4 Planeación de capacidades

Con este capítulo de las notas empezamos la última parte de nuestro curso: Planeación de Capacidad. La planeación de capacidad es el proceso que permite a una persona u organización construir un plan de crecimiento de su infraestructura a corto, mediano y largo plazo. Las fases o hitos en el plan de crecimiento deben estar basados en el plan de crecimiento o evolución del negocio. Es decir, la infraestructura solo debe crecer si el negocio así lo requiere y los directivos así lo deciden.

En la construcción del plan de crecimiento de la capacidad es necesario comprender el estado actual de la infraestructura por medio del análisis de desempeño; tema que estudiaremos en este capítulo. El capítulo se basa en una fuente principal: The Art of Computer Systems Performance Analysis de Raj Jain.

El capítulo está organizado así: la sección 1 presenta una introducción del proceso de análisis de desempeño y la sección 2 discute los objetivos de un proceso de análisis de desempeño. Finalmente, la sección 3 presenta conceptos fundamentales y recomendaciones para adelantar un proceso de análisis de desempeño.

4.1 Análisis de desempeño

El análisis de desempeño es el proceso de evaluar el desempeño, es decir, comprender qué tan bueno es el comportamiento, de uno o varios componentes de una infraestructura de cómputo, para identificar qué tipo de acciones podemos tomar a nivel técnico, o el tipo de acciones que podemos sugerir a quienes toman decisiones de negocio, para mejorar los resultados.

El análisis de desempeño es un concepto relacionado con planeación de capacidad y con evaluación comparativa (benchmarking), pero son conceptos diferentes. Por claridad, los siguientes párrafos presentan brevemente cómo se relacionan y cuáles son sus diferencias.

Planeación de Capacidad y Análisis de Desempeño. Los procesos de planeación de capacidad y análisis de desempeño están relacionados. El primero sirve para determinar qué características de desempeño debe tener la infraestructura necesaria para soportar los servicios computacionales que demandan los negocios de una organización, en el momento actual y a futuro. La predicción a futuro solo es posible a partir de la comprensión del estado actual; el proceso de análisis de desempeño nos brinda esta información y habilita la posibilidad de construir modelos para predecir cómo se comportaría la infraestructura ante diferentes cambios en la carga, en particular ante demanda más grande de recursos computacionales por crecimiento en los negocios de una organización.

Evaluación Comparativa y Análisis de Desempeño. La evaluación comparativa es la práctica de comparar una o varias métricas de desempeño de varias plataformas, o en este caso, infraestructuras. Es decir, necesitaríamos adelantar tareas de análisis de desempeño en cada una de las plataformas y posteriormente comparar los resultados. Esta práctica permite identificar cuál es la configuración o diseño que mejor responde a las demandas generadas por uno o varios servicios computacionales, habilitando así la posibilidad de cambiar o mantener una infraestructura.

Actividad 4-1: Deténgase un momento a revisar los conceptos estudiados.
1. Realice la siguiente lectura: https://www.sportperformanceanalysis.com/article/what-is-a-performance-analyst-in-sport
Responda las preguntas:
- ¿Cuál es el rol de un analista de desempeño en el contexto presentado? (¿Cómo contribuye al éxito de su organización?)
- ¿Cuál es el rol de un analista de desempeño en nuestro contexto? (¿Cómo contribuye al éxito de su organización?)

4.2 Objetivos del Análisis de Desempeño

Como se mencionó en la sección anterior, el análisis de desempeño nos permite comprender el comportamiento de un sistema. Esta sección plantea preguntas que pueden guiar a un analista de desempeño para comprender el comportamiento de un sistema.

1. ¿Cuáles son los objetivos del sistema?
2. ¿Cumple el sistema -como un todo- con los objetivos?
3. ¿Cuál es la relación entre los componentes?
4. ¿Cómo afecta cada componente el comportamiento del sistema como un todo? ¿Alguno afecta de forma adversa el cumplimiento de objetivos?
5. ¿Hay componentes que se hayan convertido en cuellos de botella? ¿Qué tanto afectan el cumplimiento de los objetivos?
6. ¿Es posible mejorar el desempeño de alguno de los componentes afinando la configuración?
7. ¿Se comporta el sistema de la misma forma a lo largo de un periodo de evaluación? (día, semana, mes)
8. ¿Hay algún componente que se convertirá pronto en un cuello de botella?
9. ¿Qué capacidad de crecimiento puedo soportar con el sistema en su estado actual?

Observe que un analista no puede responder directamente las preguntas 1 y 2. Los objetivos de la infraestructura dependen de las demandas generadas por los servicios que requiere una organización y estos objetivos son establecidos por las directivas. Sin embargo, un analista debe conocer dichos objetivos para traducirlos en medidas desempeño y poder evaluar si una infraestructura cumple o no cumple con ellos.

Después de conocer los objetivos, el analista debe buscar entender cómo se comporta cada uno de los componentes, cómo interactúan y cómo influyen en el cumplimiento del objetivo del sistema. Las preguntas 3 a 7 están relacionadas con este aspecto.

Finalmente, la información recopilada permite al analista construir un modelo y predecir el comportamiento del sistema ante variaciones en la demanda. Las preguntas 8 y 9 están relacionadas con este aspecto.

4.3 Etapas del Análisis

Para entender el comportamiento de un sistema es necesario establecer los indicadores que usaremos para medir el desempeño del sistema, recopilar datos, analizar los datos y finalmente presentar una síntesis a quienes son responsables de tomar decisiones, posiblemente con un conjunto de recomendaciones o alertas.

4.3.1 Indicadores

Los indicadores son criterios usados para medir el desempeño de los componentes de un sistema, tanto a nivel del software, como a nivel del hardware que lo soporta. La siguiente tabla presenta indicadores de hardware y software.

Hardware y Sistema Operativo	Software
% uso del procesador	Número de transacciones
% uso del disco	Número de procesos en ejecución
Tiempo promedio de respuesta del disco	Tiempo promedio de respuesta por transacción
# operaciones i/o del disco	Espacio requerido para almacenamiento
# de conexiones de red	

Además de representar el comportamiento de un componente del sistema, los indicadores nos permiten identificar relaciones entre componentes del sistema y predecir algunos aspectos del comportamiento de un sistema. La Figura 4-1 ilustra esta idea; presenta visualmente las medidas de dos indicadores del sistema: el porcentaje de uso del procesador y el número de procesos Apache ocupados (atendiendo clientes u operaciones).

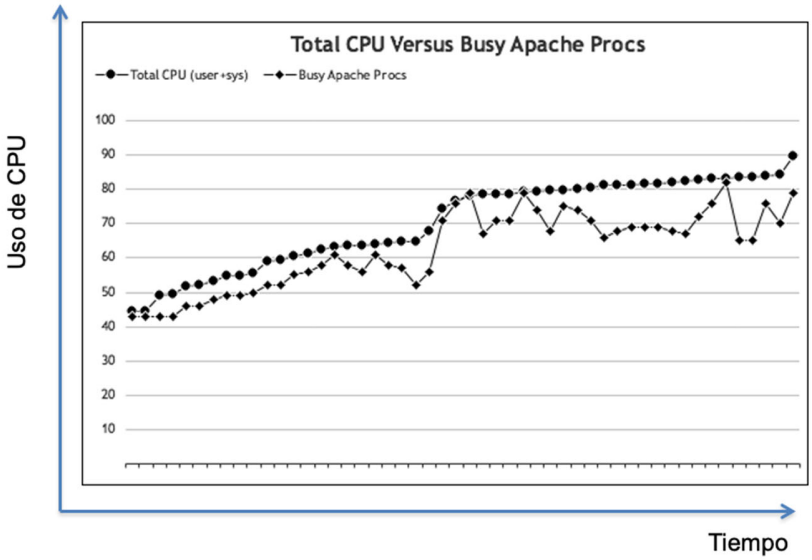


Figura 4-1. Porcentaje de uso de CPU vs. Procesos Apache ocupados. Tomado del libro The Art of Capacity Planning, 2nd edition, A. Kejariwal, J. Allspaw. 2018.

