ELIYAHU M. GOLDRATT

EL SINDROME DEL PAJAR

DIAZ DE SANTOS

El síndrome del pajar

Eliyahu M. Goldratt

El síndrome del pajar

DIAZ DE SANTOS

Versión española de:

ENRIQUE FERNANDEZ DE BOBADILLA
Asociado del A. Goldratt Institute Iberica

Título original en inglés: The Haystack Syndrome

I.S.B.N.: 84-7978-129-7
Depósito Legal: M. 410-1997

Diseño de cubierta: Estuart, S.A. (Madrid)
Fotocomposición: MonoComp, S.A. (Madrid)
Impresión: Edigrafos, S.A. Getafe (Madrid)

Contenido

PRIMERA PARTE

FORMALIZACION DEL PROCESO DE DECISION

SEGUNDA PARTE

LA ARQUITECTURA DE UN SISTEMA DE INFORMACION

TERCERA PARTE

PROGRAMACION

PRIMERA PARTE

Formalización del proceso de decisión

1

Datos, información y proceso de decisión. Cómo se relacionan

Parece que nunca tenemos suficiente información, a pesar de que nos encontramos ahogados en un mar de datos.

Es probable que ustedes estén de acuerdo con esta afirmación, que sea un tema que les preocupe.

Si es así, ¿qué tal si lo discutimos? No pretendo una discusión ociosa, ni que lloremos el uno sobre el hombro del otro, ni que nos consolemos contándonos batallas. Vamos a discutirlo seriamente, así como si nos creyéramos que «entre los dos podemos cambiar el mundo», intentando encontrar una solución práctica a este terrible problema, una solución que, de verdad, funcione.

¿Por dónde empezamos?

Parece obvio que lo lógico sería empezar por definir con precisión lo que queremos decir con las palabras *datos* e *información*. ¿Cuál es la verdadera diferencia entre ellas? Este es el núcleo de nuestras quejas, ¿no es así? ¿Se han definido ya estas palabras? Puede que estén en los diccionarios y en algún libro de texto, pero no existe una definición práctica.

¿Cuántas veces hemos visto programas de ordenador que se nos ofrecen como «sistemas de información» y que, después de un somero examen, resultan ser sólo «sistemas de datos»?

¿Qué es un dato?

La dirección de un proveedor es un dato. El precio de compra de un elemento es un dato. Los detalles del diseño de un producto, o el contenido de un almacén, también son datos. Podemos decir que *dato* es todo conjunto de caracteres que describa algo, cualquier cosa, sobre nuestra realidad. Si este es el caso, ¿a qué podemos llamar información? Parece que la única forma que tenemos de contestar a esta pregunta es refutando lo que acabamos de afirmar. La dirección de un proveedor es un dato, pero para la persona que tiene que enviar una carta de reclamación, la dirección

del proveedor es, sin duda, *información*. Podemos decir que el contenido de un almacén es un dato, pero se convierte en información cuando pretendemos saber si podemos servir el pedido urgente de un cliente. El mismo conjunto de caracteres que definimos como datos puede convertirse en información si se dan ciertas circunstancias. Parece que la información existe o no dependiendo de quién la esté mirando.

¿Estamos dando vueltas sin avanzar? No necesariamente. Entendemos, intuitivamente, que la *información* es la parte de los datos que influye en nuestras acciones, o que puede que influya en nuestras acciones si falta o no está disponible. Para distintas personas, o, incluso, para la misma persona en distintos momentos, el mismo conjunto de caracteres puede ser un dato, o puede ser información. No podemos evitar darnos cuenta de que la diferencia entre dato e información no reside en el contenido de un conjunto de caracteres dado. Más bien reside en su relación con la decisión requerida. Si no sabemos con antelación qué tipo de decisión vamos a tomar, si no sabemos qué vamos a necesitar exactamente, entonces, todo dato podría llegar a considerarse como información en algún momento. ¿Resulta extraño que sea tan difícil distinguir entre una base de datos y un sistema de información?

Dado lo cambiante de nuestro mundo actual, ¿podemos llegar a situarnos en una posición desde la que podamos distinguir, a priori, qué es información? ¿Es de verdad posible diseñar algo que se pueda llamar, sin reservas, sistema de información, especialmente cuando pretendemos que el sistema no se use sólo para tomar un tipo de decisión o para servir sólo a un área de gestión?

Nos gustaría tener un sistema que facilitara información a todos los directivos de una organización, para todo tipo de decisiones. Por lo que hemos dicho hasta ahora, parece ser que, en un momento dado, la mayoría del contenido de ese sistema estaría formado, simplemente, por datos. ¿Y qué? Si es capaz de proporcionar la información, ¿qué nos importa eso?

Este es, exactamente, el tipo de razonamiento que nos ha conducido hasta los sistemas de información actuales. Se puede suponer que el siguiente paso lógico es empezar a preguntarnos sobre el tipo de cuestiones al que nos vamos a tener que enfrentar. Y no sólo nosotros, sino todas las áreas de la organización. Así, saltamos alegremente al paso siguiente: intentamos determinar qué parte de los datos/información necesitaremos.

Desde ahí sólo hay que dar un paso muy pequeño para encontrarnos totalmente inmersos en el esfuerzo de decidir los formatos precisos para introducir los datos, la estructura de los ficheros, las rutinas de búsqueda, etc.

Hemos abierto el camino hacia la monstruosa tarea de recopilar y mantener un mar de datos. Esta misma tendencia influirá en los formatos de los informes que saquemos. Serán muy voluminosos, pretendiendo

abarcar todas las preguntas que puedan surgir. Es cierto que, en los últimos años, los ordenadores personales y la posibilidad de trabajar en línea han eliminado este fenómeno hasta cierto punto, pero no han desaparecido las ideas subyacentes que nos llevaron hasta él.

En Israel se cuenta una historia que no puedo confirmar como cierta, aunque no me sorprendería que lo fuera. Hace diez años, los listados eran la única forma práctica de sacar la información de un ordenador. En aquella época, el departamento central de ordenadores del ejército israelí estaba considerando, como posible respuesta a sus oraciones, la entonces nueva tecnología de la gigantesca impresora láser. Un capitán de ese departamento, probablemente muy arrogante y algo irresponsable, decidió atacar el problema de una forma un tanto original. Sin pedir aprobación alguna, ordenó que se dejara de imprimir y enviar todo listado que tuviera más de cien páginas. En aquella época, la descentralización de los sistemas de ordenador era sólo un sueño, y se enviaban muchas copias de los listados desde el sitio central hasta muchos puntos del ejército. La leyenda cuenta que sólo se recibió una queja desde los puntos de recepción. El que protestaba era un individuo cuyo trabajo consistía en archivar ordenadamente los listados.

Cualquier directivo de una organización grande puede identificarse fácilmente con esta historia. Y, aunque esta historia sea una leyenda, hay muchas similares que, sin duda, son ciertas. Además, ¿de qué nos estamos quejando? De que nos estamos ahogando en un mar de datos. La situación actual ha llegado a tal extremo que siempre que sugiero, durante algún acto público, que deberíamos conectar directamente las impresoras con las máquinas destructoras de papel, la audiencia responde con risas y vítores. En algún punto del camino hemos perdido el rumbo. En algún punto del camino debe de haber algún error lógico. Puede que los sistemas de información no eliminen la necesidad de tener bancos de datos, pero, ciertamente, tendrán que ser algo muy distinto de éstos. Si queremos que sean efectivos, no pueden ser idénticos a nuestras bases de datos actuales.

Volvamos al punto donde establecimos la diferencia entre datos o información. Hemos intentado definir la información como «los datos que hacen falta para tomar la decisión». Este intento no nos llevó muy lejos, pero, aun así, sentimos, intuitivamente, que la información sólo se puede definir dentro del marco de la toma de decisiones. Quizá no debamos definir la información como «los datos necesarios para responder a una pregunta», sino, como «la respuesta a lo que se ha preguntado».

Esto no es una simple distinción semántica. Si lo pensamos durante un par de minutos, esto nos hará sentirnos un poco incómodos. En el momento en el que definimos la información como la respuesta a lo que se ha preguntado, estamos diciendo que la información no es lo que entra (da-

tos) en el proceso de decisión, sino lo que sale de él, su resultado. Aceptar esta definición implica que el propio proceso de decisión debe estar imbricado en el sistema de información. Esto exige conseguir un proceso de toma de decisiones que esté formalizado con mucha precisión. En nuestro caso significa, sin duda, que vamos a abrir una nueva caja de Pandora. Hoy en día, el propio proceso de decisión está cambiando en la industria.

Cada vez hay más profesionales que ven los años ochenta como la época en la que tuvo lugar la segunda revolución industrial. Una revolución en la forma de considerar la esencia de la dirección de nuestros negocios, una revolución que influye en los procedimientos básicos que usan los directivos para tomar decisiones. Cualquier discusión lógica de la composición y estructura de los sistemas de información deberá hacerse dentro del marco de la toma de decisiones. Así, no podemos escapar a la necesidad de analizar la nueva filosofía de dirección que ha empezado a surgir. A primera vista, esto puede parecer un gran rodeo. ¿Queremos discutir sobre sistemas de información, y, de repente, puede que tengamos que dedicar bastante tiempo a analizar filosofías de dirección? Esto es inevitable si queremos explorar la posibilidad de encontrar un método sólido que nos lleve a la creación de sistemas de información satisfactorios. Además, puede que la sencillez que caracteriza a estos nuevos movimientos nos ayude a encontrar soluciones nuevas, más sencillas y más potentes, para el tema que nos ocupa.

2

Lo que intenta conseguir una empresa

«La calidad es nuestra tarea número uno». «El inventario es un pasivo». «Equilibra el flujo, no la capacidad». Estos son sólo algunos de los conceptos que han sacudido los cimientos de la gestión industrial. Durante los años ochenta, fuimos testigos de tres movimientos de gran fuerza —*calidad total* (TQM), *justo a tiempo* (JIT) y *teoría de las limitaciones* (TOC)— que cuestionaban casi todo lo que hasta entonces se daba como bueno. Cada uno de estos movimientos tuvo su humilde principio en la aplicación de alguna técnica local, pero todos han evolucionado con una rapidez vertiginosa.

Ahora nos empezamos a dar cuenta de que nuestra percepción inicial de lo que abarcaban estos movimientos era demasiado estrecha. Creo que estarán de acuerdo conmigo si describo ese cambio de percepción de la forma siguiente:

> Ya es hora de que nos demos cuenta de que el objetivo principal de JIT no es la reducción del inventario en la planta. No es sólo una técnica mecánica de aplicación del kanban. Es, sin duda, una nueva filosofía global de dirección.
>
> Ya es hora de que nos demos cuenta de que el objetivo principal de TOC no son los cuellos de botella de la planta. No es sólo una técnica mecánica para optimizar la producción, es, sin duda, una nueva filosofía global de dirección.
>
> Ya es hora de que nos demos cuenta de que el objetivo principal de TQM no es la calidad de los productos. No es sólo una técnica mecánica de control estadístico del proceso. Es, sin duda, una nueva filosofía global de dirección.

No hace falta preguntar si se nota el parecido. Pero no vamos a conformarnos con nuestro nuevo nivel de entendimiento. Debemos plantearnos dos preguntas inevitables:

1. ¿Qué es lo verdaderamente nuevo en estas nuevas filosofías globales de dirección? Cuando considerábamos estos movimientos como técnicas locales, entendíamos con facilidad las novedades que aportaban. Pero, ahora, nuestra intuición ha aceptado una frase muy exigente: nueva filosofía global de dirección. Esto no es fácil de digerir. Ciertamente, las nuevas técnicas locales no justifican estas palabras tan altisonantes. De entrada, estos movimientos se circunscriben principalmente al área de producción. Entonces, ¿por qué usamos la palabra «global»? Además, aun siendo de gran fuerza, todavía no se merecen el nombre de filosofías de dirección. Si queremos justificar nuestra intuición, necesitamos expresar mejor lo que estos movimientos nos han aportado.
2. ¿Cuántas filosofías nuevas de dirección hay? ¿Una o tres? Hasta que no expresemos nuestro entendimiento actual con palabras precisas, no estaremos en disposición de ver si nos encontramos o no ante un dilema.

Mientras no contestemos a las dos preguntas que se han hecho arriba, seguiremos estando en la situación actual, donde al *síndrome del final de mes* hemos añadido, ahora, lo que solamente podemos describir como el *proyecto de mejora del principio de mes.*

Es obvio por dónde hay que buscar las respuestas a estas preguntas. Una frase como *nueva filosofía global de dirección* sólo se justifica cuando se da un gran cambio en los fundamentos. Ninguna mejora sobre un aspecto parcial, por grande que sea, puede justificar un título tan exigente como ése.

Es posible que la pregunta más fundamental que nos podamos hacer sea ¿por qué se crea una organización? No creo que haya ninguna organización que se haya creado solamente para su propia existencia. Toda organización se ha hecho para conseguir un propósito. Así, siempre que consideremos una acción en cualquier parte de cualquier organización, la única forma lógica de considerarla será juzgando el impacto de esa acción en el propósito global de la organización.

Bastante trivial. Pero, con este breve argumento se nos revela la base de toda organización. Lo primero que debe estar claramente definido es el propósito global de la organización, o su meta, como yo prefiero denominarlo. Lo segundo son las medidas. Y no cualquier medida, sino aquellas que nos permitan juzgar el impacto de una decisión local sobre la meta global.

Si queremos buscar algo significativo debemos empezar por la meta de la organización, y si ahí no encontramos ningún cambio, debemos buscar en sus medidas.

Vamos a empezar por la meta de la organización. Probablemente se habrán encontrado, como a mí me ha ocurrido, con que, en algunos casos, resulta bastante difícil conseguir una definición precisa. Vamos a emplear un poco de tiempo en intentar aclararnos con este tema. ¿Quién tiene derecho a determinar la meta de la organización? No hace falta ser un fenómeno para encontrar la respuesta más obvia. Los únicos que deben decidir la meta de una organización son sus propietarios. Cualquier otra respuesta nos obligaría a redefinir el significado de la palabra «propiedad».

Aquí nos encontramos ante un problema. Sabemos por experiencia que casi todas las organizaciones se enfrentan a grupos de presión. Un grupo de presión es aquel que tiene el poder de destruir, o de dañar seriamente, a la organización si no le gustan algunos aspectos del comportamiento de esta. Aparentemente, tenemos que dejar que esos grupos opinen. Pero, si les dejamos opinar, entonces, los propietarios no son los *únicos* que tienen derecho a establecer la meta. Un callejón sin salida.

Podemos salir de esta dicotomía distinguiendo claramente entre la meta de una organización y las condiciones necesarias que se imponen a su comportamiento. La organización debe esforzarse por alcanzar su meta dentro de los límites impuestos por los grupos de presión, esforzándose por cumplir su propósito sin violar ninguna de las condiciones necesarias que se le han impuesto desde fuera. Es indudable que, para una empresa industrial, los clientes son un grupo de presión. Imponen condiciones necesarias, como el nivel mínimo de servicio al cliente y el nivel mínimo de calidad del producto. Si no se cumplen estas condiciones mínimas, los clientes dejarán de comprar en la organización, y ésta correrá el riesgo de extinguirse. Pero, ciertamente, nadie nos podrá sugerir que nuestros clientes tienen derecho a dictaminar o interferir en la meta de nuestra organización.

Los empleados de la organización son otro grupo de presión. Imponen condiciones necesarias, como una mínima seguridad en el empleo y un mínimo en los salarios. Si la organización viola estas condiciones necesarias, corre el riesgo de enfrentarse a una huelga. Pero esto no significa que los empleados tengan derecho, como tales empleados, a determinar la meta de la organización.

El gobierno es un grupo de presión. Los gobiernos, incluso a nivel local, imponen condiciones necesarias, como los niveles máximos de contaminación del aire o del agua. Si una planta viola estas condiciones necesarias, se enfrenta a la amenaza de cierre, por muy rentable que sea. Pero esto no significa que el gobierno tenga derecho a decirnos cuál debe ser la meta de nuestra organización.

La meta de una empresa sólo está en manos de sus dueños. Si hablamos de una sociedad, los dueños de la empresa son los «accionistas». Así

pues, la pregunta «¿cuál es la meta de la empresa?» equivale, exactamente, a la pregunta «¿por qué invirtieron los accionistas su dinero en esta empresa?» ¿Qué querían conseguir? En vista de lo anterior, ¿qué pensarían de una empresa que afirmara «nuestra meta es suministrar los productos de mejor calidad junto con el mejor servicio al cliente»? Una empresa así debe tener unos accionistas muy especiales. Aparentemente, sus accionistas han invertido en la empresa para poder jactarse en las fiestas de que su empresa es la que da mejor servicio. ¿Es esta su empresa? Probablemente, no.

Pensemos, ahora, en una empresa que afirma que su meta es llegar a ser el número uno: van a captar la mayor cuota de mercado. Probablemente, los accionistas invirtieron en esa empresa porque son unos maníacos del poder. Pero la afirmación más ridícula, que desgraciadamente podemos encontrar en tantos libros de texto, es la afirmación de que la meta de la empresa es sobrevivir. Una afirmación así, sin duda, sitúa a la mayoría de los accionistas bajo la categoría de seres humanos altruistas.

Basta con que la empresa tenga una sola acción cotizando en bolsa para que su meta haya quedado claramente definida. Invertimos nuestro dinero en bolsa para ganar más dinero ahora y en el futuro. Esta es la meta de cualquier empresa que tenga acciones cotizando en bolsa.

Debe hacerse notar que la afirmación general no es «la meta de una empresa es ganar más dinero ahora y en el futuro». La afirmación general es «los dueños son los únicos que tienen derecho a definir la meta». Si se trata de una empresa que no cotiza en bolsa, no podremos predecir cuál es su meta. Se lo tendremos que preguntar a los dueños.

Es bastante alarmante ver cómo en tantas empresas que cotizan en bolsa sus directivos confunden las condiciones necesarias, los medios y la meta. Es frecuente que esta confusión conduzca a una mala dirección y a la destrucción, a largo plazo, de la empresa. El servicio al cliente, la calidad del producto, las buenas relaciones humanas son, sin duda, condiciones necesarias, e, incluso, medios, algunas veces. Pero no son la meta. Los empleados de una empresa tienen que servir a los accionistas, se les paga para eso. El servicio al cliente es sólo un medio para cumplir con su verdadero trabajo, servir a los accionistas de la empresa.

Aquí no hay nada nuevo. Es cierto que, a veces, hay mucha confusión, pero no hay nada nuevo. Así que, para encontrar qué hay de nuevo en estas filosofías globales de dirección, no nos queda más remedio que aplicar nuestro análisis y nuestro examen a la segunda entidad fundamental: las medidas.

3

Influencia de las medidas y de los sistemas de medida. Análisis e importancia de los sistemas de medida

Las medidas son una consecuencia directa de la meta elegida. No hay forma de elegir un sistema de medidas antes de que se establezca la meta. Por ejemplo, resultaría ridículo medir los resultados de un ejército o de una iglesia en términos monetarios.

En este texto nos vamos a ocupar del caso de la empresa cuya meta es ganar más dinero ahora y en el futuro. Es un caso bastante amplio, aunque no genérico. Por tanto, el análisis que se hace no es aplicable a otros tipos de empresa, aunque creo que el proceso lógico es probablemente el mismo.

Para juzgar el funcionamiento de una empresa nos basamos en sus informes financieros. Cuando hablamos de resultados no nos referimos sólo a un número, sino a dos. El primero es una medida absoluta, el *beneficio neto*. Este número aparece en la hoja de pérdidas y ganancias. El segundo es una medida relativa y, por tanto, un número puro: *el retorno de la inversión*, el retorno de los activos o el retorno por acción. Esta segunda medida aparece, hoy en día, en la hoja de balance. Existe un tercer número que no representa una medida, pero que es una condición necesaria muy importante: *el informe de caja*.

