ponderaciones asignadas y, luego, completamos todo el escenario descrito anteriormente. Una vez concluido el escenario, el subproceso esperaría un tiempo de reflexión de usuario específico. Al final de dicho período, el subproceso completaría su iteración y finalizaría. Durante la prueba, medimos la capacidad de procesamiento de escenarios, el tiempo de respuesta, las tasas de errores, y el uso de CPU y memoria, entre otras métricas. Mientras la utilización de CPU se mantuviese debajo del 80 %, la tasa de errores fuera inferior al 2 % y el tiempo de respuesta no se deteriorase, continuamos aumentando el número de subprocesos activos. Cuando se traspasaba alguno de estos umbrales, lo consideramos el momento indicado para medir la cantidad de usuarios activos que esa topología podía admitir.

Para validar la escalabilidad lineal del servidor, identificamos el punto indicado para un solo equipo, luego aumentamos la cantidad de usuarios virtuales de manera lineal hasta el incremento siguiente y confirmamos que las condiciones preestablecidas se seguían cumpliendo bajo la nueva carga.

Resultados

Ahora que hemos visto cómo realizamos la ejecución de las pruebas, la implementación que usamos y las métricas, vamos a revisar los resultados.

Escalabilidad lineal de Tableau Server 10

Lo primero que queríamos saber es de qué manera se escala Tableau Server 10. Con cargas de usuario cada vez mayores, observamos que Tableau Server 10 se escala de manera lineal con la carga, cuando se agregan más nodos de trabajador al clúster. En la figura a continuación se muestra la cantidad de escenarios completados por segundo a medida que el número de nodos de trabajador aumenta.

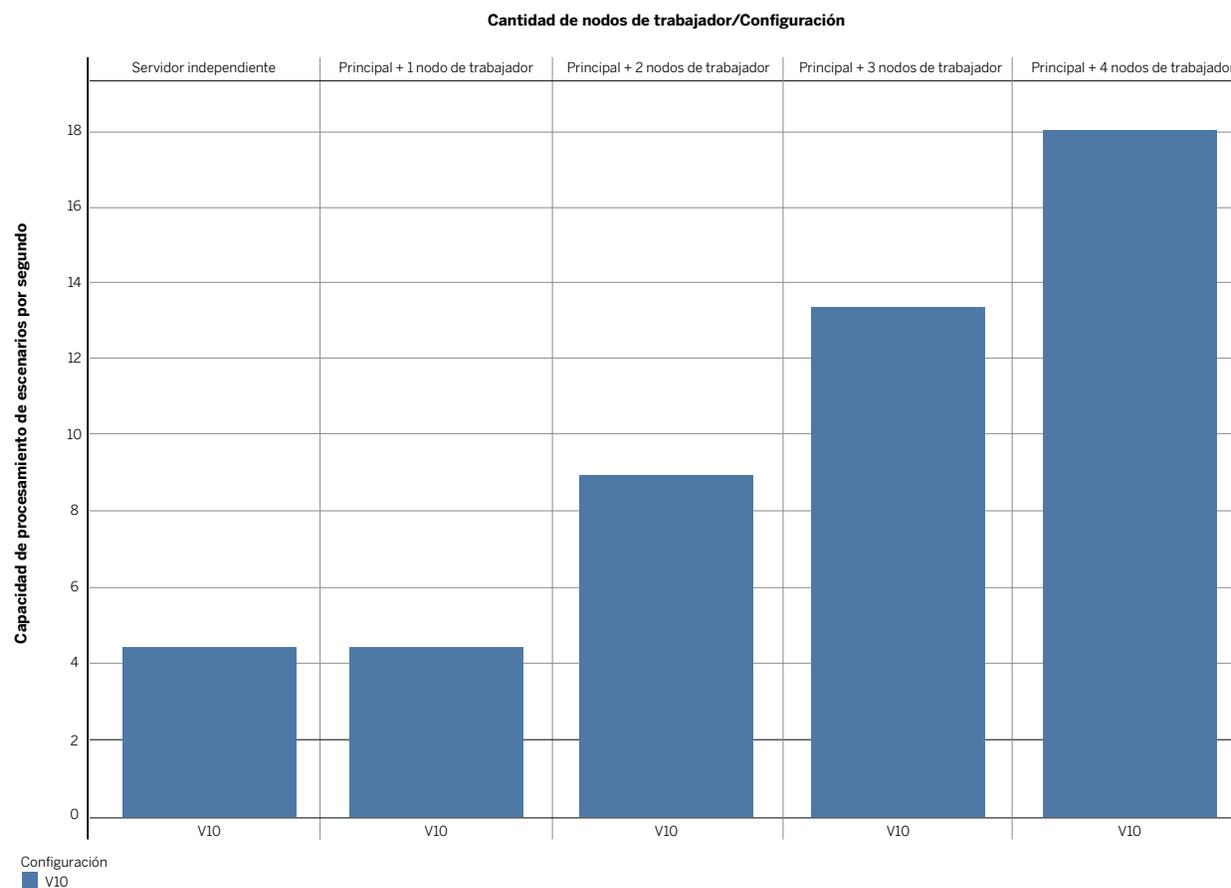


Figura 7: Capacidad de procesamiento de escenarios por segundo

En cada columna se muestra la topología del clúster. En la columna del extremo izquierdo se muestra una configuración de un solo servidor. En la segunda columna se muestra una configuración de clúster de un equipo principal/trabajador más un solo nodo de trabajador. En las columna siguientes, agregamos un nodo más de trabajador, como se detalló anteriormente. La altura de las barras indica el promedio de procesamiento de escenarios por segundo. El procesamiento de escenarios por segundo representa la cantidad de trabajo que realiza un servidor. Como se mostró anteriormente, el procesamiento de escenarios por segundo aumenta de manera lineal a medida que agregamos más nodos de trabajador. Observamos que, a medida que aumentábamos las cargas en un clúster, la utilización de CPU en general promediaba el 80 %. En la figura siguiente, se muestra la utilización de CPU en todo el clúster con cargas incrementales y una banda de confianza del 95 %.