Analizando las medidas, encontramos que los dos indicadores están correlacionados. Con base en esta correlación podemos predecir que el porcentaje de uso de la CPU crecerá con el número de procesos Apache y por tanto un crecimiento de la demanda en el servicio llevará al procesador a convertirse en un cuello de botella.

4.3.2 Recolección

Esta sección presenta primero la definición de términos que usaremos para describir el proceso de recolección de medidas de los indicadores.

- Monitor: Herramienta usada para observar la actividad de un indicador de comportamiento de un sistema.
- Evento: Cambio en el estado de un sistema. Los monitores registran los eventos/cambios en el estado de un sistema.
- Rastro: Registro de los eventos.
- Resolución: Precisión de la información que se observa.
- Sobrecarga: Medida de la sobrecarga generada por el monitor. Todo monitor genera sobrecarga perturbando ligeramente el sistema. El monitor no debe alterar significativamente el comportamiento del sistema.

Los monitores son herramientas implementadas para observar un indicador de desempeño (o un conjunto de indicadores), registrando en un log el estado del indicador ante cambios o periódicamente. Es decir, los monitores contribuyen a la recolección de datos.

Carga real vs. Carga sintética. La recolección de datos de los indicadores se puede realizar en dos ambientes: con carga real o con carga sintética. Diremos que trabajamos con carga real si los monitores toman las medidas mientras el sistema corre en un ambiente real de producción.

Por otro lado, el rastro generado por los monitores se puede usar para construir modelos del sistema y generar carga de forma controlada, este tipo de carga sería sintética. La carga sintética es útil cuando un analista quiere repetir una evaluación o medir diferentes indicadores de desempeño ante variaciones de configuración del sistema para una misma carga.

Entre los modelos que permiten generar carga sintética podemos mencionar algunas distribuciones de probabilidad. Como los ejemplos que se mencionan a continuación. Recuerde que las distribuciones de probabilidad describen la probabilidad que una variable aleatoria tome un valor x.

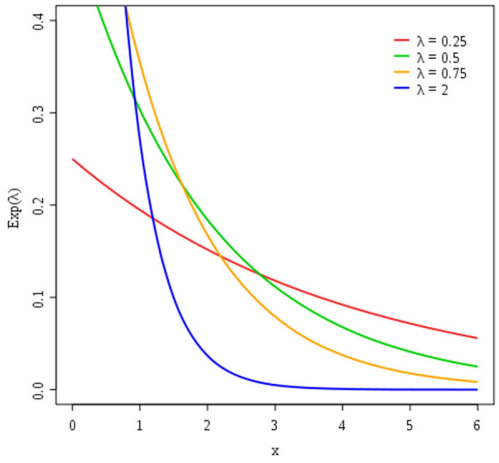
- Bernoulli permite representar el comportamiento de un sistema con dos estados, con la probabilidad de éxito como parámetro de configuración; si la probabilidad de éxito es p, entonces la probabilidad de falla es 1 - p. Por ejemplo: que un sistema esté up|down, que un paquete tenga errores al arribar si|no, que un componente en un sistema esté disponible si|no, etc.

$$E(X) = p$$
$$Var\ (X) = p(1-p)$$

- La distribución exponencial es muy usada en teoría de colas, permite representar el comportamiento de un sistema con colas de espera o manejo de eventos. De forma más precisa, permite modelar el tiempo entre la llegada de eventos, por ejemplo, tiempo entre llegada de trabajos a un sistema, llegada de

pedidos a un dispositivo o tiempo entre fallas de un dispositivo. El tiempo entre llegadas de trabajos a un sistema es exponencial con valor esperado 1/l.

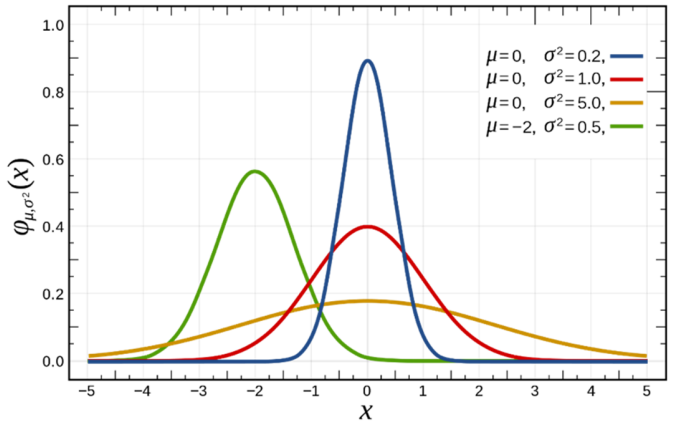
$$E(X) = 1/\lambda$$
$$\text{Var (X)} = 1/l^2$$



- La distribución normal permite modelar el comportamiento aleatorio de una variable, con un valor esperado μ y una desviación estándar σ como parámetros de configuración.

$$E(X|a < X < b) = \mu - \sigma^2 \frac{f(b) - f(a)}{F(b) - F(a)}$$

f es la función de densidad
F es la función acumulativa



Actividad 4-2: Deténgase un momento a revisar los conceptos estudiados.
¿Qué indicadores usaría para comparar el desempeño de los siguientes sistemas?:
1. Dos sistemas de archivos
2. Dos discos (uno tradicional vs. Uno de estado sólido)
3. Un algoritmo de cifrado simétrico vs. Un algoritmo de cifrado asimétrico

4.3.3 Análisis

Recordemos que el objetivo del proceso de análisis de desempeño es comprender el comportamiento del sistema para predecir cómo puede evolucionar en el futuro. La predicción se basa en la validez de los datos

recolectados; si los datos no representan el comportamiento de un sistema actual, entonces las conclusiones de un analista tampoco van a corresponder al comportamiento del sistema en el futuro.

Como consecuencia, debemos primero evaluar la validez de los datos. Debemos entender que las medidas recolectadas por un monitor son un conjunto de observaciones, pero no incluyen todos los estados de un sistema. En otras palabras, estamos construyendo una muestra de una población; no conocemos las características de toda la población, solamente las características de la muestra. Es importante que la muestra represente a la población para tomar decisiones apropiadas.

Las medidas que toman los monitores son equivalentes a los valores que toma una variable aleatoria en probabilidad y estadística. Sobre los valores recopilados podemos evaluar la desviación estándar, una medida de la dispersión de los valores con respecto a un valor central. En nuestras medidas buscamos valores que tengan una desviación estándar baja solo así podemos sintetizarlos para tomar decisiones. Si la desviación estándar es muy alta, el analista debe verificar que no haya errores en la implementación y usarlos de forma cuidadosa porque puede no ser posible sintetizar la información para predecir el comportamiento futuro del sistema. Para sintetizar los datos, los valores más usados son la media aritmética, la mediana y la moda; los tres en conjunto.

La media aritmética sola es muy usada, pero no es representativa del comportamiento de un sistema si la desviación estándar no es baja. Así que es importante identificar si este valor representa de forma apropiada el comportamiento de un sistema.

Si los datos son válidos, entonces podemos usarlos para construir modelos y predecir el comportamiento de un sistema en el futuro. Una herramienta útil en este caso es la regresión lineal: con base en las medidas tomadas para un indicador y en su evolución a lo largo de un periodo de tiempo, la regresión lineal nos permite calcular el valor que el indicador tomará en un punto específico en el futuro, es decir, el comportamiento del sistema en el futuro. A partir de esta predicción podemos generar alertas o hacer recomendaciones a quienes toman decisiones en un negocio.