Es muy importante subrayar que estas medidas del resultado no son las medidas que buscamos. Estas medidas son capaces de medir la meta, pero las medidas que definimos como cantidades fundamentales son las que nos permiten juzgar el impacto de una decisión local sobre la meta de la empresa. Cualquier directivo sabe que las medidas de resultados son bastante incapaces de juzgar el impacto de una decisión local.

¿Cuáles son las medidas que usamos para juzgar las acciones locales? No, no vamos a sumergirnos en la interminable tarea de hacer una lista con todas las medidas locales que usan los distintos departamentos de las distintas empresas. Esto es un trabajo infructuoso, especialmente cuando

sabemos que, en muchos sitios, parece que las medidas predilectas dependen fuertemente del humor del jefe, del día de la semana o, incluso, del tiempo atmosférico. En vez de eso, es más simple, y también más fructífero, dedicarnos a hacer ejercicios mentales.

Este tipo de ejercicio mental es una de las herramientas más poderosas que se usan en física. Se llaman experimentos «Gedunken». Son experimentos que nunca se llevan a cabo en la realidad, sólo se piensan (*Gedunken* significa «pensar», en alemán). La propia experiencia es lo suficientemente amplia como para indicar los resultados con exactitud, así que no es necesario llevarlos a cabo. Hagamos un experimento «Gedunken».

Primero, debemos describir la empresa. Nos interesa encontrar las medidas para una empresa cuya meta es ganar más dinero ahora y en el futuro. ¿Qué es lo que genera esta empresa? ¿Metales en bruto? ¿Equipos electrónicos sofisticados? ¿Mercancías diversas? ¿Importa, en realidad? Sí. Lo que debemos entender es que todos los ejemplos que se han puesto arriba describen el producto material de la empresa, no lo que ésta genera. Mientras definamos la meta como lo hemos hecho, lo que la empresa genera (o debe generar), es sólo una cosa: dinero. Por tanto, podemos definir tranquilamente a la empresa como *una máquina de hacer dinero.*

Ya estamos de acuerdo en la meta: hemos decidido que nos gustaría tener una máquina de hacer dinero. Imaginemos que hemos entrado en la única tienda en la que venden estas máquinas. Hay muchas máquinas de hacer dinero en la tienda, y, desde luego, queremos elegir una de ellas. ¿Qué información necesitamos que nos dé el vendedor para poder elegir? Cuando hayamos expresado con palabras la información que necesitamos, habremos expresado las medidas.

Pero, además de las medidas, también podríamos expresar algunas condiciones necesarias. Pero, como en este experimento se trata de encontrar las medidas, y como las condiciones necesarias pueden variar mucho de una empresa a otra, vamos a suponer que todas las máquinas de la tienda cumplen nuestras condiciones necesarias. Por tanto, si podemos expresar claramente lo que necesitamos saber para hacer nuestra elección, habremos expresado, de hecho, las medidas necesarias. Vamos a aclarar que lo que esperamos del vendedor es que nos dé información sobre cada máquina. No debemos esperar, ni desear, que esta persona elija por nosotros.

La primera información necesaria que nos viene a la mente es: «¿Cuánto dinero hace la máquina?» Pero, debemos tener cuidado. Supongamos que el vendedor nos dice: *esta máquina hace un millón de dólares, esta otra sólo medio millón.* Supongamos que hemos elegido la primera máquina y nos encontramos que hace el millón de dólares, pero, tarda diez años. La otra

genera medio millón en tan sólo un año. ¿Deberíamos enfadarnos con el vendedor? ¿Por qué? El nos respondió exactamente a lo que le preguntamos. El problema no es del vendedor, es nuestro. No hemos preguntado lo que pretendíamos saber.

¿Qué es lo que de verdad queremos saber? El ritmo. ¿A qué ritmo genera la máquina el dinero? Supongamos que el vendedor nos dice que hay una máquina que genera dinero al ritmo de un millón de dólares al mes, y que hay otra que genera sólo medio millón al mes. Elegimos la primera, y nos encontramos con que se desintegra a los tres meses, mientras que la otra sigue funcionando para siempre. ¿Nos caerá bien el vendedor?

Una vez más, tenemos que aclarar lo que, de verdad, queríamos decir con nuestra pregunta. Por supuesto, nuestra intención no era preguntar solamente por el ritmo actual de generación de dinero, preguntábamos el ritmo como función del tiempo. Si nos hubiéramos explicado con claridad, el vendedor se habría visto obligado a decirnos que el ritmo de la primera máquina desciende a cero después de tres meses. No hay por qué perder tiempo culpando a los demás, cuando podemos evitarnos estos problemas simplemente expresándonos con más claridad.

Hay otro punto que debemos aclarar. Nos gustaría conocer la probabilidad que existe de que se cumplan las predicciones del vendedor. Toda respuesta numérica es sólo una estimación. Deberíamos exigirle conocer la fiabilidad de sus predicciones. Bajo este aspecto, debemos entender nuestra pregunta ¿a qué ritmo genera la máquina el dinero?

¿Es esto suficiente? Desde luego que no. También tenemos en mente el coste de la máquina. Pero, para variar, seamos cuidadosos. ¿Qué queremos decir cuando decimos «coste»? Coste es una de esas peligrosísimas palabras que tienen más de un significado. Podemos preguntar por el coste de la máquina refiriéndonos a su precio de compra. O podemos hacer la misma pregunta y referirnos al gasto operativo, cuánto cuesta hacerla funcionar. Una interpretación es del reino de la inversión, la otra, del reino del gasto. Son significados bien distintos. Recordemos que podemos llegar a ser muy ricos a base de invertir con prudencia, pero no a base de gastar dinero. Pero, aun así, ambos significados son de vital importancia.

¿Cómo debemos redactar nuestra pregunta? Preguntar el precio de compra de la máquina no es suficiente. Podríamos tener que enfrentarnos a algunas sorpresas desagradables si nos encontramos con que sus dimensiones físicas nos obligan a modificar un edificio, o con que la cantidad de inventario que la máquina se traga es aún más cara que su precio de compra. Yo sugeriría que preguntáramos: ¿cuánto dinero consigue la máquina? Y, una vez más, insistamos en obtener una respuesta que esté en función del tiempo y que tenga una probabilidad adecuada.

Que la máquina consiga dinero no significa que éste no sea nuestro. Sólo significa que, en el momento que quitemos, aunque sólo sea una parte de ese dinero, la máquina será incapaz de continuar produciendo, o, al menos, se degradará su rendimiento. Esto es muy distinto de la siguiente pregunta que todavía tenemos que hacer: ¿cuánto dinero tendremos que meter continuamente en la máquina para que ésta funcione?

4

La definición de los ingresos netos

Tres preguntas fáciles: ¿Cuánto dinero genera nuestra empresa? ¿Cuánto dinero consigue nuestra empresa? ¿Cuánto dinero nos tenemos que gastar para hacerla funcionar? Las medidas resultan intuitivamente obvias. Lo que se necesita es convertir estas preguntas en definiciones formales. Estas definiciones formales ya las he propuesto en el libro *La meta*.

La primera es el *ingreso neto*. El ingreso neto se define así: el ritmo al que el sistema genera dinero a través de las ventas.

En realidad, conseguiremos una definición más precisa si borramos las últimas palabras: *a través de las ventas*. Porque si el sistema genera dinero obteniendo intereses de un banco, esto es ciertamente ingreso neto. ¿Por qué añadí las últimas palabras? Por un comportamiento que es muy común en nuestras empresas. Muchos directores de producción piensan que si han producido algo, esto merece llamarse ingreso neto. ¿Qué opinan ustedes? Si hemos producido algo, pero todavía no lo hemos vendido, ¿podemos realmente llamarlo ingreso neto?

Esta distorsión no se limita a la producción. ¿Cómo reacciona el área financiera si se duplica el inventario de producto acabado? Si los productos no son obsoletos, ¿cómo se juzgará esta operación desde el punto de vista financiero? El director financiero nos dirá que, según la forma en la que se supone que debe tratarse a los números, hemos hecho algo muy bueno. Hemos absorbido más gastos y el informe financiero mostrará un aumento en el beneficio neto. Pero, nuestra intuición no estará de acuerdo con esto. No es posible que los ingresos netos tengan nada que ver con los movimientos internos de dinero. Ingresos netos significa traer dinero nuevo desde el exterior, y por eso se añaden las palabras: a través de las ventas.

Hemos subrayar que no se deben confundir los ingresos netos con las ventas. Ingreso neto es el ritmo al que el sistema genera dinero a través de

las ventas. ¿Qué diferencia hay? Supongamos que vendemos un producto por 100 dólares. Esto no quiere decir que nuestro ingreso neto aumente en 100 dólares. Puede que, en el producto vendido, haya piezas y materiales comprados a nuestros proveedores por valor de 30 dólares. Estos 30 dólares no son dinero generado por nuestro sistema, es dinero generado por los sistemas de nuestros proveedores. Este dinero solamente fluye a través de nuestro sistema. Así, en este caso, nuestro ingreso neto aumentará sólo en 70 dólares. El ingreso neto es el precio de venta menos las cantidades pagadas a nuestros proveedores por los elementos que han entrado en el producto vendido, sin que importe el momento de compra de estos elementos.

Además de las piezas y materiales, hay otras cantidades que hemos de sustraer del precio de venta para calcular el ingreso neto. Tenemos que deducir las subcontrataciones, las comisiones pagadas a vendedores externos, los aranceles, e, incluso, el transporte, si no somos los dueños del medio de transporte. Ninguna de estas cantidades son dinero generado por nuestro sistema.

Puede que hayan notado que la definición de los ingresos netos exige que determinemos el momento en el que se realiza la venta. Hoy en día, dos son las formas más comunes de tratar este tema. En una se considera que se realiza cuando el dinero cambia realmente de manos. La otra forma, más popular, es la técnica de las previsiones, que se supone que se aplica cuando la transacción es irreversible. Desgraciadamente, hay muchas empresas que no aplican estrictamente esta técnica.

En muchas industria de bienes de consumo, el fabricante no vende los productos directamente al consumidor, sino que lo hace a través de cadenas de distribución. En la mayoría de los casos, esos canales de distribución se reservan el derecho de devolver las mercancías sin dar ninguna explicación. Parece muy inadecuado que se registre una venta cuando los productos se envían a las empresas distribuidoras, ya que la transacción es ciertamente reversible. ¿Pueden ustedes creer que algunas empresas reembolsan al distribuidor según el precio actual, y no según el precio que se pagó? Probablemente saben que, en la industria de bienes de consumo, lo propio es realizar «promociones» o, en la nomenclatura del consumidor, «rebajas». Lo que significa que un distribuidor puede comprar mercancía durante el período de promoción, esperar dos meses, devolverla y ganar el 20 por 100 en tan sólo dos meses. Es todavía mejor que un negocio de la mafia. ¿Sucede esto en la realidad? En mucha mayor medida de lo que podría imaginarse alguien ajeno a esta industria. Pregunté al presidente de una empresa de productos de consumo por qué su empresa no había cerrado este coladero, después de llevar cincuenta años en el negocio. Su contestación fue: «Te equivocas, no llevamos cincuenta años en el negocio. Llevamos en el negocio doscientos trimestres». La venta se registra durante este trimestre, la devolución se registrará en el siguiente.

Este fenómeno no es precisamente divertido, pero, ciertamente, subraya el hecho de que no debe considerarse como punto de venta el momento en el que el dinero cambia de manos. Desgraciadamente, la negligencia en la determinación del punto de venta tiene ramificaciones que alcanzan aún más lejos.

Todos sabemos que los concesionarios de la mayoría de las empresas americanas y europeas del automóvil mantienen un inventario de vehículos de unos noventa días. Las empresas de coches consideran estos vehículos como ventas ya realizadas. Los concesionarios los han comprado. Pero si miramos un poco más detenidamente, nos quedaremos asombrados. Resulta que, en la mayoría de los casos, el concesionario ha comprado los vehículos con un crédito de la misma empresa de automóviles. ¿Cuál es el aval de este crédito? El propio vehículo. Si el concesionario se encuentra con un gran inventario de vehículos cuando cambia el modelo, ¿quién creen ustedes que dará los descuentos? No será el concesionario.

A todos los efectos prácticos, y desde el punto de vista de una práctica sana del negocio, aunque los coches estén en manos de los concesionarios, no deberían haberse declarado como ventas. Este método conduce a un conflicto devastador entre el corto plazo —las ventas de este trimestre—, y el largo plazo —la respuesta rápida al mercado— y, así, el aumento de las ventas futuras. Este problema no se circunscribe a las empresas de automóviles, es el problema de toda empresa que venda a través de un sistema de distribución y no directamente al consumidor final. Aquí es importante distinguir entre un cliente y un consumidor. La venta debería registrarse cuando se ha dado una transacción irrevocable con el consumidor, y no sólo con el cliente. El exceso de productos en los canales de distribución lo único que hace es aumentar la distancia entre el fabricante y su cliente final. Es casi una receta para perder ingresos netos en el futuro. Para eliminar este conflicto entre el corto y el largo plazo, sólo tenemos que definir el punto de venta.

5

Eliminación del solape entre el inventario y el gasto operativo

La segunda medida es el inventario. El inventario se define como «todo el dinero que el sistema invierte en comprar cosas que el sistema pretende vender». ¿Por qué decimos «todo el dinero»? Por algún motivo, todo el que lee la definición de arriba llega a la conclusión equivocada de que no incluye edificios ni maquinaria. Como más tarde demostraré, esta definición es idéntica a la convencional, en lo que se refiere a edificios y maquinaria. ¿Por qué usamos la palabra *inventario*, en vez de usar *activos*, que se entiende mejor? Se ha hecho a propósito para subrayar el hecho de que esta definición difiere drásticamente de la definición convencional en lo que se refiere al inventario del material.

¿Qué valor debemos asignar a un producto acabado que está en un almacén? De acuerdo con la definición que se ha dado arriba, sólo se nos permite asignar el precio que hemos pagado a nuestros proveedores por los materiales y piezas que hemos comprado para incorporarlos al producto. El sistema en sí no le ha añadido valor, ni siquiera en mano de obra directa. Esta desviación, ciertamente, contradice todo método convencional de valorar el inventario. No es *Fifo* ni *Lifo* ni nada por el estilo. ¿Por qué necesitamos esta desviación?

Valor añadido. ¿A qué? Al producto. Pero, nuestra preocupación no es el producto, sino, más bien, la empresa. Así que, lo que de verdad nos tenemos que preguntar es: ¿cuál es el único momento en el que añadimos valor a la empresa? ¡Sólo cuando vendemos, y ni un minuto antes! Todo el concepto de añadir valor a un producto se basa en un óptimo local distorsionado. Así que no debe sorprendernos que cause distorsiones en el comportamiento de la empresa. Vamos a examinar algunas de las distorsiones más comunes.

Supongamos que somos el director de fábrica de una planta de producción que pertenece a un gran grupo. No somos responsables de las ventas

ni del marketing, estas funciones las lleva la división que, en nuestro caso, está situada en otro estado.

El año pasado nuestra planta sólo consiguió un 1 por 100 de beneficio neto. La gratificación que nos dieron fue tan pequeña que nos hizo falta un microscopio para poder verla. Nuestra esposa nos está presionando para que nos mudemos a una casa más grande, y nuestro hijo acaba de entrar en un colegio carísimo. Definitivamente, necesitamos más dinero. Estamos decididos a conseguir una buena gratificación este año.

La previsión de ventas de nuestra planta es igual que la del año pasado. No podemos hacer nada al respecto. El departamento de ventas, como ya hemos dicho, no depende de nosotros. Pero la corporación nos está indicando (de una manera bastante segura), que va a considerar la reducción del inventario como una medida importante del resultado. Ultimamente, la dirección parece haberse dado cuenta de que el inventario es un pasivo. El inventario sí está bajo nuestro control, así que este año nos hemos concentrado en reducirlo, pero sin perjudicar a los otros resultados de la empresa.

Gracias a nuestro trabajo hemos conseguido reducir el nivel de inventario de material en proceso y de producto acabado a la mitad de lo que teníamos al empezar el año. Lo hemos conseguido sin perjudicar las ventas ni el servicio al cliente. De hecho, hemos mejorado el servicio al cliente. Además, hemos conseguido estos resultados sin hacer ningún tipo de inversión. No hemos comprado más equipos, ni hemos instalado un nuevo y sofisticado sistema de ordenadores. Ni siquiera hemos aumentado los gastos operativos, ni hemos contratado a un rebaño de consultores para que nos ayudaran a hacer el trabajo. Por otro lado, tampoco hemos reducido los gastos operativos.

¿Cómo hemos conseguido la reducción del inventario? Ya que las ventas eran constantes, nos hemos limitado a reducir, durante un tiempo, las compras y la producción. En efecto, nuestra mano de obra no estuvo a plena ocupación durante ese período transitorio, pero no podíamos despedir a nadie. Si lo hacíamos, no sólo corríamos el riesgo de tener problemas con el comité, además, nos arriesgábamos a no poder recontratarlos. Nuestra gente es muy buena y no les resulta difícil encontrar otro empleo. Como directores de fábrica, tenemos la suficiente experiencia como para despedirlos y encontrarnos, seis semanas más tarde, con que tenemos que entrenar a los nuevos que hayamos contratado.

Así que vamos a resumir nuestros resultados: las ventas y el servicio al cliente no se perjudicaron, las inversiones no aumentaron, los gastos operativos se mantuvieron constantes y el inventario de material en proceso y producto acabado sufrió un agradable descenso. Cualquier director se sentiría orgulloso de estos resultados.

Por cierto, ¿qué gratificación hemos conseguido? ¿Por qué estamos buscando otro trabajo? Muchos directores de fábrica se han encontrado en esta enigmática situación. Han hecho cosas que tienen sentido, y, de repente, se ven sorprendidos cuando los informes financieros condenan las acciones que han llevado a cabo. ¿Cuál será el juicio financiero del caso que hemos descrito arriba? ¿Por qué la corporación dio instrucciones a las plantas para que redujeran el inventario? Porque el inventario es un pasivo. Pero, cuando entran a juzgar los resultados al final del año, ¿en qué columna encontramos al inventario en los informes financieros? En la columna de activos. El activo es exactamente lo opuesto del pasivo.

Reducido el inventario, es un pasivo. Bien, ¡ya lo habéis reducido! Pues, ahora, vamos a cambiar las reglas. De repende se convierte en un activo, y tenemos el hacha apuntando hacia nuestro cuello. ¿Dónde está escondida la diferencia? Vamos a examinarlo un poco más de cerca.

El inventario se registra en nuestra hoja de balance como un activo, pero ¿con qué valor? En lo que se refiere a material en proceso y a producto acabado, el valor que se usa no es sólo el precio de la materia prima, sino, también, el valor añadido al producto. Cuando se reducen las compras, sólo se convierte en dinero el precio de los materiales que no se han comprado. El valor añadido no se compensa, con lo que aparecerá como una pérdida en los resultados de este año.

El punto de *vista local* de añadir valor al producto hace que muchas empresas retrasen considerablemente sus trabajos de reducción del inventario de materiales. El único momento en el que la empresa se puede permitir una acción de este tipo es cuando las ventas han subido lo suficiente como para compensar el impacto negativo de la reducción del inventario. Este fenómeno se puede observar a gran escala por toda Europa y Estados Unidos. No es de extrañar. Sólo debemos recordar el impacto de una distorsión en las medidas.