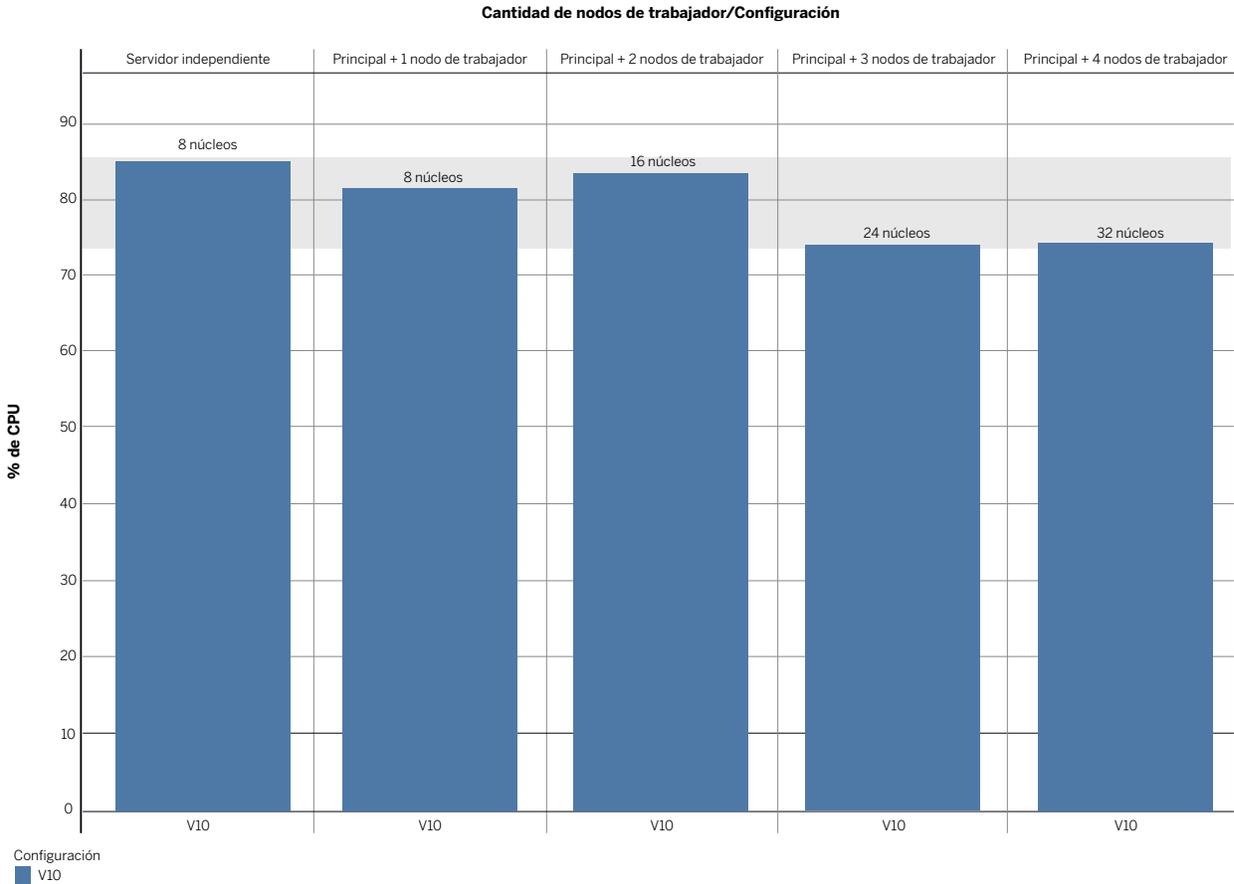


Figura 8: Utilización de CPU con cargas incrementales en todo el clúster

Como se muestra en la figura 8, la incorporación de nodos de trabajador para distribuir la carga de CPU en todo el clúster optimiza el sistema, puesto que proporciona una mayor capacidad de aumento durante las ráfagas de carga. En los casos en que se configuraron menos nodos de trabajador en un clúster determinado, la utilización de CPU era comparativamente mayor que en los clústeres con más nodos de trabajador. En estos casos con menos nodos de trabajador, los procesos ligados al procesamiento compiten por recursos limitados. Durante las pruebas, observamos que las tasas de errores del servidor estaban muy por debajo de la meta del 2 % que habíamos establecido como parte de la metodología. Según sus cargas de trabajo, es posible que experimente tasas de errores mayores (menor calidad de servicio) cuando los clústeres estén limitados por una menor cantidad de equipos o una reducción en la capacidad.

La siguiente pregunta que nos hicimos fue cómo responde Tableau Server 10 en comparación con Tableau Server 9.3 si empleamos la misma metodología de pruebas.

Según los resultados, cuando se utiliza la misma metodología con ambas versiones, la capacidad de procesamiento de Tableau Server 10 mejoró en comparación con la de Tableau Server 9.3.

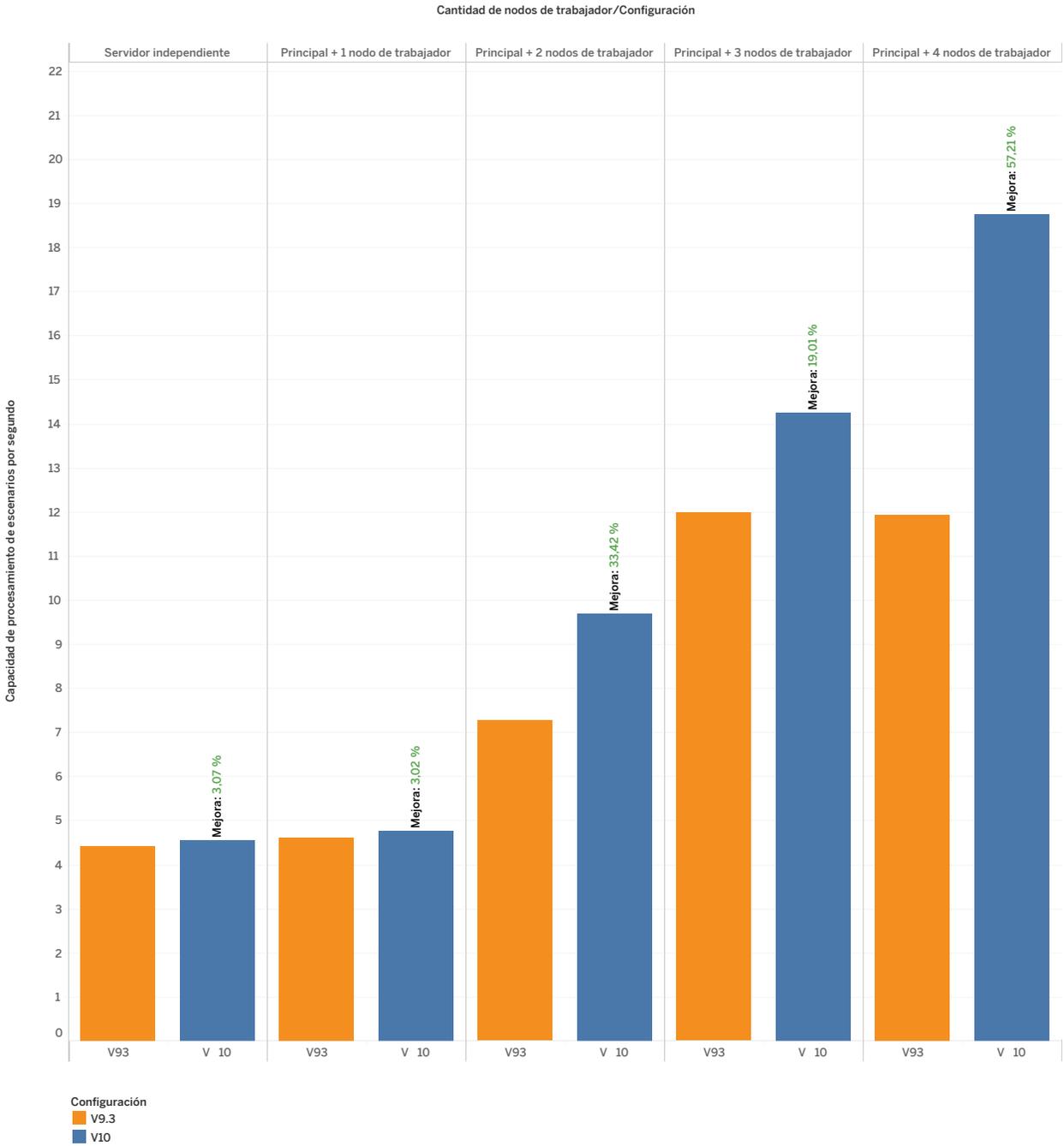


Figura 9: Comparación de procesamiento de escenarios por segundo entre la versión 9.3 y la 10.0.1