Actividad 4-3: Deténgase un momento a revisar los conceptos estudiados.

1. Suponga que tiene un monitor que mide el tiempo de respuesta de una aplicación. El monitor generó un rastro con los siguientes valores:
5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 40, 30 (todos en seg.)
Suponga que el analista debe presentar un informe que describa de forma sintetizada el comportamiento del sistema. ¿Qué valores usaría usted y por qué?

2. ¿Cómo puede usar un analista de desempeño las distribuciones de probabilidad? ¿Para qué sirven?

4.3.4 Presentación

Las gráficas son una herramienta de gran ayuda porque permiten presentar gran cantidad de información de forma sencilla; tanto integrantes de un equipo técnico como integrantes de una junta directiva pueden entender el contenido.

Las recomendaciones para construir estas gráficas son sencillas:

- Muestre los ejes y sus etiquetas
- Las etiquetas son claras y concisas
- Muestre las escalas y las divisiones para todos los ejes
- En algunos casos es útil presentar el mínimo y el máximo
- El número de curvas en la gráfica debe ser razonable
- Todos los gráficos deberían usar la misma escala. Si no lo hace, debe indicarlo explícitamente porque puede ser confuso.
- Asigne una etiqueta a cada curva
- Presente el significado de todos los símbolos en el gráfico
- Indique las unidades de medida
- Recuerde incluir un título para el gráfico
- El titulo debe ser claro y conciso
- El gráfico debe generar valor. Si no genera información adicional entonces no vale la pena incluirlo.
- El reporte que incluye la figura debe hacer referencia a la misma. No debe haber figuras aisladas, sin conexiones con el texto.

4.3.5 Errores comunes

Al hacer el análisis es posibles equivocarse. Tenga en cuenta los siguientes errores comunes y evítelos:

1. Ignorar la sobrecarga del monitor. El efecto de un monitor en el sistema no debería pasar cierto umbral. Al afectar el comportamiento del sistema, un monitor también afecta las variaciones en los resultados.
2. Olvidar evaluar la validez de los datos.
    a. Los datos recolectados para el modelaje de cargas no son, por definición, completos. Es necesario evaluar su validez, con métodos estadísticos, por ejemplo.
    b. Revisar si los datos recolectados tienen sentido. Si aparecen valores extraños buscar la razón.
    c. Correr múltiples experimentos y revisar validez solamente al final. Es importante revisar detalladamente los datos generados inicialmente para evaluar la validez de los mismos, así como la correcta ejecución del monitor.

4.4 Introducción a la Planeación de capacidades

Este documento presenta una introducción al problema de dimensionar la solución de Tecnología de Información (TI) adecuada para un cierto problema. Es decir, vamos a intentar aproximarnos a identificar la mejor solución computacional para ejecutar una o un conjunto de aplicaciones.

Este curso se concentra en dos de las dimensiones que involucran una solución computacional: procesamiento y almacenamiento. La tercera es comunicaciones, pero para esa está el curso de Infraestructura de Comunicaciones. Por lo tanto, el foco de esta parte del curso está en cómo decidir la configuración ideal de un computador en cuanto a procesamiento y almacenamiento para responder a unas ciertas necesidades.

Este proceso, conocido como *capacity planning* busca entonces responder a preguntas como ¿Qué procesador debe tener el computador que corre la nómina de mi empresa? ¿Cuánta memoria debo ponerle al servidor de correo? ¿Cuántos procesadores necesita el servidor que ejecuta banner? ¿Cómo debe ser la configuración de la solución para que Bloque Neón funcione bien? En el poco tiempo que nos queda no podremos llegar a este nivel de detalle, pero sentaremos las bases para poder empezar a esbozar las respuestas.

Hasta ahora en la carrera, hemos definido que una aplicación funciona bien si hace lo que tiene que hacer (requerimientos funcionales). De hecho, así se ha calificado hasta ahora. Ha habido algunas incursiones en temas de desempeño (p. ej. complejidad), pero todas las soluciones hasta este momento han venido de la aplicación misma. Por ejemplo, si queremos que un programa ordene unos datos más rápido, entonces buscamos un algoritmo con una menor complejidad e intentamos diferentes optimizaciones para acelerar un poco ese procesamiento.

Existe otro componente, igualmente importante a la hora de ver si una aplicación funciona bien: que cumpla con los requerimientos no funcionales (atributos de calidad), algunos de los cuales se han mencionado previamente en este y otros cursos: usabilidad, seguridad, mantenibilidad, etc. En las siguientes semanas trabajaremos con aquellos que dependen directamente de la infraestructura que ejecuta la aplicación, pero de nuevo, solamente en las dimensiones de procesamiento y almacenamiento.

El contenido de este documento está basado en tres fuentes principales: “The Art of Capacity Planning, 2nd edition” de John Allspaw y Arun Kejariwal, “Guerrilla Capacity Planning” de Neil J. Gunther y “High Availability and Disaster Recovery” de Klaus Schmidt.

4.5 Proceso de planeación

Capacity planning (planeación de capacidades) busca determinar la infraestructura de TI (procesamiento, almacenamiento, comunicaciones, etc.) necesaria para prestar eficientemente un servicio a lo largo del tiempo a un cierto nivel de efectividad considerado satisfactorio.

Esta definición es muy importante porque nos aclara las características que debe tener una infraestructura de TI: debe ser eficiente, es decir el costo de la solución debe ser el mínimo posible, debe ser una solución válida en un marco de tiempo y debe respetar unas condiciones prestablecidas. Esto último hace referencia entonces a que se debe cumplir con ciertos requerimientos: los que hacen que una solución se considere buena. Es decir, ahora diremos que una aplicación funciona bien si hace lo que debe hacer (requerimientos funcionales) y si lo hace cumpliendo con unos requerimientos no funcionales o atributos de calidad.

La planeación de una infraestructura se focaliza entonces en los requerimientos no funcionales, los requerimientos funcionales se dan por resueltos a nivel de la aplicación. Vamos a suponer que todo aquello que debe ser hecho desde la aplicación ya está hecho (optimización de software), nos queda entonces encontrar la infraestructura de TI adecuada para cumplir con los requerimientos no funcionales. Y al igual que para los requerimientos funcionales, en donde utilizamos un proceso para mejorar nuestras posibilidades de éxito, aquí utilizaremos un proceso para que se pueda validar el cumplimento de los requerimientos no funcionales por parte de una infraestructura. Como en todo proceso de ingeniería, lo que es muy importante es que podamos

justificar todas las decisiones que aquí se tomen. Esto es, si al final se propone una infraestructura como solución, debemos poder explicar por qué se seleccionaron ciertos componentes o configuraciones.

Este proceso puede ser tan complejo como se quiera y entre más exacto el resultado buscado, más complejo y costoso será el proceso. Aquí, y en el mundo real, se toma una aproximación muy pragmática para evitar caer en la tentación de hacer un proceso muy complejo donde el costo del proceso sobrepasa el beneficio obtenido. Al final, todo hay que valorarlo y necesitamos asegurar un buen balance entre lo que nos vamos a ahorrar por lograr encontrar la infraestructura óptima para una solución de TI y lo que nos cuesta (tiempo, esfuerzo, dinero) encontrar esa solución. ¡No vale la pena gastarme $10 en un proceso para ahorrarme $1 en una infraestructura!

Nuestro proceso tendrá 3 pasos:

1. Definir el nivel de servicio requerido: precisar los servicios, sus alcances y características
2. Analizar la capacidad actual: cómo responde mi computador actual
3. Hacer prospectiva: cómo van a evolucionar las cosas en una ventana de tiempo

Como veremos a continuación, con estos 3 simples pasos vamos a poder determinar la infraestructura que cumple con la definición que vimos al principio de esta sección. Esta es la labor del ingeniero de infraestructura y para ello, como veremos, debe trabajar muy de la mano de la alta gerencia de la organización.