> Dime cómo me mides, y te diré como me comporto. Si me mides de forma ilógica, no te quejes si me comporto de forma ilógica.

La serpiente del valor añadido también levanta su fea cabeza, de otra manera, que causa un efecto aún más devastador. A principios de los años ochenta había una empresa americana que facturaba unos 9.000 millones de dólares, y que acabó el año con una pequeña pérdida. Esto sucedió tras muchos años de dar beneficios, y fue totalmente inesperado.

Pueden imaginarse que Wall Street no reaccionó de forma favorable. No se puede decir que los accionistas estuvieran contentos. En un lapso de tiempo sorprendentemente corto rodó la cabeza del director ejecutivo y se contrató a otro director más duro.

El nuevo ejecutivo declaró abiertamente que a él no le interesaba el parloteo humanitario. Sólo le interesaba una cosa: el resultado. Al consejo de administración le encantó, probablemente lo habían contratado por eso mismo. Lo primero que hizo, según cuenta la historia, fue pedir una lista de todas las piezas que fabricaba la compañía. ¡Pueden imaginarse el tamaño del listado que le llevaron!

Quería saber cuánto costaba a la compañía, exactamente, fabricar cada pieza, y cuál sería el precio externo de compra si se pudiera encontrar un proveedor que la suministrara. Entonces, puso en práctica una política obligatoria: toda pieza que resulte más barata comprándola fuera —no hay que tener sentimientos, estamos aquí para hacer negocio— se debe dejar de producir inmediatamente y se debe comprar en el exterior. Por supuesto, habría que hacer los ajustes de plantilla que fueran necesarios.

¿Toma su empresa el mismo tipo de decisión cuando se considera la cuestión de fabricar o comprar? En todo caso, en esta ocasión se hizo a gran escala, perfectamente centrado y ejecutado con rapidez. No se permitió que nadie aplazara la ejecución. Algunas personas que intentaron retrasar el proceso fueron sometidas a medidas ejemplares.

Cuatro meses después, el director pidió una lista actualizada. En los negocios, es muy importante tenerlo todo bajo control. Una vez más se recopiló la lista de las piezas que todavía se producían, cada una de ellas con su coste interno y con su precio de compra actualizados. Todos sabemos lo que pasó.

Hay veces que es posible despedir a los empleados. Es mucho más difícil despedir a una máquina. Y es más difícil, todavía, despedir a un edificio. Cada una de las piezas que había quedado tenía que soportar, ahora, los costes que previamente compartía con sus «amigas», que ahora se compraban en el exterior. Cada una de las piezas se volvió más cara. Así que pasaron muchas más piezas a la categoría de suministro externo, ya que resultaban más caras produciéndolas dentro que comprándolas fuera. Tuvo lugar otra enorme oleada de recortes.

¿Parece ridículo? No es tan raro si nos acordamos de que la mayoría de las empresas occidentales utilizan el mismo concepto, aunque no en tan gran escala. Entonces, según continúa la historia, llegó el cuarto trimestre. El director se quedó más que sorprendido. Sus informes financieros no eran precisamente brillantes. La luna de miel con su consejo de administración y con Wall Street había terminado.

Hizo una evaluación rápida, y se dio cuenta de que la gran mayoría de las inversiones de la compañía se encontraban en las plantas de ensamblaje final, así, que decidió centrarse en ellas. Al menos, haría las plantas de ensamblaje tan eficientes como fuera posible. ¿Cuál es su principal excusa, falta de piezas? Les proporcionaremos un suministro generoso. Como era un ejecutivo realista, consiguió crédito de doscientos cuatro bancos. Utilizó este dinero para garantizar que todas las plantas de ensamblaje trabajaran ininterrumpidamente tres turnos diarios, siete días a la semana, durante todo el cuarto trimestre. Las eficiencias alcanzaron unos niveles que nunca se habían visto antes. Por supuesto, los pedidos existentes no podían absorber un ritmo de producción tan alto, pero no hubo gran dificultad en adelantar una parte de las previsiones de ventas. Otra práctica común en muchas empresas occidentales.

Al final de ese año, los informes financieros apoyaban totalmente las acciones del director. Se habían absorbido muchos gastos, y los resultados brillaban. El director fue generosamente recompensado, pero eso no le consoló demasiado, porque no sabía cómo continuar. Así, que se limitó a presentar su dimisión. Esta historia podría haber sido sólo una anécdota interesante, si no hubiera sido por sus ramificaciones. Al año siguiente, decenas de miles de personas perdieron sus empleos, la empresa se redujo a un tercio de su tamaño anterior, y tuvo que cambiar su nombre. ¿Pueden decir qué compañía era?

Este tipo de actuación es común en la gestión de las empresas: la búsqueda, no del beneficio real, sino de beneficios artificiales que se consiguen jugando con los números. El concepto de valor añadido permite la existencia de nociones ridículas, como «beneficios del inventario» y «pérdidas del inventario». Normalmente (aunque no siempre) las empresas paran el juego de los beneficios del inventario antes de que su situación de caja se perjudique irreversiblemente. Pero esto no significa que no se produzcan daños. Los almacenes de distribución de producto acabado se llenan hasta los topes, haciendo que la empresa quede muy separada de sus clientes en el tiempo.

Hoy en día muchas empresas están sirviendo a sus clientes a través de una distancia de tres a seis meses de inventario de producto acabado, en un mundo donde el ciclo de vida de su producto es de menos de dos años. ¿Qué le puede pasar a esa empresa si tiene que competir con otra que, aunque fabrique en el otro lado del globo, está sólo a treinta días del mismo mercado? ¿Quién ganará a la larga? ¿Quién ha ganado ya en tantas industrias? ¿Pueden estas empresas librarse de la carga de inventario que llevan a sus espaldas? Sólo de forma muy lenta, si tenemos en cuenta la mentalidad actual de los inversores.

Probablemente la alta dirección no sería capaz de explicar a sus inversores los desastrosos resultados.

> Dime cómo me mides y te diré cómo me comporto. Si me mides de forma ilógica, no te quejes si me comporto de forma ilógica.

No subestimemos el impacto que tiene esta afirmación en la viabilidad de las empresas.

Quitar al inventario el valor añadido no significa que no tengamos esas salidas de dinero. Su contabilización es tarea de la tercera medida: el gasto operativo.

El gasto operativo se define como todo el dinero que el sistema se gasta para convertir el *inventario* en *ingresos netos*.

De nuevo, las palabras «todo el dinero». El gasto operativo no es sólo el dinero que pagamos a la mano de obra directa. ¿Cuál es el trabajo de un vendedor, si no es convertir el inventario en ingresos netos? ¿Cuál es el trabajo de un encargado? ¿Cuál es el trabajo de un director o el de su secretaria? ¿Por qué diferenciamos entre personas que están haciendo exactamente el mismo trabajo, sólo porque algunos de ellos toquen físicamente los productos?

Hay que hacer notar las distintas palabras que se han elegido para las dos últimas definiciones. Se *invierte* en inventario, se *gasta* en el gasto operativo. ¿En qué categoría pondríamos los salarios de los ingenieros que están haciendo investigación y desarrollo?

Para clarificar aún más el uso de estas tres definiciones, vamos a cumplir la promesa que hicimos. Todavía tenemos que demostrar por qué declaramos que nuestra definición de inventario está en línea con la difinición convencional en lo que se refiere a maquinaria y a edificios. Vamos a considerar, por ejemplo, .la compra de aceite para la lubricación de máquinas. En el momento de la compra, no deberíamos considerar como gasto operativo el dinero que hemos pagado al proveedor. Todavía tenemos el aceite. Sin duda, es inventario. Entonces, empezamos a usar el aceite; la parte que hemos usado tiene que quitarse del inventario y tiene que ser reclasificada como gasto operativo; simple sentido común.

Ahora, vamos a considerar la compra de material. El dinero que hemos pagado a los proveedores no es gasto operativo, es inventario. Entonces, procesamos los materiales para que se conviertan en ingresos netos. Durante el proceso se estropea una parte del material. La parte que se estropea se debe quitar del inventario y se debe reclasificar como gasto operativo.

Vamos a considerar la compra de una nueva máquina: el precio de compra no es gasto operativo porque poseemos la máquina; es inventario. A medida que la vamos usando, la vamos deteriorando gradualmente, así que, de vez en cuando, hay que quitar del inventario una parte de su valor para pasarlo a gasto operativo. ¿Cómo llamamos al mecanismo que se supone que hace esta labor? *Depreciación* o *amortización*.

6

Las medidas, el resultado y la contabilidad de costes

Hemos visto dos diferencias muy apreciables entre las medidas convencionales y las que hemos propuesto aquí. ¿Hemos llegado a la causa central por la que intuimos que estamos tratando con *nuevas filosofías globales de dirección*? Aunque nos encantaría creérnoslo, desgraciadamente esto no se sostiene bajo un escrutinio serio. Primero, las distinciones que hemos mencionado sólo las hace la *teoría de las limitaciones*. Los otros dos movimientos, JIT y TQM, no se han molestado en expresar con precisión las medidas fundamentales y, por tanto, son bastante insensibles a estas distinciones tan exactas. Otra razón es que las medidas fundamentales, ingresos netos, inventario y gasto operativo también se usan en la gestión convencional.

Para probar este punto, es fácil demostrar que todo director está familiarizado con ingresos netos, inventario y gasto operativo. Estamos tan familiarizados que podemos decir, sin pensarlo, en qué dirección queremos que se mueva cada una de estas medidas. Hágase la siguiente pregunta: «¿Quiero que los ingresos netos aumenten o disminuyan?». La respuesta es obvia: *nos gustaría aumentar el ritmo al que nuestra empresa genera dinero.*

Inventario. ¿Aumentar o disminuir? Todos contestarán: *nos gustaría disminuir la cantidad de dinero que nuestra empresa absorbe.* ¿Y qué pasa con el gasto operativo? Esto es tan obvio que no merece la pena contestarlo.

Vamos a profundizar un paso más. Si tenemos tres medidas, toda acción deberá evaluarse de acuerdo con su influencia en las tres. ¿Cuál es uno de los métodos más poderosos para reducir el gasto operativo? Despedir a todo el mundo. El gasto operativo baja maravillosamente. Por supuesto que el ingreso neto se va a paseo, pero, ¿a quién le importa eso?

Al evaluar una acción, debemos recordar que tenemos tres medidas, y no sólo una. Si no es así, se llevarán a cabo acciones devastadoras. Esto quiere decir que el juicio definitivo no lo dan las medidas en sí, sino las

relaciones entre ellas. Tener tres medidas implica, matemáticamente, dos relaciones. Se puede elegir cualquier par de relaciones, siempre que incluyan las tres medidas. ¿Podemos elegir algo que ya sea conocido? Consideremos, por ejemplo, la siguiente relación: ingresos netos menos gastos operativos: IN-GO. ¿Resulta familiar? Sí, correcto, es nuestro viejo amigo, el *beneficio neto.*

¿Y qué pasa con esta otra relación, algo más complicada: ingresos netos menos gastos operativos dividido por inventario: (IN-GO)/I? Esta resulta aún más familiar. Es, simplemente, el *rendimiento de la inversión.*

No hay nada nuevo en las medidas fundamentales (IN, I, GO), como puede verse claramente por el hecho de que nos llevan directamente a los antiguos y convencionales juicios del resultado. Así que, parece que estamos atascados. Si no hay nada nuevo en la meta, y no hay nada nuevo en las medidas, ¿cómo podemos justificar el nombre de *nueva filosofía global de dirección?*

Antes de sumergirnos en ello, podría ser interesante recordar que se puede usar como indicador cualquier par de relaciones de las tres medidas. ¿Hay algún otro par que esté en uso hoy día?

Consideremos, por ejemplo, las dos siguientes relaciones directas: ingresos netos divididos por gastos operativos (IN/GO) e ingresos netos divididos por inventario (IN/I). ¿Podemos ponerles un nombre? Sí, en efecto, la primera es la definición convencional de *productividad*, y a la segunda la llamamos, normalmente, *rotación del inventario.*

Podemos usar el par *beneficio neto y rendimiento de la inversión*, o el par *productividad y rotación del inventario.* Elija la que más le guste, pero no use las cuatro al mismo tiempo, a menos que quiera acabar confundido. Ahora, piense detenidamente en las personas que nos dicen autoritariamente que debemos tener un sistema de medida para la macro, como *beneficio neto y retorno de la inversión*, y otro sistema para la micro, como *productividad y vueltas de inventario.* Qué afirmación tan absurda, especialmente porque nos damos perfecta cuenta de que mientras nuestra empresa tenga una sola meta, será mejor que tengamos un solo sistema financiero.

¿Dónde entra la contabilidad de costes en este cuadro, si es que entra? A primera vista parece que no tiene sitio. Pero, eso no es necesariamente así. La contabilidad de costes, cuando se inventó, fue una idea genial que respondió a una necesidad muy importante, casi obligatoria. Puede que sea mejor mirarlo desde un punto de vista más filosófico, si queremos entender el papel que desempeña la contabilidad de costes hoy en día.

La contabilidad de costes ha sido víctima de un proceso muy generalizado. Muchas empresas son víctimas de él, sólo porque no se han dado cuenta. En el mundo actual, tan cambiante y competitivo, es esencial saber que este proceso existe, si no queremos vernos paralizados en más de un

aspecto. De alguna forma, la mayoría de los directivos buscan un asidero desesperadamente, una solución que sea definitiva. Pero, las soluciones definitivas no sólo implican la capacidad de reconocer la verdad, también presuponen un mundo estable y sin cambios. Las soluciones definitivas no existen en la realidad, lo que sí hay son soluciones potentes.

Las soluciones potentes son las que tratan problemas muy graves. Una solución no puede ser potente, por buena y elegante que sea, si lo que trata es un problema trivial. La solucion potente es una respuesta definitiva a un problema grave, un problema que afecte a los resultados globales de la empresa, que distorsione el comportamiento y las acciones de sus directivos y de muchos de sus empleados. Así, la aplicacion de una solución potente conduce directamente a un impacto drástico en la empresa, a un cambio en su conducta y también en sus resultados.

Siempre debemos tener presente el hecho de que nuestras organizaciones no existen en un vacío, sino que están en continua interacción con su entorno. Cuando una empresa sufre un cambio significativo, afecta a su entorno. Como resultado, se demandará un funcionamiento aún mejor. La aplicación de una solución potente causa un cambio drástico en la empresa. En consecuencia, se provoca un cambio en su entorno, creándose nuevos retos que pueden convertir en obsoleta a la solución que se aplicó.

Debemos reconocer la existencia de una realidad desagradable: cuanto más potente es una solución, más fácil es que se quede obsoleta. Si ignoramos esta realidad, sólo podemos llegar a una conclusión: *¡la solución potente de ayer se puede convertir en el desastre de hoy!*

Esto es exactamente lo que pasó con la contabilidad de costes. En el próximo capítulo analizaremos hasta qué punto la contabilidad de costes, cuando se inventó, fue una de las soluciones más potentes de la historia industrial. Fue una de las principales herramientas que permitieron a la industria florecer y crecer con una rapidez considerable. Debido a este crecimiento, creció exponencialmente la necesidad de una mejor tecnología. Este mismo crecimiento proporcionó los medios para financiar la invención y el desarrollo de la tecnología. Pero, a medida que la tecnología avanzaba, cambió la relación entre la necesidad de fuerza muscular y la necesidad de fuerza cerebral. El factor de gastos generales de las empresas ha crecido en el último siglo desde un modesto 0,1 hasta la situación actual, donde la mayoría de las empresas tienen un factor entre 5 y 8. El coste de la mano de obra directa, cuando se inventó la contabilidad de costes, era unas diez veces mayor que el coste de los gastos generales. Hoy día estamos acercándonos rápidamente al momento en el que será sólo la décima parte de los gastos generales.

La contabilidad de costes fue una solución potente. Cambió la conducta y el funcionamiento de las empresas industriales. La industria, a su vez,

influyó en la tecnología. Entonces, la misma tecnología dejó a la contabilidad de costes en el aire: los supuestos en los que se basaba la contabilidad de costes ya no son válidos. La solución potente se hizo obsoleta a sí misma. Hay muchas empresas que ya se están enfrentando al desastre que ocasiona la aplicación de una solución obsoleta.

7

Descubriendo los fundamentos de la contabilidad de costes

Hemos definido tres medidas. ¿Son las que nos hacen falta? ¿Nos permitirán estas medidas juzgar el impacto de una decisión local en la meta global? Sentimos intuitivamente que son adecuadas, pero debemos reconocer que ni siquiera hemos empezado a demostrar esta afirmación.

Cualquier intento de enjuiciar una decisión local saca a la luz, inmediatamente, la necesidad de descomponer cada medida en sus componentes. ¿Cuáles son los componentes de los ingresos netos de la empresa? Los ingresos netos de la empresa resultan de la venta de un tipo de producto, más la venta de otro tipo de producto, etc. Los productos pueden ser servicios prestados. Así que, el ingreso neto de la empresa es la suma del ingreso neto obtenido a través de las ventas de cada producto individual.

Lo mismo se puede hacer con los gastos operativos. Gastamos dinero para convertir el inventario en ingresos netos. ¿A quién damos este dinero? A los trabajadores y directivos por su trabajo, a los bancos por los intereses, a las compañías suministradoras de energía, a las empresas de seguros, etc. Hay muchas categorías de gastos. Debemos hacer notar que el producto no es una de estas categorías. ¿Han pagado dinero a un producto alguna vez?

Asimismo, es importante hacer notar que los proveedores de material productivo tampoco son una categoría de gasto operativo. El dinero que se les paga no es gasto operativo, es inventario. Así, el gasto operativo total de la empresa es simplemente la suma de cada una de las categorías individuales de gasto operativo.

La descomposición del inventario en sus factores resulta obvia. Estas descomposiciones introducen una posible dificultad en el uso de estas medidas para las decisiones a nivel local. Al final, el juicio se basa en el beneficio neto y en el rendimiento de la inversión. Ahora vamos a ver la

situación tan extraña en que nos encontramos. El beneficio neto es simplemente el ingreso neto menos el gasto operativo.

La primera suma es de productos, la segunda lo es de categorías. Si intentamos sumar manzanas y naranjas, el resultado será una ensalada de frutas. Así que, ¿cómo vamos a gestionar el caso importante que se describe a continuación? Supongamos que estamos considerando el lanzamiento de un nuevo producto. Tenemos una buena estimación de cuánto vamos a vender. Lo que nos interesa de verdad no es el lanzamiento del producto, sino el impacto que tendrá en el beneficio neto de toda la empresa.

¿Cómo podemos contestar a esta pregunta si no sabemos el impacto del lanzamiento del nuevo producto sobre las ventas de los demás productos? ¿Cómo podemos contestar, si no conocemos su impacto sobre las diversas categorías del gasto operativo? Podría suceder que bajara nuestro beneficio neto total, aunque el ingreso neto obtenido con este nuevo producto fuera muy alto. Esta es una decisión muy importante, pero parece que no podemos alterar las decisiones locales, ni aun utilizando las nuevas medidas.

La contabilidad de costes se inventó para contestar a este tipo de cuestión tan importante. Probablemente, el genio que la inventó siguió el siguiente razonamiento lógico: *no puedo contestar con exactitud a vuestra pregunta, pero no hace falta contestarla con exactitud. En cualquier caso, la respuesta se basará en vuestra estimación de cuánto vamos a vender de este nuevo producto. Lo que necesitáis es una buena aproximación, y eso sí os lo puedo dar.*

Su solución intentó simplificar la situación pasando de manzanas y naranjas a manzanas y manzanas, de dos descomposiciones distintas, producto y categorías de gasto, a una sola. Probablemente dijo: *puedo encontrar una descomposición alternativa para los gastos operativos, no por categorías, sino por productos. Ya sé que no será exacto, pero será una aproximación bastante buena.*

Lo único que tenemos que hacer es cambiar la pregunta que nos llevó a la descomposición de los gastos operativos. En vez de hacer la pregunta habitual: ¿a quién estamos pagando dinero?, vamos a preguntar: ¿por qué pagamos dinero? ¿Por qué pagamos a un trabajador? Porque hemos decidido fabricar un producto específico. Por tanto, al menos, la categoría de mano de obra directa la podemos dividir producto a producto.