En la figura 9, se compara el procesamiento de escenarios de Tableau Server 9.3 (barras naranjas) con el de Tableau Server 10 (barras azules). En cada panel, se muestra la configuración de la topología del clúster, descrita en el encabezado de columna. Como se mencionó anteriormente, cuando agregamos más nodos al clúster, Tableau Server 10 no solo se escala de manera lineal, sino que, además, su procesamiento de escenarios se escaló mejor en comparación con Tableau Server 9.3.

Cuando examine las versiones, tenga en cuenta que la única comparación adecuada que podemos hacer es a través del cotejo de las dos versiones con la metodología mejorada de pruebas, que captura los escenarios de usuario reales. Por esta razón, no es riguroso comparar los resultados de Tableau Server 10 con los del informe anterior, puesto que la metodología de pruebas se modificó significativamente.

Aunque es importante analizar la capacidad de procesamiento, queríamos garantizar que el tiempo de respuesta del usuario final para todo el escenario de prueba tuviese un rendimiento adecuado y no produjera fallos bajo cargas mayores.

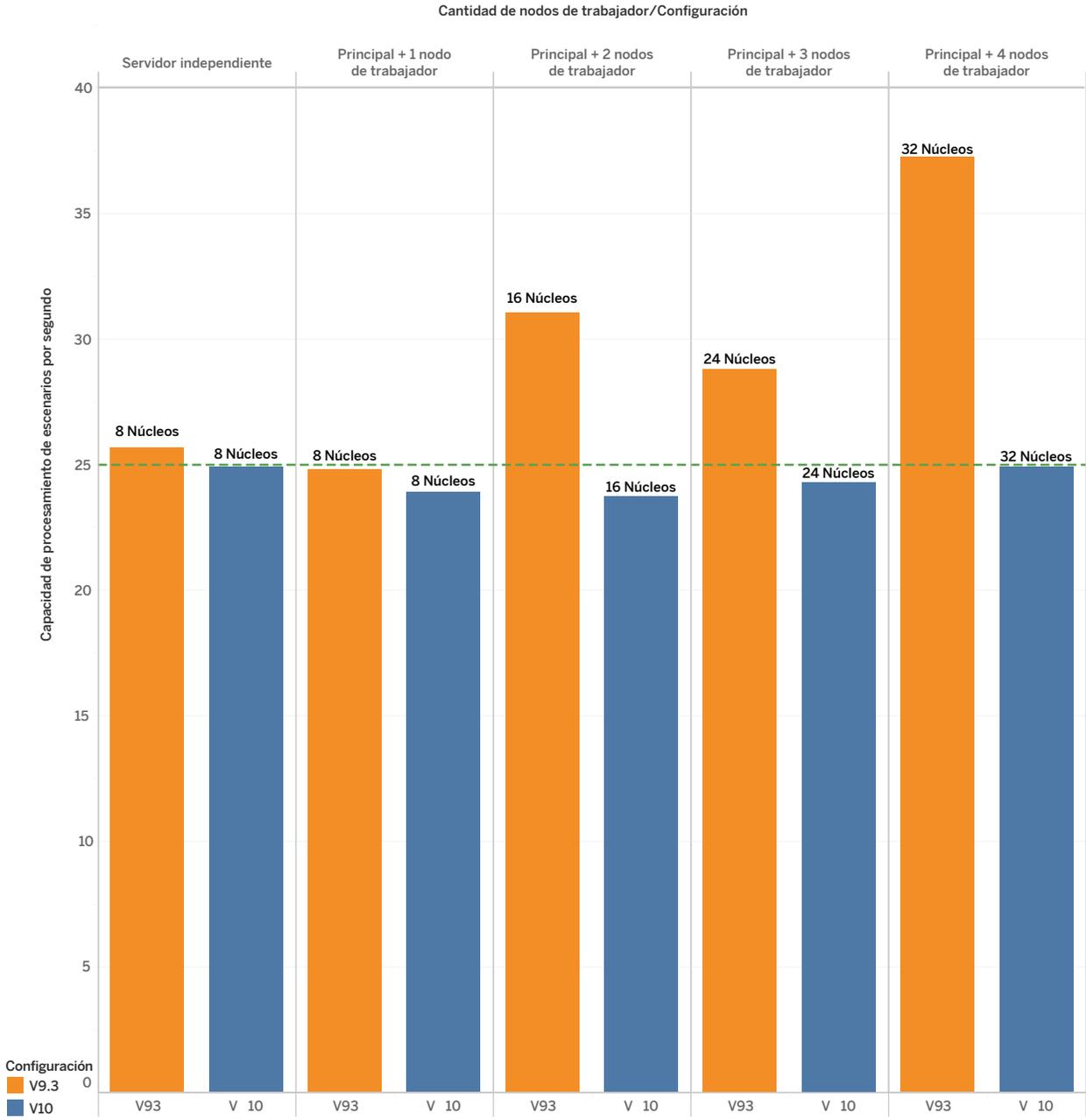


Figura 10: Comparación de tiempos de respuesta promedio de los escenarios (en segundos) en las versiones 9.3 y 10.0.1

En la figura 10, la altura de las barras naranjas indica el tiempo de respuesta promedio de los escenarios en la versión 9.3. La altura de las barras azules indica la misma métrica para la versión 10.0.1. A medida que elevamos las cargas de usuarios finales en el clúster con la versión 9.3, observamos que los tiempos de respuesta aumentaban incrementalmente. Por su parte, Tableau Server 10 demostró una mayor homogeneidad en los tiempos de respuesta a medida que continuábamos agregando usuarios virtuales al sistema. Por lo tanto, se puede inferir que, bajo las mismas cargas, Tableau Server 10 proporciona a los usuarios finales mejor rendimiento y capacidad de respuesta que Tableau Server 9.3.