4.5.1 Definir el nivel de servicio requerido

Lo primero que tendremos que aclarar es de qué requerimientos estamos hablando, al menos de cuáles nos vamos a ocupar en estas semanas. Un requerimiento es algo que alguien quiere. En el caso de los no funcionales, hace referencia a las características que se desea tenga la prestación de un servicio. De nuevo, supongamos que el servicio sabe hacer lo que le pedimos y está razonablemente bien implementado. Y como dijimos antes, nos interesan aquellas características en las que hay una responsabilidad directa de la infraestructura. Pensemos por un momento en el caso de Spotify. ¿Qué características quiere usted que tenga el servicio de Spotify y cuáles de ellas dependen de la infraestructura en que se ejecute?

Para responder a la anterior pregunta debemos empezar por entender bien cuál es el servicio que presta Spotify para poder empezar a asociarle características y ver si esas características se ven influenciadas por la infraestructura en la que se ejecuta Spotify.

Actividad 4-4: Identifique 2 servicios que preste Spotify. Enumere las características que usted espera de ese servicio y diga si la infraestructura es relevante para garantizar su expectativa sobre el requerimiento.

Uno de los elementos que surge al hacer este análisis es la necesidad de entender la infraestructura de Spotify. Como la gran mayoría de servicios a los que accedemos hoy en día (empresariales y personales), una generalización válida es suponer una infraestructura de 3 capas: cliente, servidor y almacenamiento, todos conectados a través de alguna red de comunicaciones. La Figura 4-2 nos muestra una visión general de este tipo de infraestructuras.

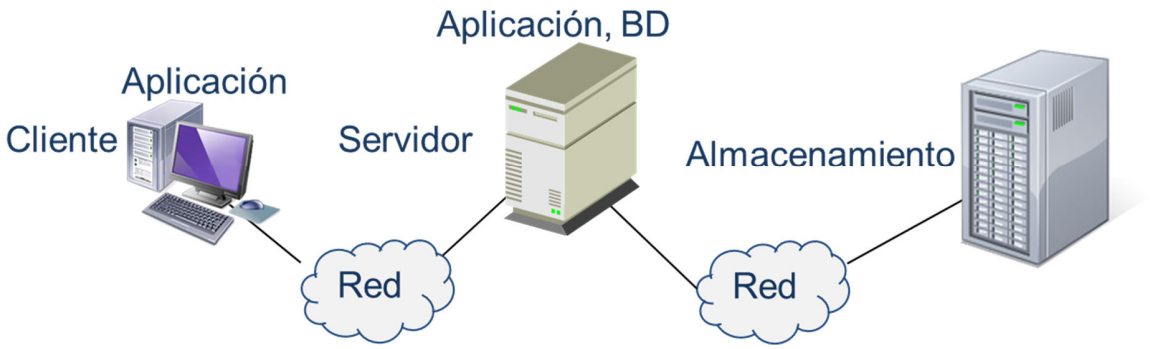


Figura 4-2. Arquitectura general de un servicio Web

Entender todos los elementos que hacen parte de la infraestructura es importante porque para cumplir un requerimiento es muy posible que varios de ellos (o todos) están involucrados. Debemos entonces distinguir los requerimientos globales (aquellos que involucran varios componentes) de los individuales (aquellos que dependen de solo un componente) porque para cumplir los requerimientos globales, habrá que identificar los individuales. En el ejemplo de Spotify, si queremos que una canción empiece a sonar antes de 2 segundos después de haberla solicitado, seguramente tendremos que imponer un requerimiento al almacenamiento para que no tarde más de 200ms transmitiendo la canción del dispositivo a la interfaz, a la base de datos (BD) para que la búsqueda de la canción no tarde más de, digamos, 100ms, al servidor para que atienda la solicitud en menos de 50ms, a la red para que transmita en menos de …., etc. Y si somos el proveedor del servicio, tendremos también que reconocer que hay cosas fuera de nuestro control sobre las que haremos supuestos pero que, si no se cumplen, no podemos hacernos responsables. Por ejemplo, esperamos que la solicitud desde el cliente al servidor llegue en menos de 300ms, pero si el teléfono desde donde se está solicitando la canción es muy viejo o está muy cargado, o la red por la que se está conectando está muy lenta, es posible que el requerimiento global termine por no cumplirse.

De todos los posibles requerimientos no funcionales de un servicio, hay 4 que son altamente dependientes de la infraestructura que ejecuta el servicio: desempeño, capacidad, escalabilidad y disponibilidad. Otros posibles requerimientos que por lo tanto no serán abordados en esta parte del curso son: confidencialidad, integridad, autenticidad (estos 3 ya los vimos), usabilidad, mantenibilidad, administrabilidad, interoperabilidad, económicos, legales, etc.

La labor nuestra (ingenieros de infraestructura) no es definir los requerimientos, es identificarlos (cuáles son aquellos que afectan la experiencia de los usuarios) y cuantificarlos con la alta gerencia. Esto es muy importante porque los valores que se asignen a estos requerimientos tendrán un altísimo impacto en el costo de la solución. Solo la alta gerencia sabe qué tipo de experiencia quiere ofrecer a sus usuarios y es nuestra labor orientarla en los costos que ese nivel de servicio acarrea.

Revisemos en detalle, aquellos que nos competen y que deberían por lo tanto considerarse al planear una infraestructura, en cada uno de sus ejes (procesamiento, almacenamiento y comunicaciones). Recuerde que aquí no estudiaremos el eje de comunicaciones. Ser capaz de identificar el tipo de requerimiento es absolutamente clave porque la solución se diseña en función de este tipo.

- Capacidad: carga que debe soportar el sistema. Volumen de información procesada, almacenada o enviada por unidad de tiempo.

- Desempeño: tiempo de respuesta. Velocidad con la que se desempeña la infraestructura
- Escalabilidad: adaptabilidad ante variabilidad en la carga
- Disponibilidad: accesibilidad ininterrumpida a la información (o con mínimos trastornos)

Actividad 4-5: Clasifique, de acuerdo con estas categorías, los requerimientos identificados en la actividad 1. Asegúrese de identificar al menos 1 requerimiento de cada tipo.

Un elemento importante de los requerimientos es que estos deben estar expresados en términos que los usuarios entiendan y, eventualmente, puedan comprobar. De nada vale un requerimiento si no es visible para los usuarios, al fin y al cabo, son ellos los que esperan que se cumpla. Esto implica que los requerimientos tienen que estar expresados en términos del objeto del servicio (del negocio). Ejemplos válidos de requerimientos son: "debe poder reproducir 1M de canciones por hora", "la canción debe empezar a sonar antes de 2 segundos de haberse solicitado", "los viernes de rumba debe poder atender un 50% más de usuarios", "es indispensable que el sistema no falle más de dos veces al año", etc.

A continuación, revisaremos en detalle cada uno de ellos.

4.5.1.1 Capacidad

Busca determinar la capacidad de procesamiento necesaria para una cierta carga de trabajo. Hace referencia a lo que debe ser capaz de procesar/almacenar/comunicar un computador en un momento dado o por unidad de tiempo. Como ya dijimos, debe estar expresada en unidades relativas al objeto del servicio.