Debemos recordar que, cuando se inventó la contabilidad de costes a principio de siglo, en la mayoría de las empresas se pagaba la mano de obra directa según las piezas que producía. No era como hoy; que se paga por horas de presencia en la fábrica, además de que existen muchas razones (y no sólo los sindicatos) para evitar los continuos despidos y la inestabilidad en las plantillas de las empresas.

¿Qué pasa entonces con las otras categorías de gastos en las que resulta imposible la división por producto? Por ejemplo, es evidente que no

pagamos un sueldo al presidente porque hayamos decidido fabricar un producto determinado. Vamos a meterlas todas en el mismo saco, preferiblemente bajo un nombre que tenga connotaciones negativas[1]. Un nombre así expresa perfectamente nuestro disgusto por los gastos que no entran limpiamente en nuestra distribución por producto. Es verdad que corremos un riesgo al llamar «cargas» a los directores, pero con un poco de suerte no se darán cuenta.

Bromas aparte, ¿qué vamos a hacer con todos esos gastos? La persona que inventó la contabilidad de costes no lo dudó. Recordemos que, a principios de siglo, estos gastos eran extremadamente pequeños comparados con los de la mano de obra directa. Ya había solucionado la parte más significativa. Así que, su sugerencia fue muy directa: *repartid todos esos gastos de acuerdo con la contribución de la mano de obra directa.* Se había inventado la *asignación.*

¿Qué ganó con este truco? Pudo dividir los gastos operativos por producto, de la misma forma que había dividido los ingresos netos. Ahora podemos dar el siguiente paso, ya que tenemos manzanas con manzanas. La presentación matemática adopta una forma más simple.

Este es un gran logro. El tratamiento que sugería la contabilidad de costes nos permitía diseccionar una empresa por cada clase de producto. Ya podíamos tomar decisiones con respecto a un producto sin tener que mirar a los demás.

Esta innovación era extremadamente potente. Permitió que las empresas crecieran en tamaño mientras ampliaban sus gama de productos. Es interesante hacer notar que Dupont y General Motors fueron de las primeras empresas que adoptaron la contabilidad de costes. Ford no lo hizo, al estar limitado a un solo producto principal.

Pero, hoy, la situación es un poco distinta. Los avances de la tecnología han cambiado la industria hasta un extremo en el que ya no son válidas las dos suposiciones fundamentales de la contabilidad de costes. Ya no se paga la mano de obra directa por las piezas producidas, sino por el simple hecho de que los trabajadores asumieron la obligación de acudir a trabajar. Los gastos no identificables directamente con un producto ya no son una pequeña fracción de los gastos generales, sino que llegan a ser mayores que los gastos de mano de obra directa.

Hoy, toda la comunidad financiera se ha dado cuenta de que la contabilidad de costes ya no es aplicable y de que hay que hacer algo al respecto. Desgraciadamente, no vuelven a las bases, a la lógica de los informes financieros, para buscar allí las respuestas a estas importantes preguntas

[1] En inglés se da a esta parte de los gastos el nombre de *overhead*, literalmente «encima de la cabeza», o *burden*, literalmente «carga». (N. del T.)

sobre el negocio. En vez de eso, la comunidad financiera está completamente inmersa en el intento de salvar la solución obsoleta.

Los nombres de estos esfuerzos infructuosos son conductores de costes y costes basados en actividad. Es evidente que no podemos seguir asignando de acuerdo con la mano de obra directa. Así que, lo hacen diciendo: podemos asignar algunos costes a nivel de unidad, otros, sólo a nivel de lote, otros, a nivel de producto, otros, a nivel de grupo de productos y, sólo algunos, a nivel de empresa. Sí, podemos hacer la asignación de esta forma, pero, ¿para qué? En cualquier caso no podemos agregar a nivel de unidad, ni siquiera a nivel de producto, así que, ¿para qué jugar tanto con los números? Recordemos que la asignación se inventó para pasar de dos divisiones distintas, por productos y por categorías, a una sola división. El único propósito era conseguir una sola clasificación, para poder diseccionar la empresa y poder tomar mejores decisiones. Ahora, bajo el nombre de asignación, en vez de reducir el número de clasificaciones, lo estamos inflando. Nos hemos enamorado de una técnica. Hemos olvidado su propósito: permitirnos juzgar el impacto de una decisión local en el resultado.

Hemos llegado a una encrucijada: podemos seguir explorando cómo las medidas fundamentales se pueden usar para las decisiones locales, y desarrollar una solución alternativa al problema que la contabilidad de costes ya no es capaz de resolver; o podemos continuar el examen de lo que es nuevo en las nuevas filosofías globales de dirección; o podemos exponer el daño que está causando a las empresas del mundo occidental el uso de la errónea aproximación de la contabilidad de costes. Las tres avenidas son importantes. Las tres deben tratarse. Pero, ya que el ataque a la contabilidad de costes provoca alegría en tantas funciones de la empresa, continuemos primero con este tema.

8

La contabilidad de costes era la medida tradicional

El problema de cómo librarse de la contabilidad de costes no reside en la comunidad financiera, sino en la dirección de otras funciones de la empresa. Los practicantes de la contabilidad de costes la dejan con alegría cuando se les presenta una alternativa lógica y práctica. Ellos son los que mejor saben hasta qué punto no funciona la contabilidad de costes.

Hable con cualquier director financiero, escuche sus quejas. Le dirán que compilan los números de la única forma que saben hacerlo. Después, otros directivos toman las decisiones, aparentemente basadas en esos números. Si hubieran preguntado, el director financiero les habría dicho que su decisión no tiene nada que ver con los números compilados. Lo que pone furiosos a los financieros es el hecho de que, después, los directivos tienen la cara dura de echarles la culpa por sus decisiones. El grito de los directores financieros es: ¡dadnos un sistema mejor! No, ciertamente el problema no reside en la gente de finanzas. Los que no quieren prescindir de la contabilidad de costes son los directivos que están en producción, diseño, compras, distribución, y, desde luego, ventas. ¿Por qué están tan apegados a este dinosaurio?

La única forma que tengo de explicarlo es reconociendo que la contabilidad de costes ha traído consigo su propia nomenclatura. Todos hemos nacido en un mundo en el que esta nomenclatura ha llegado a ser parte de nuestra realidad. Examinemos la siguiente criatura matemática: *gasto operativo de un producto*, el resultado de la asignación. Esto, ciertamente, es sólo un fantasma matemático. Nunca hemos pagado dinero a un producto. Aun así, hoy en día tenemos un nombre para ello. Lo llamamos *coste del producto*.

Si ya no aceptamos el enfoque de la contabilidad de costes, debemos ser coherentes y borrar su nomenclatura. El coste del producto sólo existe cuando aceptamos ese enfoque. En la fórmula original, ingresos netos

menos gastos operativos (la base de nuestra fórmula de pérdidas y ganancias), existen costes de diversas categorías, pero no el coste de los productos. Intente imaginar lo que pasaría si se elimina el término *coste del producto*. Probablemente, todos los ingenieros de diseño se harían el harakiri. Habrían perdido la vara de medir que les guía durante las últimas fases del diseño. Pero, no es *coste del producto* el único término que nos ha traído la contabilidad de costes. Veamos la fórmula que resulta.

La expresión «ingresos netos menos gastos operativos de un producto», que llamamos beneficio neto de un producto es, sin duda, un fantasma matemático. El beneficio neto existe sólo para la empresa, no para el producto. Esto significa que todos los términos siguientes, ganancia del producto, margen del producto y coste del producto, deben ser omitidos de nuestro vocabulario en el mismo momento en el que reconocemos que el enfoque ya no es válido. Intento imaginar la cara que pondría un vendedor cuando descubriera que tenía que eliminar estos términos de su vocabulario.

Desgraciadamente, estos términos han echado raíces profundas en nuestro proceso de decisión. Los usamos aun en los casos en los que podríamos tomar la decisión usando la fórmula original de pérdidas y ganancias más fácilmente, mucho mejor y sin aproximaciones inexactas. Vamos a explorar sólo dos ejemplos de los muchos que existen.

Todo grupo de empresas en Europa y en Estados Unidos informa de sus resultados según la fórmula original: ingresos netos totales menos gastos totales de operación. A pesar de ello, si profundizamos en estos grupos hasta el nivel de división, y, desde luego, al nivel de planta e inferiores, encontraremos otro mecanismo. Encontraremos una diabólica criatura llamada *presupuestos*. ¿Qué es un presupuesto? Es sólo la construcción de la fórmula original de pérdidas y ganancias a base de aproximaciones. La construcción del beneficio neto de la planta a través del beneficio neto de los productos individuales. Por supuesto, no coincide. Así que, llamamos *variación* al desajuste, ¡y ya coincide!

¿Qué es lo que resulta de este método tan torpe? Limítese a entrar en una planta hacia el final de mes, cuando todo el mundo se está subiendo por las paredes, e intente encontrar al director de planta. ¡No será fácil! El director de planta estará escondido con el director financiero en algún despacho remoto, intentando, de alguna forma, poner los números derechos. El resultado final es que, después de este monstruoso trabajo, no sabemos si, en realidad, la planta ha ganado dinero este mes. Depende de unas asignaciones que se hicieron hace meses.

¿Existe algún problema para calcular el beneficio neto directamente, igual que se hace para todo el grupo? Ciertamente no. Es mucho más rápido y requiere muchos menos datos. Eso es todo. Entonces, ¿por qué lo

hacemos por el camino difícil, sólo para acabar con la respuesta incorrecta? Parece ser que la tendencia a usar el coste del producto y el beneficio del producto es casi obligatoria. ¿Existe alguna otra razón?

¿Es la pérdida de tiempo y de trabajo el único daño que se hace? No, es mucho más profundo. Supongamos que hemos hecho algo correcto que de verdad ha mejorado nuestra fábrica. Pero nuestras pérdidas y ganancias, calculadas con la técnica de la variación, muestran, durante dos o tres meses seguidos, que la situación se está deteriorando. Probablemente anularemos nuestra acción correcta. La medida del resultado tiene un impacto enorme en la conducta.

Lo más triste de todo es ver a un negocio pequeño creciendo satisfactoriamente gracias a su empresario. Entonces, el negocio alcanza el nivel en el que es necesario un mejor control financiero. El empresario contrata a un financiero profesional: a veces este financiero habrá trabajado para una corporación, para la fábrica. Esta persona trae consigo la técnica de la variación, y el empresario pierde el control de su negocio.

Pero, veamos otro ejemplo en el que se use abundantemente la forma de pensar de la contabilidad de costes, cuando sería suficiente con usar la fórmula directa de pérdidas y ganancias. Supongamos que una empresa está considerando la cuestión contraria a la que provocó la invención de la contabilidad de costes. No el lanzamiento de un nuevo producto, sino la eliminación de alguno de los existentes. Este tipo de ejercicio se hace normalmente a nivel corporativo. ¿Cuáles son los primeros candidatos que se consideran en ese nivel? En efecto, los productos que dan menos beneficio, los perdedores, los «perros». ¿Se han dado ustedes cuenta de que ya estamos utilizando la terminología de la contabilidad de costes?

¿Cómo se hace el cálculo? Normalmente la primera pregunta que se hace es: ¿cuánto es el ingreso neto del producto? No hay problema, simplemente el precio de venta menos el precio de la materia prima. La siguiente pregunta, por supuesto, es: ¿cuánto nos cuesta fabricar este producto? Nos gustaría saber qué beneficio sacamos a este producto. De nuevo, no hay problema. A nivel corporativo, nos limitamos a mirar cuánta mano de obra directa se necesita para hacer este producto. Doce punto setenta y tres minutos. En el nivel corporativo lo saben todo con una exactitud de, por lo menos, cuatro dígitos. En la fábrica no sabemos si en realidad se necesitan 10 ó 15 minutos, pero en el nivel corporativo lo saben todo. Así pues, convertimos este tiempo en dólares, aplicando el precio de la mano de obra y multiplicando el resultado por el factor de distribución de gastos generales: un factor que se calculó basándose en la realidad del año o del trimestre pasado, aunque ahora estemos intentando hacer algún cambio. No importa. Ya tenemos el coste

¿Qué hacemos si el coste se aproxima demasiado al ingreso neto? ¿O,

aún peor, si es más alto? La orden para dejar de fabricar y de vender el producto se envía a la fábrica casi inmediatamente. El personal de la fabrica normalmente se rebela. Tenemos experiencia suficiente. Ya nos hemos quemado más de una vez con el efecto dominó. Hoy recortan este producto. Los gastos generales no van a bajar. Tres meses más tarde, otra lumbrera de la corporación hará los mismos cálculos y perderemos dos productos más. Y pronto, la fábrica se enfrentará a la amenaza de cierre.

Y, sin embargo, resulta tan fácil hacer el cálculo correctamente usando la fórmula original de pérdidas y ganancias: ingreso neto total menos gastos operativos totales. Primero nos tenemos que preguntar lo siguiente: *si dejamos este producto, ¿cuál será el impacto en el ingreso neto total?* Sabemos la respuesta, dependiendo de lo que nos podamos fiar de las previsiones de ventas. En realidad, no hay nada totalmente exacto. Vamos a perder las ventas de este producto en concreto.

Segundo, si dejamos este producto, ¿qué impacto tendrá, no en el coste, sino en el total de los gastos generales? De acuerdo con nuestra definición de gastos generales, lo que quiere decir esta pregunta es: ¿a cuánta gente vamos a despedir? ¿En producción? ¿En envíos? ¿En ingeniería? ¿En contabilidad? Y, por favor, ¡especifiquemos nombres y apellidos! De alguna forma, toda estimación del tipo «vamos a despedir a veinte personas» se reduce a seis o siete cuando dejamos de tratar con números y empezamos a tratar con personas reales.

¿A cuánta gente vamos a despedir? ¿A nadie? ¿Solamente los vamos a cambiar a otro departamento? Sin embargo, ¿habrán notado que, por alguna extraña razón, el cambio de gente de un departamento a otro no afecta a los gastos operativos de la empresa? ¡Aaah!, ¿me dicen que ahora van a ser más productivos en sus nuevos departamentos? Dense cuenta de que siempre que nos sentimos inseguros, cambiamos de terminología. De beneficio a productividad, por ejemplo. ¿Qué significa productivo? ¿Subirán los ingress netos de otros productos? ¿De cuáles? Y, ¿en cuánto? Si no lo sabemos, ¿cómo es que lo usamos como justificación?

Con la eliminación de un producto específico, ¿cuánto bajarán los gastos operativos? ¿A cuántas personas vamos a despedir? Es una pregunta desagradable, pero no resulta difícil contestarla. Si la reducción resultante en los ingresos netos es menor que la reducción en gastos operativos, abandonemos el producto. Si no es así, estamos tomando una decisión que perjudica a la consecución de la meta de la empresa.

Utilizando este método directo, a veces, encontramos que no tiene sentido eliminar un producto específico u otro, pero que sí lo tienen si eliminamos los dos. No podemos despedir a media persona. Pero sí podemos despedir a una persona entera. Ya es hora de darse cuenta de que en

el método convencional de coste del producto estamos intentando ahorrar el 7 por 100 de una máquina o el 13 por 100 de un trabajador. Ya es hora de que volvamos a la realidad.

¿Cuántos negocios ha perdido la industria americana durante los últimos veinte años, que se han trasladado a Méjico y a Filipinas debido a esta forma tan torpe de calcular? Incluso, hoy en día, hay corporaciones que están trasladando productos de una planta a otra basándose en este razonamiento: *es demasiado caro hacer este producto en esta fábrica, los gastos generales son excesivamente altos. Lo trasladaremos a otra fábrica.*

¿Van a despedir a alguien en la primera fábrica? *¡Por supuesto que no! Tenemos un acuerdo de estabilidad con los sindicatos.*

¿Y qué pasa con la otra fábrica?. *Bueno, allí tendremos que contratar a algunas personas más, pero la mano de obra es barata.*

«Coste del producto», «margen del producto» y «beneficio del producto» se han convertido en el lenguaje básico de los negocios industriales. El uso de la contabilidad de costes casi ha forzado el hecho de que todo el negocio esté clasificado «por producto». Los ejemplos que hemos visto se limitan a sacar a la luz el tipo de errores que estamos cometiendo. Solamente exponen el tipo de batallas a las que se enfrenta cualquier directivo que se atreva a utilizar la intuición o el sentido común. Las implicaciones son devastadoras, y alcanzan a todos los aspectos del negocio. Quizá la mejor forma de resumir este capítulo sea repitiendo una breve historia que cuenta Akio Morita, el presidente de Sony, en su libro *Made in Japan.*

Cuando Sony era sólo una empresa muy pequeña, una empresa americana que tenía una cadena de 150 tiendas, ofreció un gran pedido al Sr. Morita. Tenía que presentar una propuesta de precio para 5.000, 10.000, 30.000, 50.000 y 100.000 unidades. Lo mejor es que lean ustedes sus consideraciones de sentido común en el propio libro, pero he aquí la respuesta del director de compras americano cuando recibió los presupuestos: «Sr. Morita, llevo trabajando treinta años como director de compras, y es usted la primera persona que me dice que cuanto más compre, más subirá el precio. Es ilógico». Una respuesta típica de la contabilidad de costes al modo de razonamiento de pérdidas y ganancias.

9

La escala de importancia de las nuevas medidas

Vamos a repasar dónde estamos. Todavía estamos intentando encontrar por qué nuestra intuición eligió una frase tan exigente como *nueva filosofía global de dirección*. Expresamos la meta, al menos en lo referente a negocios que cotizan en bolsa, y no pudimos encontrar nada nuevo. Después, nos sumergimos en el tormentoso tema de las medidas, y nos dimos cuenta de que, aunque en este tema haga falta una buena limpieza, básicamente no hay nada nuevo. Ingresos netos, inventario y gastos operativos se conocían y se usaban mucho antes de que existieran los nuevos movimientos.

Lo único seguro, la única pista que todavía tenemos, es la invalidez de la contabilidad de costes. Quizá haya alguna distorsión en nuestra forma de ver las medidas. ¿Podría ser que lo nuevo no son las medidas en sí, sino la escala de importancia que les asignamos? ¿Existe alguna escala de importancia convencional? Ciertamente, no se ha expresado formalmente, pero, ¿existe en la práctica? Intentemos aproximarnos al tema sistemáticamente.

El juicio definitivo, como ya hemos dicho muchas veces, lo dan el beneficio neto y el retorno de la inversión. Los ingresos netos y los gastos operativos influyen en ambos elementos, mientras que el inventario influye sólo en el último de los dos. Naturalmente, esto sitúa a los ingresos netos y a los gastos operativos en un nivel más importante que el inventario. ¿Y qué hay de la relación entre ingresos netos y gastos operativos? A primera vista, ambos parecen tener la misma importancia, ya que el resultado lo da la diferencia entre ellos. Pero esto no es realmente así. Estamos acostumbrados a dar más importancia a lo que parece más tangible. Recordemos que en el pensamiento convencional de los directivos, los gastos operativos se consideran más tangibles que los ingresos netos. Los ingresos netos dependen de factores externos que no controlamos: nuestros clientes y nuestros mercados. Los gastos operativos se conocen mucho

mejor, están mucho más bajo nuestro control. Por lo tanto, nuestra tendencia natural es situar los gastos operativos a un nivel ligeramente más alto que los ingresos netos.