Algunas de las mejoras que observamos en Tableau Server 10, con tiempos de respuesta relativamente homogéneos y estables bajo cargas en aumento, se dieron como resultado de las mejoras implementadas en los componentes de almacenamiento en caché. Específicamente, en la versión 10, implementamos optimizaciones del almacenamiento en caché para el escenario de respuesta de arranque. El escenario de arranque es la primera solicitud para inicializar y almacenar datos en caché de una sesión de usuario. Sin el almacenamiento en caché, la versión anterior de Tableau Server procesaba y almacenaba los datos de arranque de cada solicitud subsiguiente, incluso si eran muy similares o idénticas. En la versión 10, implementamos operaciones inteligentes de almacenamiento en caché para este escenario. Gracias a estas mejoras, Tableau Server 10 es más eficaz, puesto que puede procesar una mayor carga de usuario al mismo tiempo que mantiene buenos tiempos de respuesta. Sin embargo, ahora el servidor debe realizar más trabajo de procesamiento, como observamos anteriormente con la capacidad de procesamiento aumentada.

Todos los resultados mencionados anteriormente provienen de examinar el escalamiento de Tableau Server durante un uso incrementado de análisis por parte de los usuarios finales. A medida que la carga de trabajo de usuarios finales aumenta en el servidor, se debe garantizar la capacidad suficiente a través de la incorporación de más nodos de trabajador al clúster para proporcionar una buena calidad de servicio a dichos usuarios.

El conjunto de preguntas siguiente giró en torno a cómo afectan al usuario el lugar y el modo en que se configuran los procesos en segundo plano. Concretamente, cómo afecta a los usuarios que los procesos en segundo plano se hospeden en el mismo equipo que los servicios de análisis (VizQL Server) o que se aislen dichos procesos en un equipo diferente. Analizamos esos resultados en la sección a continuación.

Resultados de los procesos en segundo plano

En primer lugar, queríamos cuantificar el impacto de ejecutar los procesos en segundo plano en nuestra instalación predeterminada de servidor único. ¿Cuál es el impacto de este escenario en los usuarios finales y en la escalabilidad general? Realizamos experimentos de carga de trabajo y registramos el procesamiento de escenarios por segundo en una implementación única de Tableau Server. Luego, hicimos pruebas de cargas en los clústeres, ampliando la topología del clúster con nuevos nodos de trabajador.

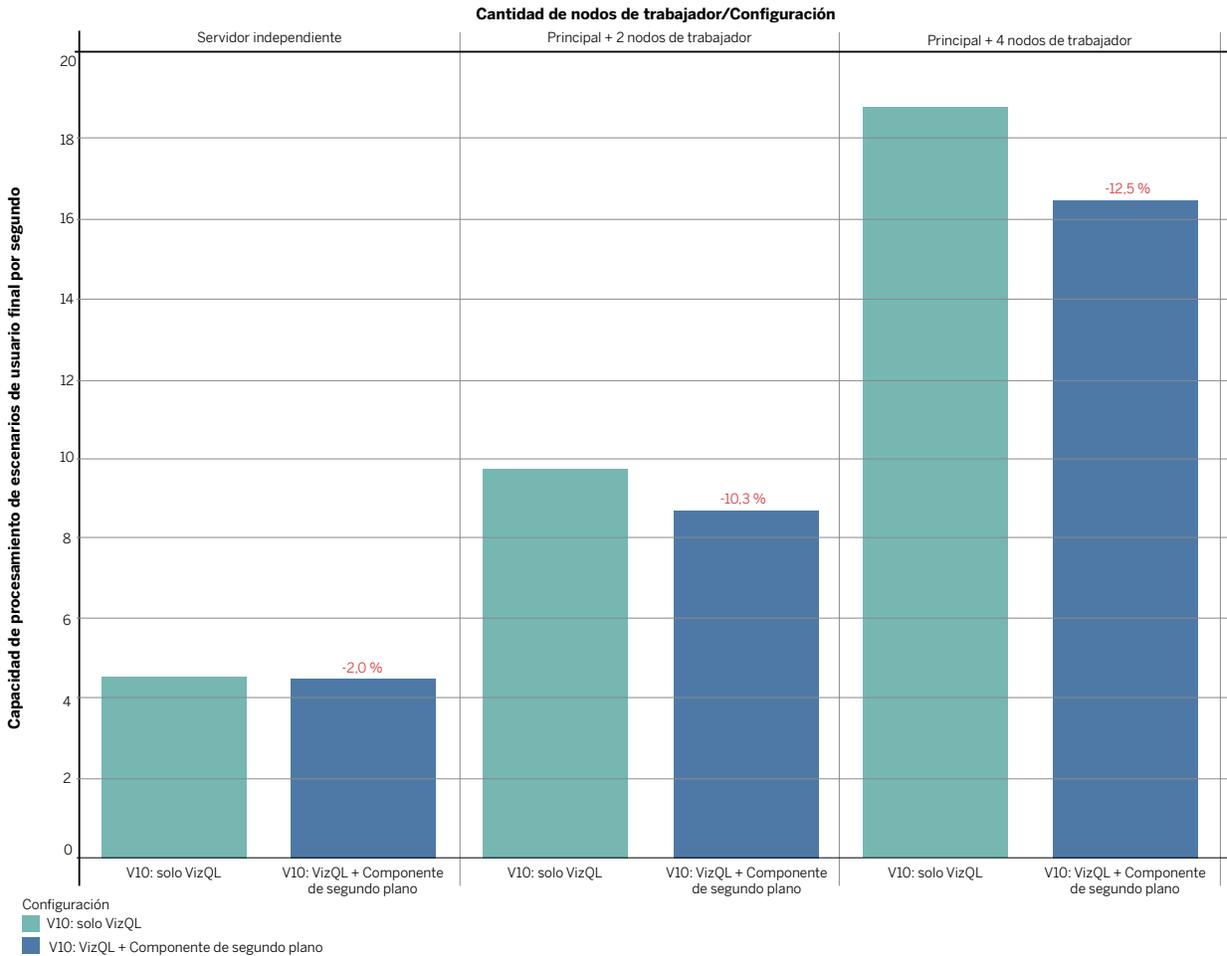


Figura 11: Impacto en la escalabilidad de usuarios finales cuando los servicios de procesos en segundo plano y VizQL se hospedan en el mismo equipo

En la figura 11, se muestran los resultados de los dos escenarios de prueba. Las barras verdes representan el escenario de prueba en el que ejecutamos únicamente la carga de trabajo de VizQL. La carga de trabajo de solo VizQL simula las cargas de trabajo de los usuarios finales. Las barras azules representan el escenario de prueba en el que agregamos una cantidad fija de cargas de trabajo en segundo plano a la carga de trabajo de VizQL en el mismo clúster. A continuación, registramos los cambios en el procesamiento de escenarios por segundo para la carga de trabajo de usuarios finales en los dos escenarios. Sabíamos que tendría un impacto, porque tanto VizQL Server como los procesos en segundo plano suponen cargas de trabajo que requieren mucho procesamiento. Sin embargo, nuestro objetivo era medir este impacto para estas cargas de trabajo específicas.

Observamos una reducción de entre 2 y 12 % en el procesamiento de escenarios general de usuarios finales. Esta reducción demuestra que la capacidad del clúster de proporcionar servicios a los usuarios finales se ve influenciada por el costo del procesamiento de la carga de trabajo en segundo plano. Al extrapolar estos datos, observamos que, si el clúster procesaba 100 escenarios de usuarios finales por unidad de tiempo, la incorporación de una carga constante en segundo plano reducía la capacidad a 88 en la misma unidad de tiempo. Esta reducción en la capacidad de procesamiento podría suponer un impacto significativo en la escalabilidad de usuarios finales o en la calidad del servicio, según múltiples factores, como las cargas de trabajo, las ráfagas de cargas pico, las limitaciones de hardware y las variables de infraestructura.