Por ejemplo, al decir que debe reproducir 1M de canciones por hora, habla de la capacidad que debe tener la infraestructura porque significa que debe contar con los suficientes recursos para que eso suceda. Se identifican dos tipos de medidas de la capacidad:

- **Productividad**, que se refiere al número de tareas despachadas (o terminadas) por unidad de tiempo. Por ejemplo, *la BD debe procesar 5000 registros por minuto*. O, *el disco debe transmitir 30 canciones por segundo desde el dispositivo hasta la interfaz*. Note que para este último tendremos que conocer el tamaño promedio de una canción y el requerimiento se traducirá a bytes/seg. Pero que la interfaz de un disco tenga una capacidad de 500MB/s no es algo que el usuario pueda entender y es él quien al final se verá impactado por el cumplimiento o no de un requerimiento.

- **Carga**, número de tareas activas al mismo tiempo. Este requerimiento habla de capacidad porque determina lo que debe ser capaz una infraestructura de proveer en un momento dado. Por ejemplo, *el sistema debe permitir que se conecten hasta 50M de usuarios en un momento dado*, habla de la capacidad del sistema. *Se debe poder almacenar 200M de canciones* es un requerimiento de capacidad de almacenamiento.

Retomando el caso de Spotify, ¿quién debe definir cuántas canciones tendrá el catálogo (y por lo tanto cuántas hay que almacenar, generando así un requerimiento de carga)? Claramente, esto no es una responsabilidad del área de Tecnología de Información (TI), es la alta gerencia quien debe definir este valor y el rol de TI es orientar la decisión con los elementos relevantes de infraestructura que se ven afectados. Por ejemplo, ilustrando sobre diferentes alternativas de almacenamiento (SSD o discos duros), sus costos, sus capacidades o su desempeño.

4.5.1.2 Desempeño

El requerimiento de desempeño está asociado al tiempo de respuesta. Cuánto se demora en procesar una cierta solicitud, y como vimos antes, puede referirse a un requerimiento global o parcial sobre una parte de la infraestructura. De hecho, la manera de asegurar que se cumpla con un requerimiento global es establecer requerimientos individuales sobre los distintos componentes involucrados en resolver ese requerimiento global.

Es posible que al principio se confundan algunos requerimientos de desempeño con requerimientos de capacidad. Para evitar esto piense siempre en que los de capacidad (productividad) hablan de cosas que pasan por unidad de tiempo. En desempeño solo medimos el tiempo en absoluto. Por ejemplo, resolver 5000 solicitudes en 2 horas es desempeño, mientras que atender 2500 solicitudes por hora es un requerimiento de capacidad. *Hacer el backup en menos de 4 horas* o *entregar la respuesta antes de 5 minutos* son también ejemplos de requerimientos de desempeño.

De nuevo con Spotify, decidir si una solicitud de un usuario se debe responder en 2 o en 3 segundos máximo es una decisión de negocio, no de TI. Por eso decíamos al principio, que en esta primera etapa del *capacity planning*, hay que tener una estrecha colaboración entre TI y la alta gerencia.

4.5.1.3 Escalabilidad

La escalabilidad se define por la variabilidad en la carga. Esto es, la carga a la que se somete un sistema no es homogénea por muchas razones. Pensemos en banner. Seguro la carga (número de estudiantes que lo consultan en un momento dado o durante una fracción de tiempo) es mucho menor el 31 de diciembre que la semana de publicación de notas. Por otro lado, si pensamos en una ventana de tiempo mayor (2 – 3 años), uno podría pensar que para esa época tendremos más estudiantes y cursos, la carga será de nuevo distinta y la solución debe prever este crecimiento (recuerde que la definición habla de una solución a lo largo del tiempo).

La variabilidad en la carga entonces habla de crecimiento, decrecimiento, picos, momentos de poco o ningún uso. En general las soluciones las debemos diseñar para que puedan lidiar eficientemente con todo esto, el requerimiento de escalabilidad nos dice los límites que estamos dispuestos a contemplar en el diseño. Por ejemplo, diremos que una solución, de acuerdo con el crecimiento esperado del negocio, será válida por solo 2 años; o, por el contrario, si no se espera mucho crecimiento, que la solución se mantendrá vigente unos 5 años. O que una solución tiene previstos unos picos en tales fechas, pero acotaremos el tamaño de esos picos.

La escalabilidad expresa entonces la relación entre una línea base de carga y una carga especial. Decir entonces que *los fines de mes la carga del sistema es el doble que en los otros días*, sería expresar un requerimiento de escalabilidad. Pedir que el servidor de banner tenga una capacidad 250% más grande la semana de inscripción de cursos, sería otro requerimiento de este tipo.

4.5.1.4 Disponibilidad

El requerimiento de disponibilidad está asociado a una medida de la frecuencia o del tiempo que un servicio o componente está disponible. Este requerimiento nos dice entonces qué porción del tiempo un servicio está disponible, y por eso frecuentemente lo encontraremos expresado como un porcentaje. Se suele medir como porcentaje de tiempo en que un servicio o componente del sistema está disponible:

$$Disponibilidad = Td/(Td + Tnd)$$

Td = Tiempo disponible
Tnd = Tiempo no disponible

Dos métricas definen particularmente la disponibilidad: la confiabilidad (definida por el fabricante) y la mantenibilidad que depende del tipo de soporte que contrate el comprador.

- Confiabilidad (Reliability): probabilidad de que un sistema se mantenga funcionando continuamente. Se suele expresar por:
  - *MTBF* = Mean Time Between Failures (o MTTF)
- Mantenibilidad (Serviceability**)**: medida de la facilidad para reparar un servicio. Se puede expresar por:
  - *MTTR* = Mean Time To Repair
- Disponibilidad (Availability):
  - *Disponibilidad* = *MTBF* / (*MTBF* + *MTTR*)

Ejemplos de requerimientos de disponibilidad son: el servicio *debe estar disponible 99% al año*, el servicio debe *operar 7x24 con máximo 1 hora de indisponibilidad a la semana*, el servicio debe *ofrecer una disponibilidad de 99,99% entre semana y de 99% los fines de semana*.

Como es muy frecuente encontrar porcentajes asociados a la disponibilidad. A continuación, se presenta el equivalente entre porcentaje y el tiempo admisible de indisponibilidad para que podamos tener una mejor comprensión de lo que significan esos porcentajes.

Disponibilidad	Tiempo de indisponibilidad (al **año**)
90.0%	36 días, 12 horas
95.0%	18 días, 6 horas
99.0%	87 horas, 36 minutos
99.5%	43 horas, 48 minutos
99.9%	8 horas, 45 minutos, 36 segundos
99.99%	52 minutos, 33 segundos
99.999%	5 minutos, 15 segundos
99.9999%	32 segundos

Actividad 4-6: Visite estas páginas Web y compare las características de los dos tipos de disco que allí se describen. Ponga especial cuidado en aquellas características que responden a los requerimientos aquí presentados. Si tuviera que decidir entre comprar un disco u otro, ¿cuáles serían sus criterios de decisión?
https://toshiba.semicon-storage.com/us/product/storage-products/enterprise-hdd/al14sxbxxex.html
https://toshiba.semicon-storage.com/us/product/storage-products/client-hdd/mq04abfxxx.html

4.5.2 Analizar la capacidad actual

Si queremos saber qué infraestructura necesitamos, lo primero es saber qué tan bien o mal estamos con lo que tenemos. El sentido común nos debe guiar en este proceso, si en mitad de una fiesta nos damos cuenta de que necesitamos cerveza, primero identificamos los requerimientos (¿cuántas personas hay en la fiesta?, ¿qué tanto están tomando?, ¿cuánto falta para que termine la fiesta?, etc.) y luego, antes de salir a comprar, miramos cuánta cerveza queda en la nevera. Con estos datos, estimamos cuánto nos va a durar la reserva que tenemos en la nevera. ¿1 minuto, 1 hora? En Tecnología, debemos hacer lo mismo, revisamos lo que tenemos antes de salir a comprar más tecnología. Y, como en el caso de la cerveza, el éxito del análisis depende de hacernos las preguntas correctas y de tener los indicadores adecuados.