Al evaluar la escala convencional de importancia, debemos tener cuidado de que no nos influyan los vientos de cambio de los ochenta. Llegados a este punto, no nos permitamos distorsionar el análisis de la escala convencional sólo porque nuestra intuición de que los ingresos netos son dominantes haya sido reforzada en los últimos cinco años. Y menos aún ahora que tenemos que mirar con valor a la realidad que nos rodea.

En cualquier empresa, las medidas dominantes no son las del resultado. Estas sólo son dominantes en la estrecha zona de la alta dirección. Pero, a medida que bajamos por la pirámide, las medidas se disuelven más y más hacia el tipo de medida de la contabilidad de costes. ¿Qué es *coste*? Es sólo un sinónimo de gastos operativos. Así, todos los procedimientos de coste están construidos para que asignen un valor a las acciones que impactan en los gastos operativos. Como resultado de esto, las acciones que tienen su efecto dominante sobre los ingresos netos se clasificarán como intangibles.

Consideremos, por ejemplo, las acciones que tienen como objetivo la mejora del servicio al cliente, o la reducción del tiempo total de fabricación; acciones que, hoy en día, la alta dirección considera como muy importantes. A pesar de esto, cuando se quiere invertir en equipo para conseguir esos resultados, y el equipo no reduce, además, los costes, la dirección de nivel medio se encuentra en una posición difícil. Cuando elaboran la petición de fondos para el equipo que se necesita para el proyecto, deben justificarlo bajo el encabezamiento de intangible. Ya hemos aprendido que no tangible no es sinónimo de no importante. Intangible es sólo una expresión que usamos cuando no podemos asignar un valor numérico. La contabilidad de costes obliga a usar este título para toda acción que esté dirigida a aumentar los ingresos netos futuros. El coste, al ser casi sinónimo de gastos operativos, está ciego en lo que respecta a los ingresos netos.

A pesar de los esfuerzos de la alta dirección, se considera que los gastos operativos son mucho más tangibles que los ingresos netos, y se sitúan en una primera posición dominante en la escala de importancia. Los ingresos netos les siguen a mucha distancia. Limítese a echar un vistazo a lo que piden los bancos y otros prestatarios cuando una empresa tiene que presentar un plan de recuperación. Los recortes, la reducción de los gastos operativos, son obligatorios.

¿Y qué ocurre con el inventario? Debido al difícil concepto de valor añadido al producto, la reducción de inventario perjudica al resultado, en vez de beneficiarlo. El inventario queda en la escala en un tercer lugar muy

distante. La escala convencional de importancia es pues: primero, los gastos operativos, los ingresos netos en un segundo puesto, y el inventario en un remoto tercer puesto. Para los nuevos movimientos, esta escala de importancia es como enseñar un capote a un toro. Llenos de celo, se toman grandes molestias para atacar y condenar esta escala. Vamos a aclarar la escala de importancia de nuestras medidas que recomiendan los tres movimientos, JIT, TOC y TQM. Los tres tienen en común una frase dominante. Si hablamos con los discípulos de cualquiera de ellos, oiremos siempre la misma canción: *un proceso de mejora continua*. De hecho, deberíamos haber tenido este lema desde hace mucho. Recordemos que la meta de la empresa no es sólo hacer dinero, es hacer *más* dinero, ahora y en el futuro. El proceso de mejora continua se deriva directamente de la definición de la meta.

¿Cuál de las tres avenidas, ingresos netos, inventario y gastos operativos, parece más prometedora, si lo que estamos buscando es un proceso de mejora continua? Si pensamos un momento, la respuesta es tan clara como el agua. Nos esforzamos en disminuir tanto el inventario como los gastos operativos. Por tanto, ambos ofrecen sólo una oportunidad limitada para mejorar continuamente. Ambos están limitados por cero.

No es este el caso de la tercera medida, los ingresos netos. Nos esforzamos por aumentarlos. Los ingresos netos no tienen ninguna limitación intrínseca, deben ser la piedra angular de todo proceso de mejora continua. Deben ser los primeros en la escala de importancia.

Entonces, ¿qué pasa con el inventario y los gastos operativos? ¿Cuál de los dos es más importante? A primera vista no parece que nuestro análisis previo tenga ningún fallo. Los gastos operativos influyen en las dos medidas del resultado, mientras que el inventario sólo influye directamente en una de ellas. Pero esta argumentación no puede ser definitiva, porque nuestra elección de beneficio neto y retorno de la inversión fue completamente arbitraria. Si hubiéramos elegido la pareja alternativa, productividad y rotación del inventario, esta argumentación caería por sí misma.

El fallo básico que se comete al situar los gastos operativos por encima del inventario se deriva del hecho de que sólo hemos tenido en cuenta el impacto *directo* del inventario, y no el *indirecto*. La sabiduría convencional estaba totalmente concentrada en los gastos operativos, y por eso reconocía solamente uno de los canales indirectos del inventario: la forma en que éste influye en el resultado a través de los gastos operativos. Si nos referimos a la parte del inventario que incluye a las máquinas, este canal indirecto se llama depreciación. Si nos referimos a la parte del material, se llama coste financiero.

Los tres nuevos movimientos han reconocido la existencia de otro canal indirecto considerablemente más importante. El canal indirecto por el que

el inventario, y especialmente la parte del mismo que se relaciona con tiempo, afecta a los ingresos netos futuros. En el libro *La carrera*, totalmente dedicado al tema del inventario, Robert Fox y yo dedicamos más de treinta páginas a describir y demostrar la existencia de este canal indirecto. «La carrera» muestra claramente cómo el inventario casi llega a determinar la capacidad futura de una empresa para competir en sus mercados. Este impacto indirecto resulta tan importante que los tres movimientos han situado al inventario en segundo lugar en la escala de importancia, dejando a los gastos operativos en un cercano tercer puesto.

Así, la nueva escala de importancia es totalmente distinta de la convencional. Los ingresos netos son dominantes. Lo segundo es el inventario, y los gastos operativos han caído desde su posición preeminente hasta un modesto tercer lugar.

Es impresionante el impacto que tiene esta nueva escala de importancia en todas las decisiones que adopta la dirección. Bastantes acciones que tienen mucho sentido bajo la escala convencional resultan totalmente absurdas cuando se observan desde esta nueva perspectiva. Por eso, los tres movimientos se han tomado tantas molestias para predicar lo que parecen ser solamente trivialidades de sentido común.

La *teoría de las limitaciones* machaca una y otra vez: *la suma de óptimos locales no da como resultado el óptimo del total. Calidad total* nos recuerda que: *hacer las cosas correctamente no es suficiente. Es más importante hacer las cosas que son correctas.* Y *justo a tiempo* levanta su bandera: *no hagas lo que no se necesita.*

10

El cambio de paradigma resultante

¿Qué fue lo que sucedió en realidad cuando depusimos al gasto operativo como rey de la montaña, y lo cambiamos por los ingresos netos? El conocimiento de la magnitud de este cambio sólo está empezando a emerger ahora. De hecho, es el cambio de considerar nuestras organizaciones como sistemas de variables independientes a considerarlas como sistemas de variables dependientes. Este es el mayor cambio que se pueda imaginar cualquier científico.

Intentemos digerirlo, desnudándolo de su «glamour» científico. Preguntémonos: ¿cuántas salidas de gasto operativo existen en una empresa? Todo trabajador es una salida, todo ingeniero, vendedor, administrativo o director es una salida de gasto operativo. Cada trocito de chatarra, cada sitio donde consumimos energía, es una salida de gasto operativo. Este es un mundo donde casi todo es importante. Este es *el mundo del coste.*

Por supuesto, no todo es importante en el mismo grado. Hay cosas que son más importantes que otras. Aun en el mundo del coste, reconocemos el principio de Pareto, la regla del 20-80. El 20 por 100 de las variables son responsables del 80 por 100 del resultado final. Pero esta regla es correcta, estadísticamente, sólo cuando estamos tratando con sistemas de variables independientes. El mundo del coste nos da la impresión de que nuestra organización es un sistema de ese tipo, de que las salidas de gastos operativos casi no están conectadas. El dinero se fuga por muchos agujeros grandes y pequeños.

Veamos ahora el cuadro cuando percibimos los ingresos netos como dominantes. Muchas funciones tienen que llevar a cabo, sincronizadamente, muchas tareas, hasta que se realiza una venta o hasta que se gana beneficio neto. El mundo de los ingresos netos, o mundo del valor, es un mundo de variables dependientes.

En el mundo del valor, hasta el principio de Pareto se tiene que enten-

der de una forma completamente distinta. Ya no es la regla del 20-80. Ahora se acerca mucho más a la regla del 0,1-99,9. Una pequeña fracción (el 0,1 por 100) de las variables determina el 99,9 por 100 de los resultados. ¿Parece extraño? ¿Casi increíble? Solamente intente recordar lo que ya sabía. Aquí estamos tratando con acciones en cadena. ¿Qué es lo que determina el rendimiento de una cadena? El rendimiento de la cadena está determinado por la fuerza de su eslabón *más débil*. ¿Cuántos eslabones más débiles existen en una cadena? Mientras las fluctuaciones estadísticas impidan que todos los eslabones sean idénticos, sólo habrá un eslabón más débil en cada cadena.

¿Qué nombre es apropiado para el concepto de eslabón más débil, el eslabón que limita la fuerza global (el rendimiento) de la cadena? *Limitación* es un nombre muy adecuado. ¿Cuántas limitaciones existen en una empresa? Esto dependerá de cuántas cadenas independientes existan. No pueden ser muchas. No son sólo los productos los que crean cadenas, combinando entre sí distintos tipos de recursos. También los recursos crean cadenas, combinando entre sí diferentes productos.

La analogía adecuada en nuestras organizaciones sería una parrilla, mejor que una cadena. En cualquier caso, hay una inmensa cantidad de interacciones entre las variables. Estas interacciones, junto con las fluctuaciones estadísticas, son un factor casi dominante en toda organización, lo que impide que una organización tenga muchas limitaciones. En la práctica 0,1-99,9 es probablemente una subestimación.

¿Están nuestros directivos gestionando los negocios de acuerdo con el proceso de enfoque que resulta obligatorio en el mundo del valor? Desgraciadamente la respuesta es: no. La mayoría de ellos se quejan de que tienen que dedicar más de la mitad de su tiempo a apagar fuegos. Ciertamente son directivos del mundo del coste. Todo, o al menos el 20 por 100 de todo, es importante. Su atención se encuentra demasiado dispersa, en demasiados problemas aparentemente igual de importantes.

Algunos directivos, incluso, se quejan de que se ven a sí mismos como inmersos en una piscina llena de pelotas de ping-pong, intentando mantenerlas a todas bajo el agua. Si todas las pelotas tienen que estar bajo el agua, si todo es importante, entonces, el descubrimiento del mundo del valor todavía no ha llegado a estos directivos.

JIT y TQM no están ayudando mucho a la hora de estimular la necesidad del cambio. En efecto, son muy activos a la hora de forzar a la dirección a cambiar a la nueva escala de importancia, pero no han hecho mucho para ayudar a los directivos a cambiar al nuevo estilo requerido para tratar con la nueva escala.

TQM, dándose cuenta de la importancia de los ingresos netos, ha cambiado la percepción de los directivos sobre las acciones que se deben

ejecutar. Si no fuera por TQM, el servicio al cliente y la calidad del producto, de vital importancia para aumentar los ingresos netos futuros, no estarían entre sus prioridades principales, como están hoy en día.

Si no fuera por JIT, el inventario todavía se consideraría como un activo. No se reconocería la importancia de reducir el tiempo total de fabricación, el tamaño de los lotes y la duración de los cambios de modelo, ni la de mejorar el mantenimiento preventivo. Todas las acciones que conducen a dar una respuesta más rápida al mercado, todas las acciones esenciales para garantizar los ingresos netos futuros, no habrían llegado hasta las salas de consejo.

De lo que no se han dado cuenta, ni TQM, ni JIT, es de las ramificaciones del mundo del valor; del enfoque que surge directamente del conocimiento de que estamos tratando con entornos de variables interactivas. ¿Es posible que sea tan importante adherirse a cada detalle de las especificaciones de diseño del producto (especialmente cuando de entrada nadie sabe si las tolerancias que se especifican tienen que estar ahí)? ¿Es posible que sea tan importante reducir los tiempos de cambio de modelo en todas las máquinas? ¿O tener la máxima fiabilidad posible en todos los recursos? Estos son los conceptos que se extrapolan incorrectamente desde la perspectiva anterior del mundo del coste.

En la actualidad, la situación es bastante asombrosa. Por un lado, nos hemos dado cuenta de que hacen falta tantos cambios drásticos, pero, por otro lado, no nos guiamos con el proceso de enfoque necesario. Empieza a parecer que muchas empresas no tienen bastante con la excitación del síndrome de final de mes, y han decidido añadirle lo que solamente podemos denominar como el nuevo proyecto de mejora para el próximo trimestre.

¿Qué tratamiento dan a la contabilidad de costes? A TQM simplemente le irrita. Le irrita el hecho de que las inversiones para mejorar la calidad, que se hacen para conseguir ganancias muy importantes en los ingresos netos, tengan que justificarse a base de consideraciones de coste mucho menos importantes. Han solucionado este problema limitándose a empujar a un lado las medidas financieras, y afirmando que la calidad es su trabajo número uno.

Básicamente, JIT ha hecho lo mismo. Cuando conocí al Dr. Ohno, el inventor de *Kanban*, el sistema JIT de Toyota, me dijo que había tenido que luchar contra la contabilidad de costes durante toda su vida. «No fue suficiente con echar de las fábricas a los contables de costes, el problema era sacar la contabilidad de costes de las mentes de mis empleados.»

Debemos explorar las ramificaciones que brotan de una percepción de la realidad totalmente nueva, una percepción en la que hay muy pocas cosas que sean de verdad importantes. ¿Qué se debe cambiar al pasar del

mundo del coste al mundo del valor? Pero, antes de hacer exactamente eso, no debemos olvidar que hemos dejado pendiente de resolver un enorme problema. Las medidas financieras son esenciales mientras la meta de nuestra empresa sea ganar dinero ahora y en el futuro, no podemos contiuar sin ellas. Si abandonamos la contabilidad de costes nos quedaremos sin una forma numérica de juzgar muchos tipos de decisiones. Esto abre la puerta a las medidas no financieras, que ya están empezando a colarse.

JIT se equivoca al ignorar este problema. TQM es aún peor, porque recomienda la utilización de medidas no financieras. Recordemos que una de las bases para dirigir una organización es la capacidad de juzgar cómo impactará una decisión local en el resultado. Si intentamos medir con tres o más medidas no financieras, básicamente habremos perdido el control. Las medidas no financieras equivalen a la anarquía. Simplemente, no se pueden comparar naranjas, manzanas y plátanos, y, desde luego, no se pueden relacionar con el resultado. La meta es ganar dinero. Por definición, toda medida debe contener el signo del dinero.

Recordémonos que condenar las soluciones que aportó la contabilidad de costes puede ser muy importante, pero esta acción no nos da, por sí misma, una solución al problema. El problema original todavía subsiste. ¿Cómo podemos juzgar el impacto de una decisión que estamos considerando? ¿Cómo podemos juzgar, por ejemplo, si debemos lanzar o no un producto nuevo? ¿Cuál será su influencia en el resultado?

Intentemos atacar este problema. Supongamos que estamos tratando la cuestión de lanzar un nuevo producto cuando tenemos exceso de todo. Tenemos exceso de mercado, de clientes y de recursos. Cuando se da esta situación, ¿qué impacto tendrá el lanzamiento del nuevo producto en las ventas de los demás? Absolutamente ninguno.

¿Qué impacto tendrá el lanzamiento del nuevo producto en los gastos operativos? En el caso descrito, ninguno, ya que tenemos recursos excedentes en todas las áreas, recursos que ya estamos pagando. ¿Cuándo tendrá influencia en otras cosas el lanzamiento de un nuevo producto? Cuando no tengamos bastante. Si no tenemos bastante mercado y lanzamos un nuevo producto, dirigido a los mismos clientes para satisfacer las mismas necesidades que ya satisface el resto de nuestros productos, debemos esperar una reducción de las ventas de esos productos.

Si el producto nuevo necesita un recurso que no tiene bastante capacidad, la única forma posible de ofrecerlo es, o bien reduciendo la oferta de los demás, o aumentando las inversiones y los gastos operativos para conseguir que ese recurso tenga más capacidad.

En resumen, el único caso en el que de verdad nos enfrentamos a un problema, el único caso en el que el lanzamiento de un producto nuevo influye en otras cosas, es cuando hay algo de lo que no tenemos bastante. ¿Cómo llamamos a algo de lo que no tenemos bastante, hasta el punto de que limita el rendimiento de toda la empresa? *Limitación* es un nombre adecuado. Otra vez, la misma palabra.

En el mundo del valor, la clasificación esencial se hace por limitaciones, sustituyendo el papel que desempeñaban los productos en el mundo del coste. Parece que si continuamos explorando las ramificaciones del mundo del valor podremos resolver el problema de encontrar un sustituto para la contabilidad de costes.

11

Formulación del proceso de decisión del mundo del valor

Si pretendemos enfocarlo todo, en realidad no conseguiremos enfocar nada. Enfocar significa: tengo esta gran área bajo mi responsabilidad. Yo decido concentrar la mayoría de mi atención en una pequeña parte de ella. Dispersar la atención, por igual, a todas las partes del área significa que no hay concentración, que no hay enfoque.

En el mundo del coste, es muy difícil concentrarse. En el mejor de los casos, tenemos que enfocar una parte muy grande de los detalles. No es este el caso en el mundo del valor. ¿Cuál debería ser el primer paso? ¿Dónde debemos concentrarnos? ¿No es totalmente obvio? En los eslabones más débiles, en las limitaciones. Ellos son los que determinan el rendimiento global de la empresa.

En vista de lo anterior, ¿cuál sugeriría usted que debe ser el primer paso? Sí, está claro, lo primero de todo es encontrar todas las limitaciones del sistema.

En cualquier caso, ¿tenemos garantías de que vamos a encontrar algo? En otras palabras, ¿es obligatorio que todo sistema tenga, al menos, una limitación? Quizá esté más clara la respuesta si hacemos la misma pregunta con distintas palabras: ¿Ha visto alguna vez alguna empresa sin una sola limitación? La respuesta intuitiva es obvia: Nunca. En toda cadena tiene que haber un eslabón más débil. Pero, vamos a intentar establecerlo un poco mejor. Si existe una empresa sin limitaciones, ¿qué significa esto? Que no hay nada que limite su rendimiento. ¿Cuál será el rendimiento de esta empresa? ¿Cómo serán su beneficio neto y su retorno de la inversión? Infinitos. ¿Han visto alguna vez alguna empresa con beneficio neto infinito? La conclusión es obvia: Todo sistema debe tener, al menos, una limitación. Por otra parte, en la realidad, todo sistema debe tener un número muy pequeño de limitaciones. Por tanto, el primer paso de enfoque de la *teoría de las limitaciones* es intuitivo:

1. *Identificar las limitaciones del sistema.*

(A pesar de que hemos expresado «limitaciones» en plural, debemos recordar que puede haber sistemas que tengan solamente una limitación.)

Identificar una limitación supone que ya tenemos alguna apreciación de la magnitud de su impacto sobre el rendimiento total. Si no es así, podríamos tener algunas trivialidades en la lista de limitaciones, lo que yo llamo «choopchicks».