Esta prueba demuestra la importancia de contar con recursos de hardware adecuados para el planeamiento de cargas de trabajo de Tableau Server. Ejecutar Tableau Server en un equipo de menor capacidad o con limitaciones puede dar lugar a capacidades de procesamiento reducidas, fallos en las tareas en segundo plano, retrasos en las notificaciones de suscripción o errores para el usuario final que se manifiestan como problemas de rendimiento. Cuando se producen dichas condiciones, considere la posibilidad de expandir el clúster para aislar las cargas de trabajo en segundo plano en un equipo dedicado. Si aísla los servicios en segundo plano, liberará procesos paralelos y ligados al procesamiento que, de otra manera, competirían entre sí por los mismos recursos cuando se hospedan en un mismo equipo.

Revisemos el impacto en la carga de trabajo de suscripciones cuando los procesos en segundo plano se hospedan en el mismo equipo que los servicios de VizQL Server y cuando se hospedan por separado.

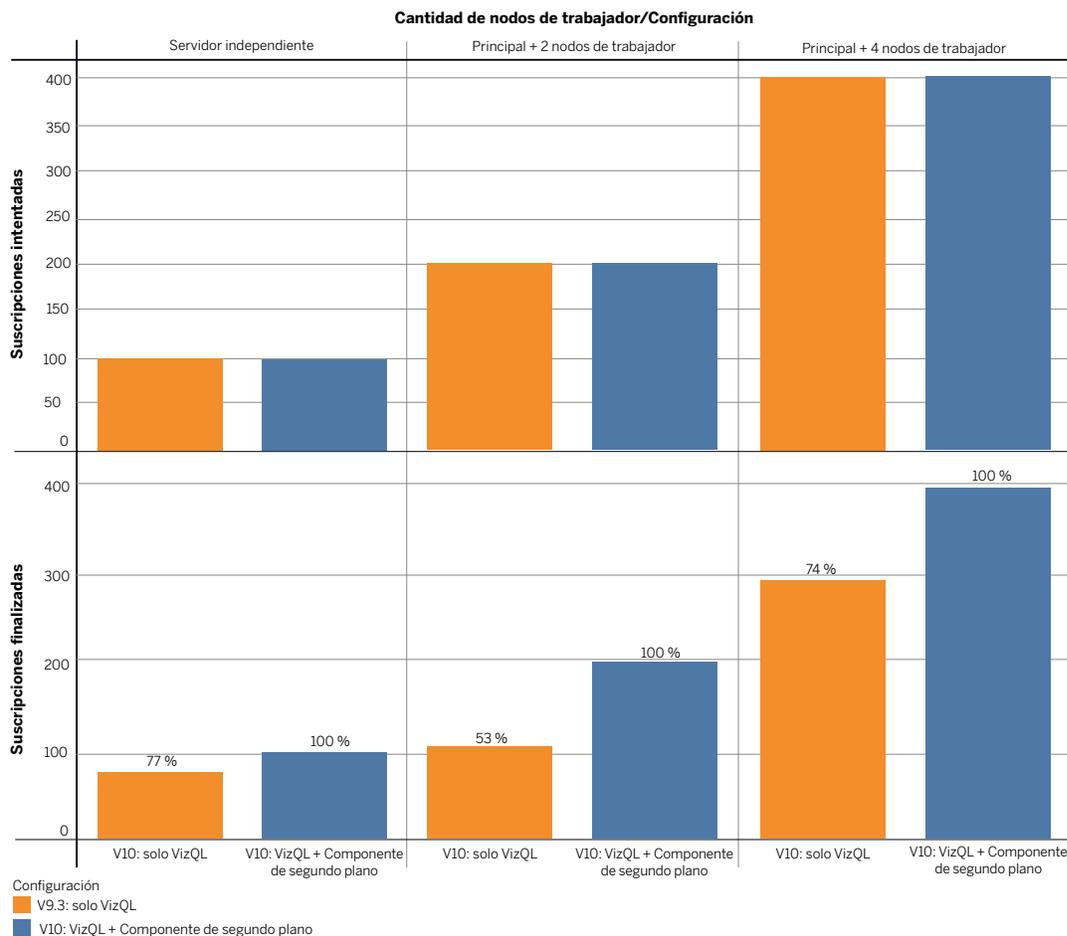


Figura 12: Mejoras en las notificaciones de suscripción

En la figura 12, cada panel representa la topología de clúster de Tableau Server 9.3 (naranja) y 10 (azul), mientras que cada barra indica la cantidad de notificaciones de suscripción que se intentaron enviar y la cantidad de notificaciones completadas con éxito. En cada iteración, aumentamos la cantidad de nodos de trabajador en topologías de 0, 2 y 4 nodos. Además, cada nodo de trabajador tenía configurado un solo proceso en segundo plano. (La omisión de las pruebas con 1 y 3 nodos de trabajador no comprometió la fidelidad de los resultados buscados). Limitamos el proceso en segundo plano a uno por nodo de trabajador a fin de establecer una medida de comparación homogénea.

A medida que se aumentó la carga de suscripciones en segundo plano, Tableau Server 10 pudo completar todo el trabajo que le fue solicitado. En las mismas pruebas, Tableau Server 9.3 comenzaba a saturarse y algunas suscripciones fallaban. Este comportamiento podría verse agravado en una topología sin los recursos suficientes y con una carga mayor que su capacidad. Además, observamos que Tableau Server 10 podía procesar una mayor cantidad de suscripciones a medida que se agregaban más nodos de trabajador al sistema. Una mejora de Tableau Server 10 que puede explicar esos resultados está relacionada con la introducción de almacenamiento en caché de imágenes para las notificaciones. Esta funcionalidad permite que el proceso en segundo plano realice mucho menos trabajo para las mismas tareas vinculadas a una programación. En los escenarios en que se ejecutan las mismas tareas en la misma programación, Tableau Server 10 ahora almacena en caché los resultados de la primera ejecución y los proporciona a las solicitudes subsiguientes con la misma carga de trabajo. Esto significa que el mismo trabajo se completa de una manera más eficaz.

Para las suscripciones que utilizan los mismos libros de trabajo en la misma programación, observamos que el tiempo necesario para completar la suscripción se redujo en un 60-90 %, en comparación con la versión 9.3. En la figura 13 se muestran estas mejoras de manera gráfica.