Una vez identificados los indicadores, que deben estar directamente relacionados con los requerimientos, el siguiente paso es medirlos para poder analizarlos. Dos nuevas preguntas surgen entonces ¿Qué medir? Y ¿Cómo realizar esas medidas? Se debe definir si necesitamos datos globales o locales, los períodos de tiempo en los que es relevante medir, las condiciones que se deben considerar y la relación entre lo que mido y las acciones de los usuarios.

Como vimos en Análisis de Desempeño, el monitoreo es el proceso continuo para recopilar estas mediciones. Lo que buscamos es poder construir un modelo que represente nuestro sistema, identificando la respuesta del sistema ante variaciones de la carga, y que nos permita identificar eventuales cuellos de botella y su impacto para priorizar su atención.

Tomar los datos correctos y con la frecuencia adecuada es un requisito para el análisis, pero el análisis mismo tiene que ver con la interpretación que hagamos de esos datos. Una tentación que debemos evitar, por ejemplo, es basarnos exclusivamente en los promedios obtenidos. El promedio de un consumo no es un buen indicador de la capacidad que debemos tener dispuesta para atender dicho consumo. Como vimos anteriormente, la varianza, la desviación estándar, la moda y otros valores similares deben ser considerados.

El monitoreo mide datos brutos del sistema: porcentaje de uso de CPU, espacio disponible en disco, porcentaje de memoria libre, etc. El análisis parte de ser capaces de relacionar estos datos con el consumo que realizan los usuarios de nuestro sistema. Esto es, por ejemplo, cómo varía el porcentaje de CPU utilizado en función del número de canciones que se solicitan en Spotify, o cómo varía el espacio disponible en disco según el número de canciones que estamos almacenando.

4.5.2.1 Rendimiento vs carga

Una vez logramos establecer estas relaciones, lo que buscamos es entender los puntos de inflexión que hacen que el comportamiento del sistema cambie fuertemente ante cambios leves en la utilización. La relación entre carga y rendimiento del sistema está en la base de todo análisis. Rendimiento tiene que ver con la capacidad de entregar respuestas ante las solicitudes que recibe un sistema (cantidad de requerimientos atendidos cumpliendo su requerimiento de desempeño). Así, cuando un sistema recibe pocas solicitudes, su rendimiento es igual a la carga que recibe. Pero a medida que aumentamos la carga, se llegará a un punto en que su rendimiento empezará a estancarse. Estos puntos de quiebre se conocen como codos y se ilustran en la siguiente figura. El primer codo corresponde al punto en el que si aumentamos la carga el sistema ya no mejora su rendimiento de manera lineal, esto es, empezamos a experimentar demoras en la respuesta de las peticiones

individuales. El segundo codo, corresponde al punto donde el sistema se encuentra saturado y por lo tanto si continuamos agregando carga, lo único que haremos es empeorar la situación.

Este comportamiento se explica porque al principio hay recursos disponibles para atender requerimientos, pero a medida que estos aumentan se llega a un punto en el que hay que compartir estos recursos entre varios requerimientos (primer codo). Y si seguimos incrementando la carga, llegaremos a un punto en donde el sistema invertirá los recursos en recibir los requerimientos, no en atenderlos. Eso genera el segundo codo en donde empezamos a perder rendimiento.

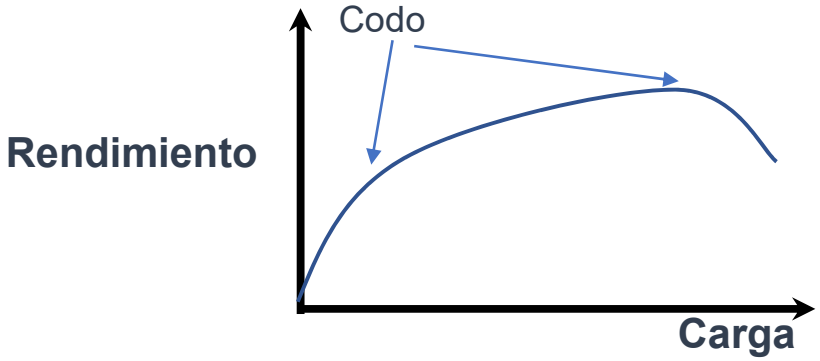


Figura 4-3. Relación entre carga y rendimiento

El rendimiento de un sistema está altamente relacionado con el requerimiento de desempeño. Los codos de la gráfica de rendimiento coinciden con los codos de la gráfica del tiempo de respuesta. En la Figura 4-4 vemos que en efecto el primer codo de rendimiento coincide con un cambio en la pendiente de la curva del tiempo de respuesta. El requerimiento de desempeño es un valor que el tiempo de respuesta no debe superar (cf. sección 2.2), la Figura 4-4 nos ayuda a entender entonces en qué momento la carga que está recibiendo el sistema haría que dejara de cumplir con el requerimiento de desempeño especificado. Este es el límite de nuestra infraestructura. Debemos notar que el límite de la infraestructura puede suceder con una carga inferior a la del segundo codo del rendimiento. Esto es normal, porque el límite de la infraestructura lo definimos con respecto a los requerimientos del negocio, no a las capacidades de la infraestructura.

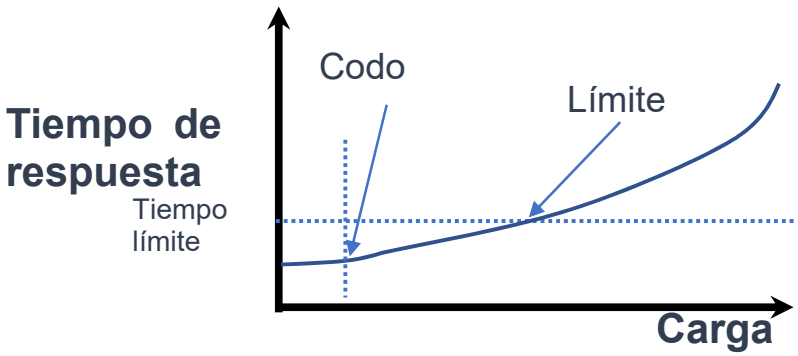


Figura 4-4. Relación entre carga y desempeño

Los sistemas operativos desglosan el tiempo de respuesta en dos elementos: el tiempo del usuario (t_u) que corresponde al tiempo consumido por el procesador ejecutando las instrucciones del programa y el tiempo del sistema (t_s) que corresponde al tiempo que se consume ejecutando llamados al sistema (cosas como el fork(), los fallos de página de la memoria virtual, las operaciones de entrada salida, etc.). El tiempo total o de CPU que consume un proceso es la suma de t_u+t_s, pero este tiempo no corresponde en realidad al experimentado por el usuario. El tiempo real (t_r) o también llamado tiempo de reloj de pared es el tiempo que le toma al usuario obtener la respuesta, desde el momento que lanza la aplicación. Si bien el que nos importa para los requerimientos es t_r, t_u y t_s nos servirán de indicadores sobre la salud del sistema. Por ejemplo, un valor de t_s > 10% t_u, es un síntoma de problemas de sincronización!

Actividad 4-7: Piense en la relación entre las métricas t_u, t_s y t_r con respecto a lo visto anteriormente en el curso. ¿Cómo afectan los estados Listo, Dormido y Ejecutando estos tiempos? ¿Qué podría explicar que t_u+t_s << t_r? ¿Podría ser mayor?