¿Es importante asignarles prioridades de acuerdo con su impacto? No necesariamente. Lo primero, recordemos que en este punto todavía no tenemos estimaciones precisas. Lo segundo que hay que recordar es que el número de limitaciones es muy reducido. En cualquier caso, tenemos que tratarlas todas, así que no perdamos el tiempo desperdiciando esfuerzos.

Lo que cuenta es identificar las limitaciones.

¿Cuál debe ser el paso siguiente? Ya hemos identificado las limitaciones. Hemos encontrado esos puntos, esas cosas de las que no tenemos suficiente, hasta el punto de que limitan el rendimiento global de todo nuestro sistema. ¿Cómo debemos gestionarlas?

La respuesta intuitiva es que hay que librarse de ellas. Pero todos sabemos que, a veces, se tarda mucho en librarse de una limitación. Por ejemplo, si la limitación es el mercado, podemos tardar muchos meses, incluso un año, en romper esta limitación. Si la limitación es una máquina y hemos decidido comprar otra, puede que el plazo de entrega sea de más de seis meses. ¿Qué vamos a hacer mientras tanto? ¿Sentarnos sin hacer nada? No parece un consejo muy bueno para el segundo paso.

¿Cómo debemos gestionar las limitaciones, las cosas de las que no tenemos bastante? Al menos, no las desperdiciemos. Vamos a exprimirlas al máximo. Cada gotita cuenta. Poniéndolo de una forma más civilizada, el segundo paso de la *teoría de las limitaciones* es:

2. *Decidir cómo explotar las limitaciones del sistema.*

Explotar significa simplemente sacarles el máximo provecho. He elegido deliberadamente una palabra con connotaciones ligeramente negativas. *Explotar*, haciendo lo que haga falta para ello. Tenemos que entender bien lo siguiente: no creo que exista seguridad en el empleo en una empresa que esté perdiendo dinero. En una empresa que pierda dinero, a pesar de lo que pueda decir la dirección, la seguridad del empleo está amenazada. Aquí tenemos las limitaciones, las que limitan el rendimiento global. La seguridad del empleo de toda la plantilla depende del rendimiento de esos puntos. Hay que exprimirlos al máximo, sin piedad.

Por ejemplo, supongamos que la limitación está en el mercado. Hay

suficiente capacidad, pero no hay suficientes pedidos. Entonces, explotar la limitación significaría: 100 por 100 de las entregas dentro del plazo. No el 99 por 100, ¡el 100! Si el mercado es la limitación, no debemos desperdiciar nada.

Bien, ya hemos decidido cómo vamos a gestionar las limitaciones. ¿Qué pasa con la gestión de la inmensa mayoría de los recursos de la empresa que, por definición, no son limitaciones? ¿Debemos dejarlos solos? En un período de tiempo muy corto habrán dejado de trabajar correctamente, y su disponibilidad se reducirá hasta tal punto que se convertirán en limitaciones. ¿Cómo debemos gestionarlos?

La respuesta es intuitivamente obvia. En el paso anterior, decidimos un plan de acción que obtendría el máximo rendimiento posible en la situación actual, pero, para conseguirlo, la limitación necesita consumir cosas. Si los recursos no limitados no suministran lo que necesita consumir la limitación, la decisión tomada se quedará en el papel, nunca se llevará a cabo. ¿Debemos alentar a los no limitados a que suministren más de lo que las limitaciones pueden absorber? Esto no ayudará a nadie. Por el contrario, hará daño. Así pues, los no limitados deben suministrar todo lo que necesitan consumir las limitaciones, pero no más. Escribamos el tercer paso del proceso de enfoque:

3. *Subordinarlo todo a la decisión anterior.*

Ya hemos llegado a un punto en el que estamos gestionando nuestra situación actual. ¿Es este el último paso? Por supuesto que no. Las limitaciones no son un acto divino, se puede hacer algo con ellas. Ahora es el momento de que hagamos lo que antes nos tentaba tanto. Vamos a abrir las limitaciones. El que no tengamos lo suficiente no significa que no podamos sumar algo más. El paso siguiente es intuitivamente obvio:

4. *Elevar las limitaciones del sistema.*

Elevar significa levantar la restricción. Este paso es el cuarto, no el segundo. Muchas veces hemos sido testigos de situaciones donde todo el mundo se quejaba de una gran limitación, y cuando se aplicaba la exploración del segundo paso, simplemente, no desperdiciar lo que hay disponible, resultaba que la limitación desaparecía. Así que, no salgamos corriendo apresuradamente a subcontratar trabajos, o a lanzar una sofisticada campaña de publicidad, etc... Cuando hayamos completado el segundo y el tercer paso, y todavía tengamos una limitación, entonces será el momento de dar el cuarto paso. A menos que estemos tratando con un caso muy claro, donde la limitación está desproporcionada con respecto a todo lo demás.

Con el cuarto paso también hemos conseguido mover la empresa hacia adelante. ¿Podemos pararnos aquí, o debemos añadir un quinto paso? Una vez más, la respuesta es intuitivamente obvia. Si elevamos la limitación, si añadimos más y más de las cosas de las que no teníamos bastante, debe llegar un momento en que ya tengamos lo suficiente. Se ha roto la limitación. Aumentará el rendimiento de la empresa, pero, ¿saltará hasta el infinito? Obviamente no. El redimiento de toda la empresa se verá ahora restringido por alguna otra cosa. La limitación habrá cambiado de sitio. Así que el quinto paso es:

5. *Si en los pasos previos se ha roto una limitación, hay que volver al primer paso.*

Pero esto no es todo el quinto paso. Debemos añadirle una advertencia muy importante. La limitación impacta en el comportamiento de todos los demás recursos de la empresa. Todo se debe subordinar al nivel de máximo rendimiento de la limitación. Por tanto, debido a la existencia de la limitación, surgen muchas reglas en la empresa, tanto formales como intuitivas. Ahora se ha roto una limitación. Resulta que, en la mayoría de los casos, no retrocedemos para volver a examinar esas reglas. Quedan atrás. Hemos creado una limitación *política*. Así pues, debemos ampliar el quinto paso:

6. *Si se ha roto una limitación en los pasos anteriores, hay que volver al primer paso, pero no hay que permitir que la inercia provoque una limitación del sistema.*

No puedo exagerar lo suficiente la importancia de la advertencia que hemos añadido. En la mayoría de las empresas que he analizado, no he encontrado limitaciones físicas. He encontrado limitaciones políticas. Nunca he visto una empresa que tenga su limitación en el mercado. He visto muchas que tienen limitaciones políticas en su marketing. Es muy raro encontrar una empresa que tenga una verdadera limitación de capacidad, un verdadero cuello de botella, pero es frecuente encontrar empresas que tienen limitaciones políticas en producción y en logística. Por cierto, este es el caso que se describe en *La meta*. ¿Era el horno una limitación de capacidad? ¿Compró Alex Rogo un horno nuevo? En absoluto. Se limitó a cambiar algunos procedimientos internos de producción y de logística. Al poco tiempo le rebosaba capacidad hasta por las orejas.

Excepto en dos casos, nunca he visto limitaciones de proveedores, aun cuando la mayoría de las empresas se quejan de ello. Sin embargo, he visto limitaciones políticas en las compras que han resultado devastadoras.

Resulta muy significativo que cada vez que nos sumergimos para encontrar las razones que hay tras estos procedimientos tan torpes (y la verdad es que algunas veces casi hay que organizar una expedición arqueológica), nos encontramos con que hace más o menos treinta años, cuando estos procedimientos se pusieron en práctica, tenían mucho sentido. Hace mucho que han desaparecido las razones que los originaron, pero los procedimientos siguen con nosotros. Cinco pasos: un procedimiento de enfoque sencillo, intuitivamente obvio. Todo el mundo los conocía ya, todos se dan cuenta de que tienen sentido. No es de extrañar, ya que nuestra intuición surge de nuestra experiencia en el mundo real, y nuestro mundo real es el mundo del valor. A pesar de esto, ¿de verdad usan los directivos estos pasos? Puede que los usen en alguna emergencia. ¿Y en otros casos? La «garra» de la educación del mundo del coste es demasiado fuerte. A pesar de nuestra clara intuición de sentido común, nuestras acciones se ven mucho más dirigidas por los procedimientos formales del mundo del coste que por los cinco pasos de enfoque, directos y totalmente obvios, del mundo del valor.

12

¿Cuál es el eslabón perdido? Construcción de un experimento decisivo

Hemos hecho lo que yo aconsejé, hemos gastado horas y docenas de páginas para encontrar lo que era nuevo en las *nuevas filosofías globales de dirección*. Bien, pero, ¿hasta dónde hemos llegado? Parece que no hemos avanzado ni un solo paso hacia el problema real. ¿Cómo podemos diseñar un sistema de información?

Así que, dejemos esta desviación y volvamos al centro de nuestro problema. Empecemos con la pregunta que todo directivo dice que no se puede contestar por falta de información.

La meta de la empresa es hacer más dinero ahora y en el futuro. ¿Cuál será el beneficio neto de nuestra empresa el próximo trimestre? ¿No es ésta una de las preguntas más importantes? No, no queremos una estimación, queremos una respuesta exacta con un error de, digamos, dos centavos. ¿Podemos contestarla? No, no hay suficiente información. Esta es la respuesta normal.

¿Qué es lo que nos impide contestar exactamente cuánto beneficio neto vamos a tener el próximo trimestre? Oh, muchas cosas. Por ejemplo, no sabemos qué fiabilidad tiene nuestra previsión de ventas. Y los pedidos en firme que tenemos no son exactamente firmes. A veces, nuestros clientes tienen tendencia a cambiar de opinión. Qué vamos a hacer, ¿ponerles una denuncia?

Pero el problema no se presenta sólo con la información de marketing. También podemos tener problemas internos. Nadie garantiza que no se vaya a romper una máquina. De hecho, podemos garantizar que se va a romper una máquina, lo que no sabemos es cuál, ni cuándo, ni durante cuánto tiempo. Nuestros proveedores no son totalmente fiables, muchas veces no entregan a tiempo, o envían cantidades equivocadas. Hay veces que todo un envío, cuando llega, resulta ser defectuoso. Y no sé cómo estará la cosa en su empresa, pero nuestro personal no es exactamente

muy fiable, y tenemos problemas de absentismo. Tenemos piezas malas, en parte por los procesos, en parte por los trabajadores. Y nuestros supervisores no son totalmente disciplinados, les decimos lo que tienen que hacer, pero ellos creen que saben más que nosotros. La lista puede seguir, y seguir, y seguir. ¿Es falta de información? Más bien parece una lista de quejas:

Los clientes cambian de opinión.
Vendedores poco fiables.
Procesos poco fiables.
Máquinas poco fiables.
Personal sin entrenamiento.
Gestión poco disciplinada.

Si miramos esta lista, hay algo que de verdad nos empieza a preocupar. Todos sabemos cómo se reconoce una excusa. Una excusa se reconoce por: «es culpa de otro». ¿Han notado qué es lo que tiene esta lista en común? Sí. El responsable es otro. Los clientes, los proveedores, las máquinas, el personal... Nosotros somos perfectos, ellos tienen la culpa. ¿No creen que es un poco sospechoso?

¿Es esto una lista de razones por las que no podemos responder a la pregunta de cuánto beneficio neto vamos a tener el próximo trimestre, o es sólo una lista de excusas? Esta pregunta es muy importante, porque si repasamos la lista podremos ver que es un resumen muy bueno de todos los esfuerzos que estamos haciendo actualmente para mejorar nuestra empresa.

Nos estamos esforzando por mejorar nuestra previsión de ventas. Se están haciendo grandes esfuerzos para mejorar las relaciones con los clientes, y tenemos un amplio programa llamado *programa de proveedores*. En cuanto a nuestras máquinas, nos hemos embarcado decididamente en el mantenimiento preventivo, y también hemos invertido mucho en nuevos equipos para mejorar la fiabilidad. En cuanto a los procesos, estamos entrenando y volviendo a entrenar a cada trabajador en los métodos del control estadístico de procesos, etc.

Si esto sólo es una lista de excusas, y no es el problema real, no nos estamos enfrentando a un problema, sino a dos. El primero es que estamos utilizando como excusa la falta de información y, por tanto, puede que la razón de que no tengamos suficiente información derive simplemente de que no la hayamos definido correctamente. El segundo problema es la diversidad de enfoques con los que la empresa pretende conseguir mejoras. Puede que sea un error. ¿Cómo podemos comprobarlo?

Puede que la mejor forma sea un experimento «Gedunken» otra vez. Supongamos que nuestros esfuerzos actuales de mejora tienen más éxito

de lo que nunca habríamos podido soñar. Supongamos que hemos resuelto cada problema de la lista con un éxito espectacular. Ya no existe en nuestra fábrica ninguno de los problemas de la lista anterior. Ahora tenemos lo que algunos podrían llamar una fábrica perfecta. Todo está arreglado, cada dato se conoce con precisión. ¿Tenemos la información? ¿Sabemos con precisión cuánto beneficio neto va a tener nuestra empresa en el próximo trimestre?

Vamos a describir nuestra fábrica perfecta. Vamos a dar todos los datos que alguien pueda pensar que son necesarios. En nuestra fábrica, hemos racionalizado nuestra gama de productos hasta dejarla sólo en dos: los vamos a llamar P y Q. Son productos muy buenos, y nuestro personal está muy bien entrenado para fabricarlos. Nuestra tasa de defectos es cero, no una parte por millón. Cero.

El precio de venta de estos dos productos se ha fijado hasta el centavo. Hemos superado el síndrome de las ofertas que tiene todo vendedor. Ahora son personas disciplinadas. ¿Pueden imaginarse un mundo así? El precio de venta de P es, digamos, 90 dólares por unidad, y el de Q un poco más alto, 100 dólares por unidad.

¿Qué hay de la previsión de ventas? Aquí nos espera una gran sorpresa. La previsión ya no está hecha a ojo. Es precisa hasta la última unidad. Llamamos a esta previsión el *potencial del mercado*. El potencial del mercado para P es de 100 unidades por semana, y para Q, de sólo 50 unidades por semana. Vamos a aclarar lo que queremos decir con potencial de mercado. No es lo que nos hemos comprometido a entregar. Somos tan buenos que no nos tenemos que comprometer a nada. Estos números representan lo que el mercado nos comprará, si nosotros lo servimos. Por supuesto, el hecho de que P tenga un potencial de mercado de 100 unidades por semana significa que, si producimos más de 100, se nos quedarán colgadas como inventario de producto acabado.

Ahora vamos a ver los datos de ingeniería. El producto P se fabrica ensamblando una pieza, que compramos fuera, con dos piezas que fabricamos en casa. Cada una de las piezas que nosotros fabricamos se hace a partir de material comprado, en dos procesos distintos (véase Fig. 12.1).

Fíjense en que esta misma estructura podría describir diferentes entornos, ya sean un diagrama de diseño de producto, un proyecto, o, incluso, un proceso de decisión. Todo tiene el mismo aspecto. Tenemos que ser fieles a una terminología específica, porque, si no, nada se entenderá claramente, pero esto no significa que necesariamente estemos tratando con un entorno de producción. Lo que de verdad estamos intentando describir aquí es el caso genérico de utilizar recursos para realizar tareas con objeto de conseguir un objetivo predeterminado. Ahora, necesitamos algunos datos numéricos. Esto, ciertamente, nos va a obligar a meternos en

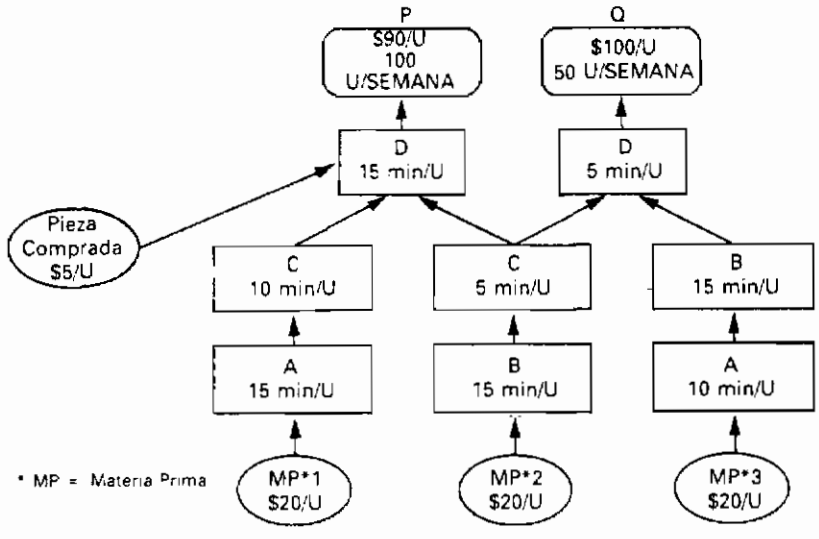


Fig. 12.1. *Nuestra empresa artificial de la que se han eliminado todas las incertidumbres.*

una terminología más específica, pero no olvidemos que esto es un ejemplo de una situación mucho más genérica.

Supongamos que el precio que pagamos por la pieza comprada es de 5 dólares por unidad, mientras que el precio de la materia prima es, en ambos casos, 20 dólares por unidad. El primer material empieza su «viaje» a través del departamento A. Podría ser un ingeniero de tipo A, el almacén situado en A, el vendedor de la región A, o un directivo de nivel A... En este experimento estamos tratando con un entorno de fabricación, así que vamos a usar el término para un trabajador del oficio A. Y vamos a suponer que este trabajador tarda 15 minutos en procesar una unidad. Por supuesto, si estuviéramos en un entorno de proceso, hablaríamos de piezas por hora o, en ingeniería, usaríamos días o semanas (y rezaríamos para que no fueran años). El entorno determina la terminología. Aquí, la que estamos usando es minutos por unidad.

La primera operación de la segunda materia prima la hace otro tipo de trabajador, un trabajador del oficio B, y tarda exactamente lo mismo, 15 minutos por pieza. El segundo paso del proceso para ambas piezas lo hace un tercer tipo de trabajador, un trabajador del oficio C. Tarda 10 minutos en procesar la primera pieza, y sólo 5 minutos en la segunda. Por supuesto, esto implica que el trabajador del oficio C no se dedica a producir un solo tipo de pieza, sino que es multifuncional. ¿Tienen ustedes recursos multifuncionales? ¿No están seguros? ¿Tienen cambios de modelo? Si la

respuesta es afirmativa, tienen ustedes recursos multifuncionales. En nuestro caso, el tiempo de cambio de modelo es cero. Somos tan buenos que hemos reducido todos los tiempos de cambio a cero. No a un segundo, sino a cero.

El ensamblaje lo hace el trabajador D. Tarda 15 minutos en ensamblar una unidad. Con esto ya están completos los datos del producto P. Ahora, vamos a describir el producto Q.

El producto Q sólo se compone de dos piezas. Como estamos embarcados en la tecnología de grupo, estamos intentando usar la mínima cantidad de diseños distintos, así que, en su ensamblaje, usamos la segunda pieza de P, y otra pieza que se procesa en nuestra fábrica, en dos pasos distintos (véase Fig. 12.1). Por supuesto, esto hace que la pieza central sea común a dos productos distintos, un caso muy frecuente en la industria. De todas formas, vamos a aclararlo un poco. Para poder servir un P y un Q, necesitamos dos piezas centrales. ¿Por qué lo subrayamos? Porque, por ejemplo, en ingeniería de diseño esta misma situación implicaría que necesitamos diseñar la pieza central una sola vez, aunque se necesite tanto en el diseño de P como en el de Q. Hasta la interpretación del diagrama de flujo nos la dicta el entorno. Ahora, vamos a completar los datos: la materia prima para la tercera pieza se compra al mismo precio que las otras dos, 20 dólares por unidad. Digamos que la primera operación del proceso la hace el mismo trabajador A que hace la primera pieza. (Hemos lanzado en nuestra fábrica un intenso programa de *enriquecimiento del trabajo.)* Tarda 10 minutos en procesar una unidad de la tercera pieza. La segunda operación la hace el trabajador B, el mismo trabajador B que hacía la primera operación de la segunda pieza, y tarda el mismo tiempo, 15 minutos por unidad. El ensamblaje lo hace el mismo trabajador D de ensamblaje, pero, en este caso, solamente tarda 5 minutos en ensamblar una unidad.