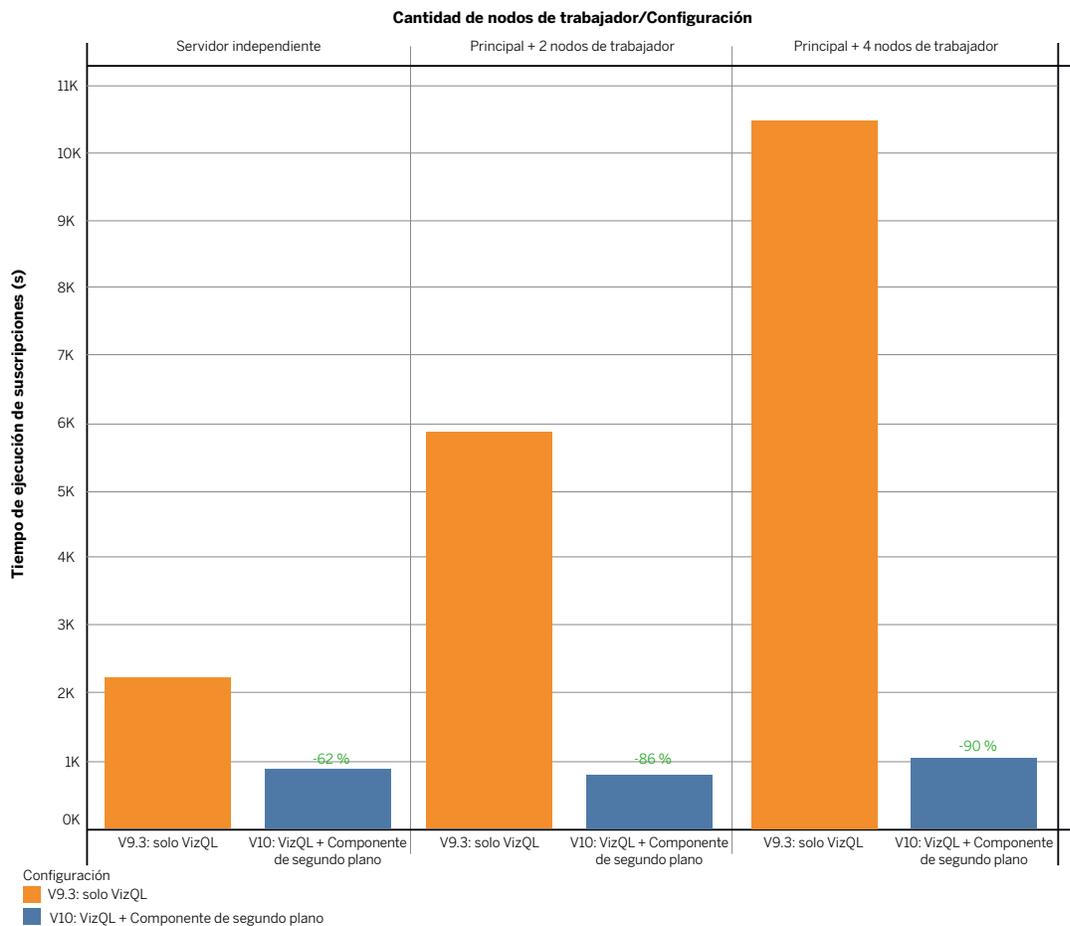


Figura 13: Tiempo de ejecución de suscripciones (tiempo para finalización) en las versiones 9.3 y 10

Esto significa que cualquier notificación sobre el mismo libro de trabajo en la misma programación tardará mucho menos tiempo en completarse. Sin embargo, si las notificaciones se relacionan con diferentes libros de trabajo o se configuran filtros de usuarios, las suscripciones consumirán tiempo de procesamiento y cálculo antes de que se puedan enviar al usuario. Los distintos libros de trabajo y filtros de usuarios requieren que Tableau Server procese toda la canalización de visualización para el usuario final, lo que incluye procesos de consulta y visualización de datos para cada libro de trabajo o vista filtrada.

La ventaja de las suscripciones es que proporcionan datos pertinentes a los usuarios corporativos de manera oportuna. Las actualizaciones de extracciones eficaces ayudan a tomar mejores decisiones, puesto que se basan en datos adecuadamente actualizados. Debido a que el proceso en segundo plano administra estas dos funciones cruciales, si planea las programaciones de suscripciones de manera que el trabajo duplicado esté vinculado a la misma programación, puede lograr beneficios garantizados en el almacenamiento en caché.

Aislamiento del proceso en segundo plano

Realizamos los mismos experimentos con el proceso en segundo plano aislado en su propio nodo de trabajador. Observamos que el nodo de trabajador con el proceso en segundo plano fue capaz de completar 400 suscripciones en un único equipo de 4 núcleos. Es el mismo número de suscripciones que se registraron cuando un único proceso en segundo plano estaba hospedado junto a los procesos de VizQL distribuidos en 4 máquinas de trabajador.

Aquí, lo importante es que, mientras se escala el proceso en segundo plano, aislarlo proporciona una escalabilidad similar, aunque no tiene un impacto en la calidad del servicio para el usuario final. El proceso en segundo plano es de un solo proceso y está diseñado para completar tareas lo más rápido posible. Debido a este diseño, el proceso en segundo plano consumirá el núcleo completo cuando tenga trabajo que hacer. En un equipo aislado, el uso intensivo de capacidad de procesamiento del proceso en segundo plano no interferirá con otros servicios orientados al usuario.