Una de las reglas de oro al analizar el comportamiento de un sistema es que los distintos componentes alcanzan su nivel de saturación alrededor del 80%. Esto es de nuevo algo que funciona en el mundo real: cuando alcanzamos el 80% del consumo de nuestro paquete de datos de telefonía celular, debemos tener precaución para llegar a fin de mes. De la misma manera en tecnología, cuando alcanzamos el 80% del espacio ocupado en el disco, debemos prepararnos para ampliar la capacidad del disco o empezar a borrar cosas que ya no usamos. Y si el procesador está bordeando el 80% de utilización, debemos ir pensando qué vamos a hacer porque nos encontramos en un punto de saturación. Pero, a diferencia del disco en donde si compramos un nuevo disco de la misma capacidad que el actual, básicamente doblamos nuestra capacidad utilizable, poner otro procesador no siempre nos arroja los mismos beneficios. Dependiendo del recurso del que estemos hablando, tendremos que revisar el costo de la adición de recursos versus el beneficio, esto es, la **escalabilidad** de la infraestructura.

4.5.2.2 Rendimiento y carga en almacenamiento

En el caso de los discos, tenemos que recordar cómo es su operación. Al final del tema de sincronización, revisamos los diferentes elementos que componen un dispositivo de entrada/salida. La Figura 4-5 muestra cómo es esta operación para un disco magnético. Si queremos, por ejemplo, analizar el desempeño de este disco, debemos considerar la velocidad interna (entre el dispositivo y su controladora) y la velocidad externa (entre la interfaz y la memoria RAM de la máquina). Pero además de estas velocidades, debemos tener en cuenta que la controladora actúa como un servidor que recibe peticiones, es decir, una solicitud de lectura no sale del procesador central y llega directamente a la controladora para ser ejecutada. Por el contrario, como la velocidad externa es mayor que la interna, por aquello de que la electrónica es más rápida que la mecánica, es de esperar que en la controladora se formen colas de peticiones que deberán ser atendidas por el dispositivo a medida que se vaya desocupando. Entonces, el tiempo de respuesta de un disco es el tiempo de llegada de la petición a la controladora (despreciable), más el tiempo de la petición en la cola de la controladora (depende de la carga), más el tiempo de servicio del disco (en el que hay que considerar el desplazamiento del brazo, la latencia rotacional y el tiempo de transferencia interna), más el tiempo de transferencia entre la controladora y la RAM. Una vez el dato solicitado está en la RAM, podemos decir que ha terminado la operación de lectura solicitada inicialmente.

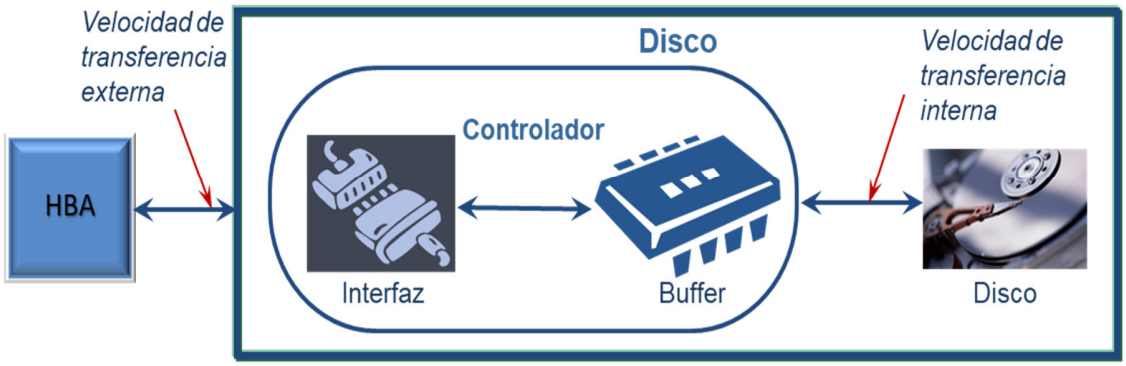


Figura 4-5. Arquitectura de conexión de un disco

Todos los tiempos mencionados anteriormente son fáciles de calcular, excepto el tiempo en la cola. Para esto hay que recordar la teoría de colas de cursos de probabilidad que nos da una herramienta muy útil para calcular este tiempo: la ley de Little. Esta ley dice que "el número medio de usuarios «*N*» que hay en un sistema (durante un tiempo determinado) es igual a la velocidad media «*a*» a la que entran en el sistema multiplicada por el tiempo medio «*R*» que permanecen dentro. Esta teoría, que se puede aplicar para calcular cuántas personas habrá en la fila de una caja para pagar un producto en un supermercado, nos permite también determinar el tamaño de la cola de una controladora del disco y con un poco de manipulación, saber el tiempo que pasaremos en esa cola y, por lo tanto, con las consideraciones anteriores, el tiempo de respuesta (desempeño) de un disco.

$$N = aR$$

Para calcular los tiempos de respuesta, podemos echar mano de otra ley asociada a la Ley de Little: la Ley de uso que nos habla del porcentaje de tiempo que se encuentra ocupado un servidor, el disco en este caso. La ley de uso estipula que "la utilización de un sistema «*U*» es igual a la velocidad media «*a*» a la que entran en el sistema multiplicada por el tiempo medio «R_s» de atención del sistema". Y R_s es conocido porque es un parámetro del hardware (movimiento del brazo o *seek time*, latencia rotacional y transferencia interna). Así, tenemos:

$$U = aR_s$$
$$R_a = 1/a$$
$$R = R_s/(1-U)$$

Actividad 4-8: Calcule el tiempo en cola en la controladora de un disco de una petición de lectura teniendo en cuenta que la controladora recibe 100 peticiones por segundo y que el tiempo de servicio del disco es de 8ms. ¿Cómo cambia su respuesta si cambiamos el disco por uno que tenga un tiempo de servicio de 4ms?

El análisis de la capacidad y de desempeño que hemos ilustrado en esta sección cubre implícitamente el requerimiento de escalabilidad pues este no es otra cosa que analizar los límites de la capacidad de la infraestructura actual, eventualmente, y dependiendo de la arquitectura de esta solución, analizando los costos de ofrecer una capacidad variable. Este punto será retomado más adelante cuando hablemos de soluciones específicas.

4.5.2.3 Disponibilidad

Para la disponibilidad entran en juego dos principios muy importantes: robustez y redundancia. La robustez promueve la simplicidad porque entre menos complejo sea un sistema, más robusto es: la complejidad es fuente de dificultades. En cambio, la redundancia favorece el contar con recursos adicionales para utilizar en caso de fallos. Son dos principios antagónicos porque la redundancia busca que haya más componentes en el sistema para poder remplazar los que fallen, mientras que la robustez que haya menos para que haya menos probabilidad de fallo. Una solución de TI debe buscar equilibrio entre estos dos principios.

El aspecto más importante de la disponibilidad es que es una decisión puramente económica y por tanto de negocio. Una vez más, la cooperación entre la alta gerencia y la dirección de tecnología es muy importante. Hoy en día, todos los negocios desearían que sus soluciones de TI les ofrecieran una disponibilidad del 100%. Solamente a la hora de ver los costos que eso implicaría es que aceptan considerar disponibilidades inferiores. Herramientas del estilo de un BIA (B*usiness Impact Analysis*) son muy útiles para ayudar a determinar los requerimientos de disponibilidad.

Actividad 4-9: Investigue cuál es la garantía de disponibilidad de servicios como Dropbox, Twitter, Spotify o Whatsapp

El análisis de disponibilidad no intenta determinar la disponibilidad actual que ofrece un sistema. Esto si bien es posible hacerlo (solo habría que medir el porcentaje de tiempo que la solución está funcionando en un periodo de tiempo), no sería más que una medida que no permite inferir el cumplimiento o no de un requerimiento. Que algo funcione o falle este mes no me garantiza nada sobre si volverá a fallar. Y la disponibilidad es un requerimiento a futuro, lo que quiero saber es cómo se va a comportar el sistema un tiempo más adelante.