En nuestra fábrica hay trabajadores con cuatro oficios distintos: A, B, C y D. Aunque hemos impartido mucho entrenamiento interdisciplinario, todavía tenemos cuatro tipos de recursos diferentes. Creo que nunca llegaremos al punto en el que todos sabrán hacer de todo. ¿Dónde encontraremos al genio que sea capaz de convencer a un torno para que también suelde? Pero, aunque pudiéramos conseguir eso, seguro que no podríamos encontrar a nadie que convenza al ingeniero jefe para que barra el suelo de otro departamento. Así que, seguimos teniendo distintos tipos de oficios, incluso en nuestra fábrica ideal. Ya hemos entrenado a todos los que eran susceptibles de ser entrenados en trabajos adicionales. No caigamos en la trampa de usar un solo oficio para todos los trabajos.

Ahora la cuestión es: ¿cuántos trabajadores tenemos de cada oficio? Por esta vez, vamos a ser buenos de verdad. No vamos a decir que tenemos 17

trabajadores del oficio A en el primer turno, pero solamente 12 en el segundo, pero el sábado un trabajador B, si se le compensa con un 27,945 por 100, podría hacer el trabajo de... No, tendremos compasión. Vamos a coger el caso más simple posible. En nuestra fábrica sólo tenemos un A, un B, un C y un D, y no son intercambiables en absoluto. Ni B puede hacer el trabajo de A, ni A puede hacer el trabajo de B.

¿Cuánto tiempo está disponible en la fábrica cada uno de estos recursos? Vamos a coger otra vez el caso más simple. Vamos a suponer que cada uno de los trabajadores está disponible 5 días por semana, 8 horas por día y 60 minutos por hora. Esto equivale a 2.400 minutos por semana. ¿Se han dado cuenta de que no hay absentismo? Ni siquiera van al cuarto de baño.

¿Qué más nos falta? Los gastos operativos. Vamos a suponer que los gastos operativos totales de la fábrica son de 6.000 dólares e incluyen los salarios de estos trabajadores, sus beneficios sociales, los salarios de los encargados. Los vendedores de la empresa, la dirección, el dinero que pagamos por la energía y el que pagamos a los bancos por los intereses. Todo ello está incluido en los 6.000 dólares. Pero, ¿qué es lo que no está incluido?

Lo que no está incluido es el dinero que pagamos a nuestros proveedores por las materias primas y las piezas compradas. Este dinero no es gasto operativo, es inventario. Si queremos vender algo tendremos que comprar material. ¿Cuánto tendremos que pagar? Esto depende de la cantidad que compremos. Ya está dado el precio por unidad de cada material. Pero recordemos que este dinero es además de los 6.000 dólares.

Ya está dado todo. Todo es preciso. No hay excusas. Así que vamos a repetir la pregunta original, aunque no exactamente, ya que todo está planteado por semana. Vamos a reconstruirla de la forma siguiente: ¿cuál es el máximo beneficio neto (mínima pérdida) por semana que puede alcanzar esta empresa? Tenemos todos los datos, están disponibles y son exactos. ¿Tenemos la información? ¿Podemos contestar a la pregunta de la dirección?

Les recomiendo especialmente que ahora se tomen el tiempo necesario para intentar resolver esta adivinanza por ustedes mismos, antes de seguir leyendo. Arrojará una luz completamente nueva sobre lo que llamamos intuitivamente información, que difiere bastante del uso normal que se le da a esta palabra.

13

Demostración de la diferencia entre el mundo del coste y el mundo del valor

Durante los dos últimos años he tenido ocasión de presentar el caso anterior a más de 10.000 directivos. Resulta asombroso que sólo uno de cada cien haya tenido éxito resolviéndolo correctamente. Y esto no es lo más interesante, es mucho más importante la forma de tratar la cuestión que suele tener la mayoría de los directivos. Casi todos ellos son muy sistemáticos. Empiezan con la pregunta y, recordando la definición de beneficio neto (ingresos netos menos gastos operativos), inmediatamente se embarcan en el cálculo del eslabón que falta, los ingresos netos.

Sigamos sus pasos:

Los ingresos netos se consiguen con la venta de los productos. Empecemos con el producto P. ¿Cuánto se puede vender por semana? Cien unidades. Los clientes están dispuestos a pagar 90 dólares por cada una de ellas. Pero si multiplicamos estos dos números obtendremos las ventas, no los ingresos netos. Para calcular los ingresos netos le tenemos que restar al precio de venta la cantidad que tenemos que pagar a los proveedores. En el caso del producto P, es 45 dólares. Así, el ingreso neto generado por el producto P es:

$$\text{P: } 100 \text{ unidades} \times (90\$ - 45\$) = 4.500\$$$

Ahora, hagamos lo mismo con Q. La cantidad que se puede vender es de 50 unidades por semana. Con cada unidad obtenemos 100 dólares, pero cada unidad requiere un pago a los proveedores de 40 dólares. El ingreso neto conseguido con el producto Q es:

$$\text{Q: } 50 \text{ unidades} \times (100\$ - 40\$) = 3.000\$$$

El ingreso neto total de la empresa es la gama de los ingresos netos de los productos individuales; resulta ser 7.500 dólares. Pero esto no es el benefi-

cio neto. Para llegar al beneficio neto, tenemos que restar los 6.000 dólares de los gastos operativos (ya nos hemos ocupado del dinero pagado a los proveedores). El beneficio neto por semana será:

$$BN = 7.500\$ - 6.000\$ = 1.500\$$$

Muy sencillo. Es asombroso cómo esta respuesta es, por mayoría, la más común. La gente no obedeció a su intuición, sino a su educación. ¿Cuál debería ser siempre el primer paso cuando nos aproximamos a cualquier sistema? Ya lo hemos acordado antes: *identificar las limitaciones del sistema.* Antes de hacer eso, cualquier cálculo que se haga será sólo un ejercicio sin contenido.

En el cálculo anterior hemos supuesto que el mercado es la única limitación. ¿Por qué? ¿Puede que haya una limitación interna? En este caso, desde luego que la hay, y puede que usted ya la haya identificado. A pesar de todo, vamos a seguir los pasos sistemáticamente (sistemáticamente no significa correctamente). En este caso, vamos a usar los números, los datos, para identificar las limitaciones. En la vida real, tenemos que ser conscientes de que los datos suelen ser pura basura.

Todo experto en MRP sabe que, si se tiene que comprobar el tiempo de proceso de una pieza, es mejor preguntar al encargado que al ingeniero. Probablemente el encargado nos engañará en un 30 por 100, pero, por lo menos, sabemos con qué tendencia nos está engañando. El ingeniero puede equivocarse en un 200 por 100, y ni siquiera sabremos cuál es la tendencia. Cualquier sistema de información que valga para algo tendrá que identificar los muy pocos datos de los que se ha derivado la información. Estos datos deben ser verificados cuidadosamente. Si no es así, tendríamos que comprobar la validez de todos los datos, lo que es una imposibilidad, como han podido comprobar miles de empresas. Hemos invertido tanto tiempo y esfuerzo durante los últimos veinte años para llegar a darnos cuenta de esto, que sería una pena desperdiciar el conocimiento adquirido.

Pero, en este experimento hemos establecido que los datos son absolutamente exactos, así que podemos continuar. Lo que tenemos que hacer es encontrar si hay alguna limitación física interna. Puede que haya otro tipo de limitación, limitaciones políticas, pero no se pueden encontrar a partir de los datos. Para encontrarlas directamente necesitamos el método científico de identificación de problemas, la técnica efecto-causa-efecto. Las limitaciones políticas se encuentran fuera del campo de los sistemas de información, y por eso todo sistema de información debe partir de la atrevida base de que no hay limitaciones políticas. Para identificar las limitaciones internas de los recursos, simplemente calculamos la carga que

el programa asigna a cada uno, y la comparamos con la disponibilidad de cada recurso. Para el recurso A la carga del producto P es de 100 piezas por semana multiplicadas por 15 minutos por pieza, ó 1.500 minutos. El producto Q supone una carga adicional para el recurso A de 50 piezas por 10 minutos, lo que convierte el total de carga en 2.000 minutos por semana. La disponibilidad del recurso A es de 2.400 minutos por semana. Aquí no tenemos ningún problema.

Recurso B: El producto P supone una carga semanal de 100 unidades por 15 minutos por unidad, o sea, 1.500 minutos. El producto Q supone una carga de 50 unidades por 30 minutos. Sí, solamente 30 minutos, aunque se procesen dos trabajos distintos. Ya hemos establecido que el tiempo de cambio era cero. La carga total en este caso es de 3.000 minutos, lo que excede con mucho la disponibilidad. Ciertamente, tenemos una limitación en un recurso. Si hacemos el mismo cálculo para los recursos C y D veremos que la carga es de sólo 1.750 minutos por semana para cada uno. Absolutamente ningún problema. B es el único recurso limitado y destaca enormemente, igual que en la vida real cuando existe un verdadero cuello de botella.

Ahora nos enfrentamos a una decisión. Es obvio que no podemos satisfacer todo el potencial del mercado, no tenemos suficiente capacidad en el recurso B. Así que, tenemos que elegir qué productos vamos a ofrecer al mercado, y cuánto vamos a ofrecer de cada uno. La mayoría de los directivos que han alcanzado este punto del experimento (no cayeron en la primera trampa), continuaron de la forma siguiente: no podemos satisfacer todo el mercado, así que vamos a ofrecer al mercado el producto más rentable que tenemos, la «estrella». Si nos queda alguna capacidad residual, ofreceremos el «perro». Tiene sentido.

Bien, ¿cuál es el producto más rentable? Vamos a examinarlo desde más de un punto de vista. Veamos, primero, el precio de venta: P se vende a 90 dólares por unidad, y Q se vende a 100 dólares por unidad. Si ésta fuera la única consideración, ¿qué producto nos gustaría vender? Ciertamente Q.

Ahora, vamos a examinarlo desde el punto de vista del material: P requiere que paguemos a nuestros proveedores 45 dólares por unidad; Q sólo requiere 40 dólares por unidad. Así que, si ésa fuera nuestra única consideración, ¿qué producto preferiríamos vender? De nuevo, la misma respuesta, Q.

También podemos mirar los ingresos netos, que son el precio de venta menos el precio de la materia prima. Nos lleva al mismo sitio. El ingreso neto de P es 45 dólares, mientras que el de Q es 60 dólares. Pero éstas no son las únicas consideraciones. Normalmente miramos la cantidad de trabajo que se necesita para fabricar un producto para el mercado.

Si calculamos la cantidad de trabajo para el producto P, llegamos al número:

15 + 15 + 10 + 5 + 15 = 60 minutos de trabajo.

En el producto Q, los mismos cálculos nos llevan a:

15 + 10 + 5 + 15 + 5 = 50 minutos de trabajo.

Desde el punto de vista del trabajo, ¿qué producto preferiríamos vender? Una vez más, el mismo producto, Q. Es muy importante que nos demos cuenta de que, como las tres consideraciones nos han llevado, en nuestro caso, a la misma conclusión, cualquier sistema de costes que pueda existir nos daría la misma respuesta fuera cual fuera el factor de gastos generales que utilizáramos: indudablemente, Q es un producto más rentable que P.

De acuerdo. Utilizando esto como guía, vamos a calcular el beneficio neto. El primer producto que ofrecemos es Q. Podemos vender 50 unidades de Q a la semana. Cada una de estas 50 unidades demanda 30 minutos de nuestra limitación (el recurso B), lo que absorbe 1.500 minutos de su tiempo disponible cada semana. Esto sólo deja 900 minutos para que se usen en la producción de P. ¿Cuántos P podemos producir en 900 minutos? Cada P requiere 15 minutos del recurso B, así que sólo podremos fabricar y ofrecer al mercado 60 unidades, por semana, del producto P. Sí, el mercado quiere 100 unidades por semana, pero no podemos hacer nada. No tenemos suficiente capacidad.

La mejor mezcla que podemos ofrecer al mercado es de 50 Q y 60 P por semana. El producto Q nos reportará 50 unidades × 60 dólares = 3.000 dólares de ingresos netos, y el producto P nos reportará 60 unidades × 45 dólares = 2.700 dólares de ingresos netos. Los ingresos netos totales serán 5.700 dólares por semana, menos los gastos operativos, que son 6.000 dólares... ¡Vaya! Vamos a perder 300 dólares por semana. Bueno, ¿qué le vamos a hacer? Los japoneses han entrado en nuestros mercados y...

Intente imaginar lo que puede pasar a un directivo que promete a su corporación unas ganancias de 1.500 dólares por semana, y que acaba dando unas pérdidas semanales de 300 dólares. ¿Cuánto durará esta situación antes de que se vea obligado a visitar la agencia de colocación más cercana? Podemos ignorar nuestras limitaciones, pero ellas nunca nos ignorarán.

Pero, un momento, este último cálculo no está en línea con el mundo del valor. No es suficiente con utilizar la terminología de las limitaciones. Tenemos que librarnos de los bloqueos mentales que ha implantado el mundo del coste. Como sin duda habrán notado, este último cálculo usa la

terminología equivocada del *beneficio del producto*. En el mundo del valor no existe el beneficio del producto, sólo existe el beneficio de la empresa.

Vamos a intentarlo otra vez. ¿Cómo debemos calcular el beneficio neto? Ya hemos descrito el proceso que debe utilizarse en cualquier cuestión de dirección. *Decidir cómo explotar la limitación*. ¿Qué hemos intentado explotar? El esfuerzo de la mano de obra. ¿Por qué? ¿Porque no tenemos suficiente A o C o D? No. ¿Entonces, por qué hemos hecho el cálculo que se ha descrito arriba? Porque no tenemos bastante del recurso B.

¿Qué significa explotar la limitación? No significa «tenerla trabajando todo el tiempo». Recordemos que la meta de la empresa no es hacer trabajar a los empleados, es hacer más dinero ahora y en el futuro. Lo que queremos es sacar el máximo dinero de las cosas que nos limitan, de las limitaciones. Cuando ofrecemos al mercado el producto P, el mercado paga 45 dólares por el esfuerzo de la empresa. Recordemos que 45 dólares de los 90 dólares del precio de venta se pagan por el esfuerzo de los proveedores. Pero, ¿cuántos minutos de la limitación tenemos que invertir para conseguir los 45 dólares de ingreso neto? B tiene que invertir 15 minutos. Así que, cuando ofrecemos al mercado el producto P, obtenemos 3 dólares de ingreso neto (*45$/15:00*) por cada minuto de nuestra limitación.

Cuando ofrecemos Q al mercado, la empresa obtiene 60 dólares de ingreso neto, pero tenemos que invertir 30 minutos de la limitación. Así, cuando se ofrece Q al mercado, sólo recibimos 2 dólares (*60$/30:00*) por cada minuto de nuestra limitación. Vean que estos 2 y 3 dólares no tienen nada que ver con el coste, son contribuciones a los ingresos netos. Con estos números frente a nosotros, y estando convencidos de la necesidad de explotar la limitación, ¿qué preferimos vender ahora? Exactamente el opuesto a la respuesta de todos los sistemas de coste del mundo.

¿Quién tiene razón? ¿Todos los sistemas de costes, o nuestra intuición de sentido común? Sólo hay un juez posibe, el resultado. Así que, vamos a calcular cuál será el resultado si seguimos nuestra intuición. Primero ofrezcamos P al mercado. ¿Cuántos P podemos vender por semana? 100 unidades. ¿Cuántos minutos de la limitación se necesitan? Mil quinientos minutos. Ahora, sólo quedan 900 minutos para Q. Cada Q necesita 30 minutos de la limitación. Por tanto, sólo se ofrecerán al mercado 30 Q. Ahora, la mezcla es de 100 P y 10 Q por semana.

Pero, un momento. ¿De verdad entendemos el significado de esta última afirmación? Enséñenme un directivo que sea capaz de levantarse y decir: si tenemos una «estrella» y un «perro», vamos a ignorar a la «estrella» y vamos a apoyar al «perro», todo lo que podamos. Si nos queda alguna capacidad residual, entonces, graciosamente, también ofreceremos la «estrella». ¿Creen ustedes que este directivo tendría muchas oportunida-

des de promoción? Gracias a Dios que aquí sólo estamos tratando con un caso supuesto, así que, sigamos.

El producto P generará unos ingresos netos de 100 × 45 = 4.500 dólares semanales, mientras que Q añadirá otros 30 × 60 = 1.800 dólares de ingresos netos semanales. Ahora, los ingresos netos totales de la empresa serán 6.300 dólares. Si restamos los 6.000 dólares de gastos operativos semanales, el beneficio neto de esa misma empresa es, ahora, 300 dólares más semanales. Más o menos, ¿qué importa?

¿A quién le afecta lo que acabamos de hacer? Sin duda, a los accionistas, a la alta dirección y al departamento de finanzas. ¿Y a quién más? ¿A la gente de producción? En absoluto. A nuestro equipo de ventas. Piénsenlo. Actualmente, ¿qué producto tendría la comisión más alta? El producto Q, por ser el más «rentable». Si la comisión de Q es más alta que la de P, ¿qué producto preferirá promocionar nuestro equipo de ventas? Aunque nosotros sepamos que es mejor vender P, si ellos han vendido Q ya es demasiado tarde. Tendremos que entregar lo que ellos han vendido.

En realidad, a producción no le importa tener que fabricar más de esto o más de aquello. El trabajador B está trabajando a toda velocidad en cualquier caso. Lo que nos indica el último cálculo es que, en el intento de ganar más dinero, necesitamos iniciar un cambio drástico en nuestros planes de comisiones de ventas. Esto es el significado de dependencia: puede que estemos tratando con, por ejemplo, producción, pero las conclusiones pueden afectar principalmente a cualquier otro departamento, en este caso al personal de ventas.

Un pequeño ejemplo de la diferencia entre el mundo del coste y el pensamiento en línea con la realidad del mundo del valor. Pero, ahora, volvamos atrás y hagamos lo que pretendíamos hacer originalmente, evaluar las ramificaciones de nuestro problema de información.

14

Aclarando la confusión entre datos e información. Algunas definiciones fundamentales

¿Cuál era la información que necesitábamos en realidad? La cantidad de beneficio neto que íbamos a tener. Antes de conseguir la respuesta, tuvimos que tomar una decisión: ¿qué mezcla de productos debemos ofrecer al mercado? Esta decisión, a su vez, se basó en la identificación de las limitaciones de la empresa. Este último paso es el único que depende de lo que solíamos llamar datos. Cosas como tiempo de proceso de una unidad, disponibilidad de recursos y previsión de ventas. Empieza a emerger una cadena, una cadena en la que cada eslabón se puede considerar tanto dato como información, dependiendo del lado desde el que lo miremos.

Para aclarar esta última afirmación, hagámonos, una vez más, la pregunta clave de nuestra discusión: *¿qué son datos y qué es información?* ¿Qué fue lo que dijimos al principio de nuestra discusión? El contenido de un almacén es un dato, pero, para la persona que tiene que responder al pedido urgente de un cliente, es información. Ya teníamos la impresión de que la misma serie de caracteres podía ser tanto un dato como una información. Creo que ahora nuestro entendimiento es un poco más profundo. La relación *datos-información* es aún más intrigante de lo que parecía al principio.