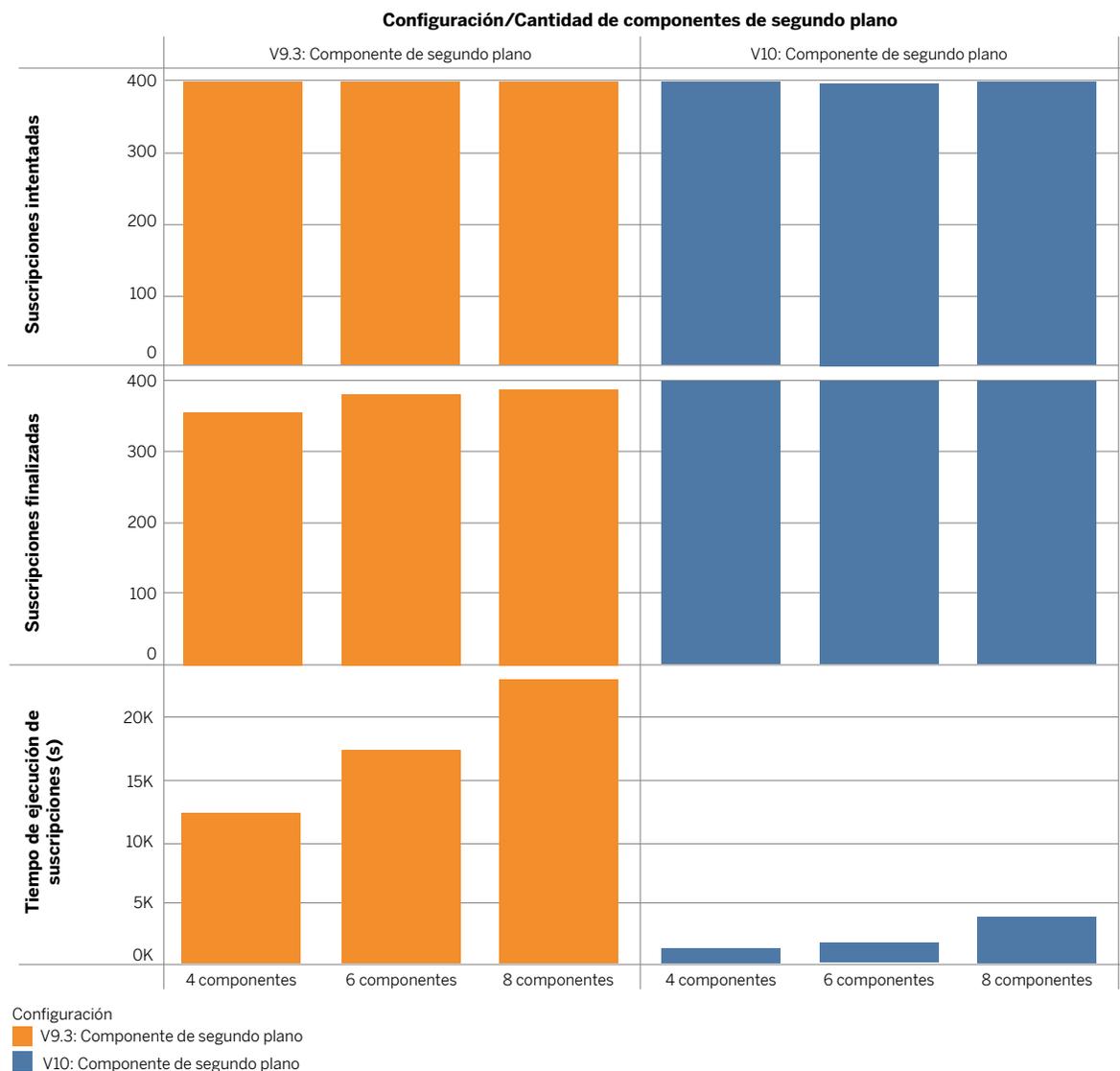


Figura 14: Incorporación de procesos en segundo plano a un equipo de 4 núcleos en las versiones 9.3 y 10

Como se muestra en la figura 14, al incorporar procesos en segundo plano a un único equipo de 4 núcleos, Tableau Server 10 (azul) pudo completar suscripciones en un tiempo significativamente menor, en comparación con la versión 9.3 (naranja). Sin embargo, si se continúa incorporando procesos en segundo plano a un equipo limitado físicamente por sus núcleos, se producirá un impacto negativo. Como se indica, la incorporación de 8 procesos en segundo plano a un equipo de 4 núcleos dilata el tiempo de finalización. Esto se debe al diseño de un solo proceso del proceso en segundo plano.

Por último, en los experimentos en que evaluamos las actualizaciones de extracciones en nodos de trabajador que ejecutaban procesos aislados en segundo plano, observamos que Tableau Server 9.3 y 10 tenían resultados similares. Ambas versiones completaron la misma cantidad de actualizaciones de extracciones en aproximadamente la misma cantidad de tiempo. Un detalle importante que debe tener en cuenta al escalar para dar cabida a las actualizaciones de extracciones es que su rendimiento depende en gran medida de las bases de datos externas. (En nuestras pruebas, actualizamos datos mediante libros de trabajo que utilizaban extracciones publicadas de MS SQL Server). El rendimiento y la escalabilidad de las actualizaciones de extracciones dependen en gran medida de las especificaciones de hardware de base

de datos. Además, las características de los datos, como los tipos de uniones y la complejidad de las consultas ejecutadas, tendrán un impacto en la escalabilidad. Por esta razón, debe asegurarse de que las actualizaciones de extracciones y las notificaciones tengan capacidad suficiente disponible para finalizar el procesamiento antes de que se produzcan cargas pico de usuarios finales en el sistema.

Consideraciones sobre los procesos en segundo plano

El proceso en segundo plano asume la mayor parte del trabajo relacionado con las actualizaciones de extracciones, las suscripciones y otros trabajos en segundo plano programados. Estos trabajos no afectan demasiado a la capacidad si se programan para que se realicen en horas de poca actividad. Si esto no es posible, debe planearse una mayor capacidad para los procesos de segundo plano y otras cargas de trabajo no orientadas al usuario, a fin de que se ejecuten de forma simultánea a los procesos orientados al usuario.

Los procesos en segundo plano están diseñados para consumir la capacidad por proceso de un núcleo completo porque, por naturaleza, deben terminar el trabajo lo más rápido posible. Cuando ejecute varios procesos en segundo plano, debe tener en cuenta que dichos procesos competirán por los recursos de procesamiento y de red con otros servicios que se ejecuten en el mismo equipo.

Prácticas recomendadas: pruebas de escalabilidad personalizadas

A continuación le ofrecemos algunas prácticas recomendadas por si desea realizar sus propias pruebas de carga para saber cómo escala Tableau Server en su entorno con sus cargas de trabajo.

1. **No trate Tableau Server como si fuera una caja negra.** A menudo, las pruebas de carga tradicionales tratan a la aplicación evaluada como si fuera una caja negra. Esto implica que no se realizará ningún tipo de ajuste o configuración de la implementación para satisfacer las condiciones de carga. Tableau está diseñado para poder escalarse en vertical y en horizontal. Comprender la arquitectura de Tableau a fin de impulsar resultados que funcionen para su situación es útil para las pruebas de escalabilidad.
2. **Elija la herramienta adecuada para las pruebas.** Tableau Server es un caballo de batalla y realiza tareas complejas que requieren muchos recursos. Existen muchas herramientas disponibles para aplicar cargas en Tableau Server. Aunque Tableau no es directamente compatible con ninguna de estas herramientas, usted debe elegir la que le resulte más fácil de usar y represente mejor su entorno de producción. También debe asegurarse de conocer adecuadamente tanto las herramientas como Tableau Server cuando realice las pruebas de carga. Para nuestras pruebas, utilizamos [TabJolt](#). TabJolt es una herramienta para pruebas de carga basada en JMeter y desarrollada para eliminar el mantenimiento de scripts mientras trabaja con soluciones de análisis ad hoc, como Tableau.
3. **Seleccione libros de trabajo representativos.** A menudo, las quejas sobre el rendimiento o la escalabilidad se deben a que los libros de trabajo que se usaron no se crearon teniendo en cuenta las prácticas recomendadas. Si en una prueba con un solo usuario, sus libros de trabajo muestran un tiempo de respuesta muy lento, debería optimizar los libros de trabajo antes de realizar pruebas de carga. De la misma manera que evitaría el uso de un dashboard con mal rendimiento en el entorno de producción, tampoco lo debería usar en el de pruebas.

4. **Comience con la configuración predeterminada.** Cuando realice pruebas de libros de trabajo con conexiones en tiempo real, recuerde que con la introducción de las consultas en paralelo en Tableau Server 9.0 es posible que ya no necesite tantos VizQL Server como con las versiones anteriores de Tableau Server. Comience con la nueva configuración predeterminada de 2 procesos y escale en vertical de forma incremental según sea necesario.