Lo que buscamos con un análisis de disponibilidad es saber qué tan bien preparada está una solución para las posibles eventualidades que se puedan presentar. En ese sentido, la pregunta que buscamos responder es *¿cuál es la disponibilidad de un sistema dada la manera como está diseñado y construido?* Y la disponibilidad de un sistema depende de la disponibilidad de sus componentes y cómo estén articulados en la solución.
Para la articulación de los componentes en una solución existen dos posibilidades: ponerlos en serie o ponerlo en paralelo. Se dice que dos componentes de una solución están en serie si se necesita del funcionamiento de ambos para que la solución opere correctamente. Por el contrario, están en paralelo si para que la solución funcione es suficiente con que uno de los dos funcione correctamente. En este último caso, decimos que un componente está presente por redundancia. La Figura 4-6 ilustra este concepto.

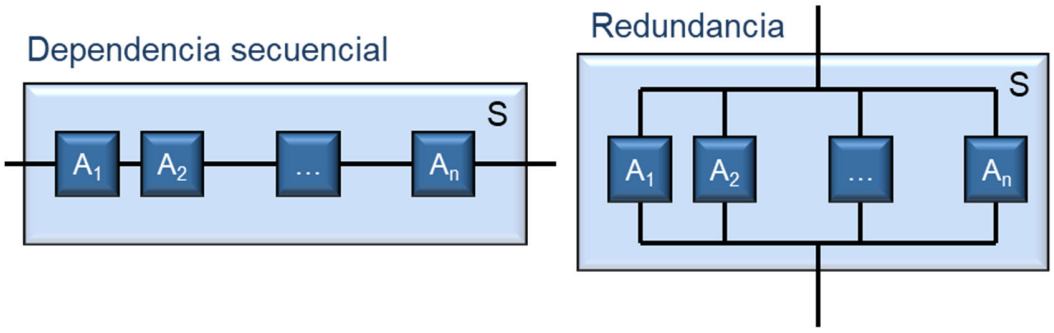


Figura 4-6. Arquitecturas secuenciales y paralelas

Las disponibilidades secuenciales y paralelas de un sistema S con dos componentes A y B se calculan de la siguiente manera:

$$P(Ssec) = P(A\ y\ B) = P(A)P(B)$$
$$P(Spar) = P(A\ o\ B) = P(A) + P(B) - P(A)P(B)$$

La disponibilidad de un componente debe ser reportada por el fabricante y normalmente está expresada como un porcentaje: el porcentaje de tiempo de un año que se espera que el dispositivo no falle. Dicho de otra manera, el complemento de la disponibilidad representa el tiempo en el cual el dispositivo no estará funcional, pero que como ha sido anunciado, no podríamos reclamar. Por ejemplo, si compramos un dispositivo que tiene una disponibilidad de 99.99%, esto querría decir que estamos dispuestos a aceptar que el dispositivo no esté disponible durante máximo 53 minutos al año.

Actividad 4-10: Calcule la disponibilidad de un sistema compuesto de 7 componentes en serie, cada uno con una disponibilidad de 99.99%. ¿Cuánto es el tiempo máximo tolerable de indisponibilidad de dicho sistema? Si cambiamos uno de los componentes (disminuimos la calidad, por ejemplo) por un componente con disponibilidad 99%, ¿cómo se altera la disponibilidad? ¿Y el tiempo de indisponibilidad al año?

Actividad 4-11: Considere un sistema compuesto de un solo equipo con disponibilidad de 99%. ¿Cómo cambia la disponibilidad del sistema si compramos otro equipo igual para que funcione como redundante del primero? Suponga que cada equipo cuesta $X. Note el incremento de disponibilidad al comprar el primer equipo con respecto al incremento al comprar el segundo. En su opinión, ¿Quién debe tomar en la organización la decisión de pasar de una disponibilidad a otra?

4.5.3 Hacer prospectiva

El último paso del proceso de planeación de capacidad tiene que ver con mirar hacia adelante. Recuerde que, desde la definición, dijimos que un proceso de planeación de capacidad de pensar en el servicio “a lo largo del tiempo”. Esto es, la solución que queremos debe funcionar bien hoy y en un horizonte de tiempo donde las cosas van a ser dinámicas.

Por definición, las organizaciones quieren crecer. Nadie monta un negocio sin la ambición de crecer. Eso implica que la capacidad computacional debería poder responder para atender la nueva demanda. Para ello, es importante elaborar un pronóstico del crecimiento del volumen de negocio basado en los planes y la tendencia observada, las nuevas aplicaciones que se esperan lanzar, limitaciones presupuestales, etc. Una vez más, se trata de un trabajo entre la alta gerencia y la dirección de tecnología.

Recuerde que en la sección anterior establecimos la relación entre la operación de un sistema por parte de los usuarios y el nivel de utilización de los recursos computacionales. Esta relación la podemos extender para establecer un modelo muy sencillo que nos permita relacionar los elementos necesarios para realizar el dimensionamiento de una infraestructura dados unos requerimientos. El nivel de servicio que se puede ofrecer

con una solución de tecnología depende de la carga a la que es sometido el sistema y los recursos con los que se cuentan:

$$Nivel\ de\ servicio = f\ (recursos, carga)$$

La anterior función tiene dos lecturas adicionales posibles. Para asegurar un nivel de servicio, dada una carga, es necesario disponer de X recursos. O, un conjunto definido de recursos puede soportar una carga X con un nivel de servicio predeterminado. Como mencionamos, los pronósticos se hacen sobre la carga, basados en ellos podemos determinar entonces la necesidad de nuevos recursos o la inminente degradación del nivel de servicio.

Al revisar la Figura 4-4 podemos ver que en esa gráfica encontramos la carga máxima que permitía mantener el tiempo de respuesta dentro de los límites aceptables por los usuarios. Con una proyección de crecimiento de negocio podemos determinar el momento en el que vamos a alcanzar ese nivel de carga. Ese día tendremos que aumentar los recursos o disminuir nuestras expectativas sobre el nivel de servicio ofrecido. Dependiendo de los procesos de adquisición de la organización, un ingeniero de infraestructura debería levantar una alarma sobre la necesidad de nuevos recursos con el tiempo suficiente para definir, cotizar, adquirir, recibir, instalar y configurar el nuevo sistema.

5 Referencias

5.1 Concurrencia

- *Fundamentos de Sistemas Operativos*. Silberschatz, Galvin, Gagne. Wiley, 2019
- *Sistemas Operativos modernos*. Andrew Tanenbaum. Pearson. 4th Edition.
- *Operating Systems: An Introduction*. Gard and Verma. MLI, 2017
- *Operating Systems: Three Easy Pieces.* Remzi y Andrea Arpaci-Dusseau.

5.2 Virtualización

- *Fundamentos de Sistemas Operativos*. Silberschatz, Galvin, Gagne. Wiley, 2019
- *Sistemas Operativos modernos*. Andrew Tanenbaum. Ed. Pearson, 2008.

5.3 Seguridad

- *The Code Book. Simon Singh*
- *Network Security – Private Communication in a Public World. Charlie Kaufman, Radia Perlman y Mike Speciner*
- *Operating System Security. Trent Jager. Editorial Morgan & Claypool Publishers. 2008*
- *Conceptos de Sistemas Operativos. Abraham Silberschatz, Peter Galvin y Greg Gagne. Editorial Wiley. 2013*
- *Sistemas Operativos Modernos. Andrew S. Tanenbaum. Editorial Prentice Hall. 2009*

5.4 Planeación de Capacidad

- *The Art of Computer Systems Performance Analysis. Raj Jain. Ed. John Wiley and Sons Inc. 1991.*