Prácticas recomendadas para la optimización en el mundo real

Además de usar un sistema de diseño óptimo, conviene seguir ciertas prácticas recomendadas para mejorar en gran medida el rendimiento y reducir el tiempo de respuesta promedio.

- Diseñe sus libros de trabajo teniendo en cuenta tanto la apariencia como el rendimiento. A menudo, cuando los clientes comentan que su libro de trabajo funciona lento, se debe a que fue diseñado sin tener en cuenta el rendimiento. Si el tiempo de carga para un único usuario es lento, los tiempos de respuesta del libro de trabajo también serán lentos con cargas más grandes. Para adoptar una cultura de análisis, es recomendable proporcionar guías y equipos que ayuden a los usuarios a diseñar libros de trabajo informativos, de apariencia agradable y con un buen rendimiento. Esto también garantizará la capacidad de desarrollar y ofrecer visualizaciones escalables. El informe [Diseño de libros de trabajo eficaces](#) contiene detalles acerca de cómo generar dashboards eficaces que tengan un buen rendimiento.
- El tiempo de respuesta total para un usuario final es una combinación de varios aspectos. Pero, principalmente, es tiempo que tarda el procesamiento de Tableau combinado con la recuperación de datos. Si sus bases de datos back-end o los tiempos de consulta son lentos, la visualización también será lenta. Es importante que tenga esto en cuenta cuando defina su estrategia de datos. A menudo, se seleccionan y comparten las fuentes de datos de una organización. Debe asegurarse de proporcionar datos pertinentes de manera que satisfagan las necesidades de productividad de los usuarios corporativos. Esto significa que debe optimizar los datos. Por ejemplo, para lograr que las consultas de tablas indexadas sean rápidas, debe garantizar que las uniones sean óptimas y que los niveles de agregación sean pertinentes. Contar con un adecuado proceso de limpieza de datos es importante para mantener en perfecto estado las visualizaciones y el rendimiento.
- Utilice extracciones de datos de Tableau. Si las consultas de su base de datos son lentas, considere la posibilidad de usar extracciones para aumentar el rendimiento de las consultas. Las extracciones se almacenan localmente en el servidor y se ejecutan en la memoria. Por esta razón, los usuarios pueden acceder a los datos de manera rápida sin necesidad de enviar solicitudes a la base de datos. Las extracciones se pueden filtrar y se agregan fácilmente. Además, son ideales cuando los usuarios no necesitan detalles del nivel de fila. Las extracciones mejoran significativamente el tiempo de respuesta y permiten que los usuarios permanezcan en el flujo de análisis.
- Programe las actualizaciones en horas de poca actividad. A menudo, las fuentes de datos se actualizan en tiempo real, pero los usuarios solo necesitan los datos una vez al día o a la semana. La programación de las extracciones en horas de poca actividad puede reducir la carga de la hora pico tanto en la base de datos como en Tableau Server. Además, podría agregar procesos en segundo plano adicionales en máquinas existentes o usar hardware dedicado si tiene la capacidad de núcleos suficiente. Considere esta opción para completar las extracciones más rápidamente.

- Evite realizar operaciones que consumen muchos recursos durante las horas pico. La publicación, especialmente la de archivos muy grandes, es una tarea que consume muchos recursos. A menudo es sencillo cambiar el comportamiento de publicación de los usuarios: pídale que publiquen durante las horas de poca actividad y que eviten los momentos de mucho tráfico, como los lunes por la mañana. Para saber los momentos de mayor actividad de sus servidores, utilice las [vistas del administrador](#) y cree una política basada en el uso real. Según la configuración de Tableau Server 10.0, la publicación conllevará la realización de una copia de las extracciones en cada uno de los nodos del clúster para que haya una alta disponibilidad. Realizar esta tarea en horas de poca actividad le permitirá maximizar el ancho de banda de su red.
- Vistas de la caché A medida que varios usuarios van accediendo a Tableau Server, el tiempo de respuesta aumenta inicialmente debido a la disputa por los recursos compartidos. Cuando está activado el almacenamiento en caché, las vistas de cada solicitud que recibe el sistema se almacenan en caché y se representan más rápidamente para el próximo usuario del mismo dashboard.
- El proceso de servidor de caché, que fue una de las novedades de Tableau Server 9.0, puede prepararse programando un correo electrónico que envíe vistas comunes en cuanto haya finalizado la actualización de extracciones. De esta manera, los usuarios futuros estarán usando los datos de la caché de la solicitud anterior. Puede usar otros métodos para preparar la caché, como una herramienta automatizada para cargar las visualizaciones clave que normalmente tienen un tráfico elevado. En cualquier momento, el usuario puede invalidar manualmente la caché de la consulta externa para actualizar sus datos a partir de la fuente de datos. Esta acción también fuerza la regeneración de la memoria caché. De esta manera, los usuarios siempre pueden obtener una copia actualizada de los datos, independientemente de la versión que haya en la caché.

Resumen

Tableau Server 10 es una plataforma escalable y preparada para el uso empresarial, capaz de ofrecer servicios a organizaciones de cualquier tamaño. Puede operar en nubes privadas hospedadas en instalaciones físicas o en nubes públicas. Además, puede escalarse linealmente aumentando la capacidad de nodos de trabajador. Aunque cada entorno es único y tiene sus propias características y configuración, la arquitectura de Tableau Server le permitirá adaptar sus implementaciones a las necesidades de los usuarios.

A pesar de que la escalabilidad y el rendimiento pueden variar y que estas no son recomendaciones específicas para todas las situaciones, Tableau Server 10 puede dar soporte a equipos o departamentos de entre 25 y 100 usuarios con una capacidad de entre 8 y 16 núcleos. Si está buscando una configuración que pueda dar soporte a entre 100 y 1000 usuarios, dependiendo del nivel de uso y necesidades de actualización de datos, es posible que entre 16 y 24 núcleos sean suficientes para comenzar. En el caso de que cuente con más usuarios o necesite mayores cargas en segundo plano, puede ampliar su implementación a entre 32 y 64 núcleos. Podrá dar soporte a cargas de trabajo cada vez mayores mediante la incorporación de nodos de trabajador, llegando incluso a escalas de la nube.

Acerca de Tableau

Tableau ayuda a las personas a transformar los datos en información útil para generar un impacto positivo. Conéctese con facilidad a datos almacenados en cualquier formato y lugar. Haga, rápidamente, análisis ad hoc que revelen oportunidades ocultas. Arrastre y suelte para crear dashboards interactivos con análisis visuales avanzados. Después, compártalos con toda su organización y permita que sus compañeros de equipo exploren los datos por sí mismos. Multinacionales, empresas pequeñas y emergentes... Todo el mundo usa la plataforma de análisis de Tableau para ver y comprender sus datos.

Recursos

[Tableau para la empresa: descripción general de TI](#)

[Guía de administrador de Tableau Server](#)

[Alta disponibilidad de Tableau Server 10.0: análisis imprescindibles a gran escala](#)

[Tableau on Amazon Web Services \(Tableau en Amazon Web Services\)](#)