Vamos a examinarlo más cuidadosamente. «El recurso B es una limitación». ¿Esto es un dato o una información? Para el director de producción, seguro que es información. Responde a su pregunta principal: «¿a qué recurso tenemos que prestar más atención?» Pero, al mismo tiempo, para el director de ventas es sólo un dato. La pregunta del director de ventas es: «¿a que producto debemos apoyar más en el mercado?» Está claro que, para el director de ventas, la mezcla de productos deseada, «primero P y después Q», es información. ¿Se puede derivar sin saber que B es una

limitación? No. «B es una limitación» es, por tanto, un eslabón que, para un director, es información y, al mismo tiempo, para otro director es un dato necesario.

Pero ése no es el único eslabón de nuestra cadena. Por ejemplo, la pregunta de la alta dirección era *¿cuánto beneficio neto vamos a tener?* La respuesta a esta pregunta, *300 dólares*, es información, mientras que la afirmación *es mejor vender P que vender Q* no es información, es un dato.

Parece que la información no es el dato que hace falta para contestar a la pregunta, sino que es la propia respuesta. Los datos parecen ser los trozos que hacen falta para contestar a la pregunta. ¿Qué hay de nuestra definición anterior: cualquier conjunto de caracteres que describa algo sobre nuestra realidad? Parece que vamos a tener que hacer otra distinción, entre «dato» y «dato requerido». Un momento, esto es importante. Un error en el nivel de las definiciones básicas puede llevar la discusión al caos. Deberíamos volver a examinar nuestras definiciones en situaciones menos complicadas, en situaciones de la vida diaria, donde nuestra intuición es más fuerte. Debemos comprobar si las difiniciones formales de datos e información que se han propuesto coinciden con nuestra percepción intuitiva de estas palabras.

Tomemos un caso en el que preguntamos a nuestra secretaria algo como: ¿cuál es la mejor forma de ir a Atlanta hoy? Yo esperaría una respuesta como: deberías coger este vuelo. A una respuesta así le llamaríamos, sin duda, información. Por supuesto que si este vuelo en particular no va a Atlanta, seguiría siendo información. Simplemente sería información errónea. Por otra parte, si en vez de darnos una contestación directa, nos da todo el horario de vuelos, no estaríamos tan satisfechos. Hemos recibido datos en vez de información. Para la secretaria, el mismo horario de vuelos es información. Su pregunta es: ¿cuáles son todos los vuelos posibles para ir a Atlanta? Incluso, en este caso tan simple, vemos que lo que se considera dato y lo que se considera información depende de la pregunta que se haya hecho: lo que se considera información en un nivel puede ser sólo un dato en otro nivel.

¿Qué palabras usaríamos si se nos entregara un horario de vuelos caducado? No dejan de ser datos. Ni siquiera los podemos llamar datos erróneos. Recordemos que un dato es cualquier serie de caracteres que describa algo sobre nuestra realidad. Parece que sería muy útil añadir el término «datos requeridos». Sí, ahora las definiciones anteriores parecen estar mucho más en línea con nuestro entendimiento intuitivo.

Creo que ahora estamos en una situación mejor para darnos cuenta de la validez de lo que dijimos nada más empezar nuestra discusión: «lo que es dato y lo que es información depende de quién lo mire». ¿Necesitábamos emplear todo este tiempo en analizar la nueva filosofía, sólo para

conseguir un entendimiento un poco más profundo? Creo que la respuesta es ciertamente sí. Definir la información como: la respuesta a lo que se ha preguntado significa que la información sólo se puede deducir como resultado de la utilización de un proceso de decisión. Los datos requeridos son lo que entra en el proceso de decisión, la información es lo que sale, y el mismo proceso de decisión debe estar integrado en un sistema de información. No, ciertamente no hemos perdido el tiempo tratando de formalizar el nuevo proceso de decisión.

Hay una palabra clave escondida en lo que acabamos de decir que no nos debe pasar inadvertida. Hemos llegado a la conclusión de que la información se dispone de forma jerárquica, que en cada nivel la información se *deduce* de los datos. La palabra *deducir* es un término clave. Nos revela que, para poder derivar la información, necesitamos algo más que los datos, necesitamos el proceso de decisión.

¿Qué fue lo que nos enseñó claramente el caso anterior? Disponíamos de todos los datos que nos hacían falta. A pesar de ello, fuimos incapaces de deducir la información deseada: el beneficio neto. Aún peor, fuimos desviados hacia una respuesta errónea.

Se deben cumplir dos condiciones para adquirir información. Ciertamente, los datos son una de las condiciones, pero, es igual de importante el proceso de decisión en sí mismo. Sin un proceso de decisión adecuado no hay forma de deducir de los datos la información necesaria. En el pasado estábamos inmersos en el mundo el coste, del proceso de decisión era totalmente inadecuado, y por eso éramos incapaces de conseguir la información deseada.

Nuestra frustración nos ha llevado a intensificar nuestros esfuerzos, pero se han orientado en la dirección equivocada. En vez de buscar el proceso de decisión adecuado, nos hemos limitado a gastar nuestro esfuerzo en reunir más datos y, después, cuando esto no resultó útil, datos más precisos. Los procedimientos de decisión detallados, y, por tanto, el proceso genérico de decisión, deben ser parte integral de un sistema de información.

Los *cinco pasos de enfoque* parecen darnos el proceso de decisión genérico necesario para subir por la escalera de la información: del dato básico al siguiente nivel, el de identificación de las limitaciones del sistema, y, luego, al nivel más alto de deducción de respuestas tácticas, que, por fin, nos lleva a la información financiera del resultado.

Puedo oír sus comentarios: un pequeño ejemplo y, sin dudarlo un momento, esta persona tan extraña salta a conclusiones globales. Pero, un momento, estas conclusiones no se han derivado del ejemplo, proceden directamente de los *cinco pasos de enfoque* de la intuición del sentido común. El ejemplo era solamente una ilustración del tema. En cualquier caso, no importa, tendremos muchos más ejemplos que podremos discutir.

Lo que debemos tener presente es que los cinco pasos no son suficientes por sí mismos. Para poder subir por la escalera de la información tenemos que desarrollar los procedimientos detallados que se derivan de ellos. Por ejemplo, la forma en que utilizamos el segundo paso, el concepto de *explotar*, no fue precisamente trivial. El procedimiento para determinar la mezcla apropiada de productos, de acuerdo con los «dólares de ingreso neto por unidad de limitación» es muy sencillo, pero no es fácil de encontrar. Además, los dos sabemos que este procedimiento no se puede generalizar. Todavía no se ha terminado la búsqueda de la mezcla de productos más apropiada.

Antes de seguir explorando las ramificaciones de los *cinco pasos de enfoque* en otras cuestiones tácticas, antes de examinar más el alcance de los procesos de decisión resultantes, puede que debamos dedicar algo de tiempo al tema de los datos. Llevamos tanto tiempo intentando desesperadamente conseguir la información necesaria a base de ampliar nuestros esfuerzos de recolección de datos, que puede que nos hayamos sobrepasado en esa dirección. Puede que estemos exagerando en nuestra propia percepción de cuántos datos se necesitan de verdad y de cuánto esfuerzo hay que dedicar, de hecho, a aumentar su precisión.

Repasemos lo que hemos hecho en el supuesto anterior. Necesitábamos datos, pero no todos los datos. Pregúntese si necesitábamos saber el coste por hora de cada uno de los recursos. ¿Para qué se emplea hoy en día tanto tiempo y esfuerzo en intentar determinarlo? Para controlar nuestros gastos necesitamos saber cuánto dinero pagamos en cada categoría, o cuánto dinero pagamos a los trabajadores por salarios y beneficios, pero ¿el coste por hora? Este es un eslabón intermedio que se decidió que hacía falta en el proceso de decisión del *coste*. Este paso resulta totalmente superfluo en el nuevo proceso de decisión. ¿Cuántos más anacronismos como éste existen? Tendremos que estar muy alerta para ir eliminándolos.

Repasemos una vez más estas conclusiones tan importantes.

Respire hondo y sumérjase en esta larga cadena de conexiones lógicas. ¿Qué fue lo que dijimos? Como la información tiene una construcción jerárquica, y como el propio proceso de decisión es el que nos permite subir de un nivel de información al siguiente, esto implica que cualquier cambio en el proceso de decisión puede dejar obsoleto a todo un nivel de información. Algo que se consideraba como un eslabón dato/información, algo que se tenía que deducir de los datos para permitir la derivación a un nivel superior de información, puede ser completamente innecesario cuando se cambia el modo de deducción.

Ya habíamos empezado a sospechar que esto, de hecho, era así cuando examinamos la necesidad del coste por hora, pero, puede que merezca la pena buscar si hay más ejemplos así, o si esto era sólo un caso aislado.

Una de las clases de datos más buscada es el *coste del producto*. Los directores financieros e informáticos a veces se desesperan intentando resolver bien el tema. ¿Para qué necesitamos este dato? ¿Para determinar el precio de venta? Esta no puede ser la razón, porque el precio del producto no lo decidimos nosotros, lo determinan nuestros mercados. Para encontrar el precio que tenemos que poner al producto tenemos que mirar hacia fuera. Si miramos dentro, en nuestra propia fábrica, no llegaremos a ninguna parte.

¿Para qué necesitamos el coste del producto? Para poder tomar decisiones como determinar qué producto debemos favorecer en el mercdo y qué producto no debemos favorecer. Sí, ésta es la única razón, pero reconozcamos que el caso anterior nos ha enseñado claramente que no hay forma de tomar esta decisión utilizando la noción de coste del producto. Todos los sistemas de coste habrían indicado que había que favorecer a Q en el mercado, nuestros resultados nos enseñaron justo lo contrario. Como ya indicamos antes, el coste del producto es un concepto que debe eliminarse junto con su creador, el proceso de decisión que se basa en el mundo del coste.

¿Qué hay de la precisión de los datos? ¿De verdad que nos incumbe conocer cada dato con absoluta precisión? Pregúntese lo siguiente: de todos los tiempos de proceso que aparecían en nuestro supuesto, ¿cuál era el que tenía que ser exacto? ¿No nos hizo falta mucha exactitud para identificar la limitación? ¿A quién le importa si el recurso D va a estar ocupado 1.000 minutos por semana ó 2.000 minutos? Sólo necesitábamos ser precisos hasta el nivel en el que pudiéramos determinar si un recurso era o no una limitación.

Necesitábamos mucha más exactitud en los tiempos de proceso de B. Los utilizamos como dato para un nivel superior de información: para determinar la mezcla de productos que deseábamos. Pero, ni siquiera aquí hace falta exagerar: tampoco necesitamos una exactitud muy grande. Habríamos llegado a la misma conclusión, aunque el tiempo de proceso hubiera sido de 17 minutos en vez de 15. Sólo cuando estamos tratando con el nivel más alto de información, «¿cuál será el beneficio neto?», y sólo entonces, necesitamos que esos tiempos de proceso sean exactos. Su exactitud determinará la exactitud de la información que se derive de ellos. Los demás tiempos de proceso podrían haber tenido grandes inexactitudes sin afectar para nada al resultado final.

Esto es algo a lo que nos tendremos que acostumbrar. En el mundo del coste teníamos la impresión de si aumentábamos la exactitud de cualquier dato requerido, aumentaríamos también la exactitud del resultado final. La lucha por conseguir una mayor exactitud se convirtió en una forma de vida. Este ya no es el caso. En el mundo del valor, la mayoría de los datos

tienen un límite, y aumentar la exactitud más allá de este límite no tiene ninguna influencia en el resultado final. Es frecuente que una mayor exactitud en los datos no se traduzca en una mejor información.

Los directivos necesitan datos para poder tomar decisiones, para poder derivar la información necesaria. Un cambio en el proceso de decisión no sólo implica un cambio en el resultado final, también supone un cambio en la naturaleza de los datos requeridos y en su nivel necesario de exactitud. Lo que hemos visto en el capítulo anterior era un ejemplo de este cambio.

Este capítulo ha resultado más bien denso. Quizá sería útil un sumario intermedio o, si no un sumario, al menos, una lista con las definiciones de los términos que tan generosamente hemos esparcido por estas últimas páginas.

Información: la respuesta a lo que se ha preguntado.

Información errónea: una respuesta equivocada a lo que se ha preguntado.

Dato: cualquier serie de caracteres que describa algo sobre nuestra realidad.

Dato requerido: el dato requerido por el proceso de decisión para derivar la información.

Dato erróneo: una serie de caracteres que no describe la realidad (podría ser un residuo de un procedimiento de decisión erróneo).

Dato inválido: un dato que no se necesita para deducir la información específica requerida.

Dejemos, por ahora, el tema de los datos y continuemos explorando las ramificaciones del proceso de decisión en otros aspectos de nuestra empresa, utilizando, para ello, el mismo caso.

15

Demostración del impacto del nuevo proceso de decisión sobre algunos aspectos tácticos

Podría ser una buena idea conseguir primero una mejor apreciación de las ramificaciones del cambio a un nuevo proceso de decisión. Esto podría arrojar alguna luz nueva sobre la calidad de la información obtenible, así como darnos alguna idea preliminar de los datos que se requieren y de su nivel deseable de exactitud. Así que, vamos a usar la misma adivinanza para ilustrar el cambio en un aspecto de nuestra empresa completamente distinto.

Supongamos que yo soy el encargado del centro de trabajo A. Usted es el jefe de fábrica. Yo voy y le pregunto: ¿cuántas unidades de cada pieza debo procesar por semana? Como la meta de la fábrica es ganar más dinero, probablemente su respuesta será: produce 100 de P y 30 de Q. Tiene sentido, pero me acaba de poner usted en una situación muy desagradable. Probablemente tendré que recordarle, con todos los respetos debidos, que no entiendo su respuesta. Al ser un encargado del área que fabrica piezas, mi terminología no es de productos, sino de *códigos de pieza*. Al darse cuenta de ello, cambia usted a la terminología de la persona que debe ejecutar sus instrucciones.

Me pedirá que produzca 100 unidades por semana de la pieza de la izquierda, ninguna de la del medio (al ser el encargado del centro A, no tengo nada que ver con la pieza del medio), y sólo 30 unidades de la pieza de la derecha. Lo que ha hecho usted, simplemente, es seguir, intuitivamente, el tercer paso: *subordinar todo lo demás a la decisión anterior.*

Pero, ahora, observe lo que me va a pasar a mí. Recuerde que todavía soy el encargado del centro de trabajo A. ¿Cuánto tiempo tengo que invertir para producir 100 unidades de la pieza de la izquierda? Cada una requiere 15 minutos, por tanto, necesito 1.500 minutos. La pieza del medio no requiere tiempo alguno, y la de la derecha 10 minutos multiplicados por 30.

¿Ayudo a la empresa si produzco más? Ciertamente no. Las piezas de sobra no se convertirán en ingreso neto. Lo que determina el ingreso neto es la capacidad de B. La producción adicional sólo contribuirá a engordar el inventario. Pero, ¿cuánto tiempo necesito yo para producir completamente sus requerimientos? 1.500 + 300 = 1.800 minutos. ¿Cuánto tiempo tengo disponible en mi centro de trabajo? Dos mil cuatrocientos minutos. Así que, si hago exactamente lo que usted me ha pedido que haga, ¿qué pasará con mi eficiencia? Mi eficiencia caerá. ¿Qué pasará con mi cabeza? Probablemente, también caerá.

Si hago exactamente lo que usted me ha pedido, seré castigado. Así que, ¿qué cree usted que voy a hacer en realidad? Hablaré con mis amigos, el que hace el programa o el jefe de almacén. Si hace falta, robaré el material. Voy a hacer mis propios cálculos. Y recuerde, el muerto no se va a encontrar en mi departamento, el inventario se acumulará más adelante, en piezas acabadas o en producto acabado. En tierra de nadie.

¿Qué otra opción me deja? Hacer lo correcto y recibir un castigo, o jugar con los números y ser un heroe. ¿Qué cree usted que soy, un santo? O lo que es peor, ¿un mártir?

El concepto del mundo del valor que dicta el tercer paso de enfoque, «subordinar todo lo demás a la decisión anterior», implica un cambio drástico en nuestras medidas de rendimiento local. ¿Durante cuánto tiempo podemos seguir recompensando la estupidez, y castigando las actuaciones correctas? ¿Cuánto dinero, tiempo y esfuerzo se gasta hoy en día en recopilar datos para medir el rendimiento local, sólo para que el resultado final distorsione el comportamiento de la gente que estamos midiendo?

Si hoy decimos: fabrica sólo 100 de éstas, luego, 30 de aquéllas y, luego, párate. ¿Qué creen que se registrará en la mente de los trabajadores? ¿Cuándo fue la última vez que la dirección les dijo que dejaran de trabajar? ¡Cinco minutos antes de que empezara una reducción de plantilla! Cada uno de los trabajadores bajará su ritmo, instintivamente, para probar que se les necesita. El tema que acabamos de tocar es un tema muy sensible, el cambio de las medidas del rendimiento local se aproxima mucho al cambio de la *ética de trabajo*.

¿Cuál es la ética de trabajo actual? Creo que la podemos resumir en la siguiente frase:

«Si un trabajador no tiene nada que hacer, ¡hay que buscarle algo que hacer!»

El concepto de subordinación está en contradicción directa con el comportamiento actual. No nos engañemos pensando que el cambio de las medidas locales será tarea fácil. Es verdad que podemos, sin mucha complicación, diseñar los procedimientos para conseguir la información vital de que es lo que debe hacer cada uno. De hecho, nos encontraremos con que tenemos que recopilar bastante menos datos, y los requerimientos de exactitud se reducen. Pero, ésta no es la cuestión. ¿De verdad piensan que podemos cambiar la cultura a base de cambiar las medidas de rendimiento local? No es tan fácil. Hemos dicho antes: «dime cómo me mides y te diré cómo me comporto». Pero, ahora, tenemos que recordar la otra mitad de la historia: *cambia mis medidas por unas nuevas que no entiendo bien, y ni siquiera yo sé cómo me comportaré*.

No podemos cambiar una cultura simplemente cambiando sus medidas de rendimiento.

La transformación desde el mundo del coste hasta el mundo del valor, el nuevo proceso de decisión, nos permite construir, por primera vez, un sistema de información relativamente simple, pero su utilización depende por completo de la capacidad que tenga la empresa de transformar su cultura.

Llegados a este punto, tomemos nota de que tenemos que desarrollar las nuevas medidas de rendimiento local, ciertamente, una de las piezas clave del sistema de información que necesitamos. Pero continuemos, por ahora, explorando las ramificaciones del cambio, para que podamos llegar a ver el cuadro completo.

Supongamos que ya hemos desarrollado todos los instrumentos necesarios para la nueva ética de trabajo. Las empresas que ya han hecho el cambio han descubierto que, en contra de lo que se cree normalmente, los trabajadores buscan trabajo. Algunas empresas han llegado al extremo de poner estanterías con periódicos en la fábrica, pero esto no ha supuesto ninguna ayuda. El tiempo ocioso, resultado inevitable de forzar a los recursos no limitados a que se abstengan de sobreproducir, se debe llenar con algo significativo. La solución natural podría ser utilizar el tiempo libre para mejorar los procesos locales. Desgraciadamente, nuestros trabajadores suelen ser mucho más «inocentes» que nosotros, no se engañan a sí mismos tan fácilmente. Darles *programas de mejora* sin sentido, que no lleven a una verdadera mejora de los resultados de la empresa, no puede ser una solución a largo plazo.

Debemos diseñar el procedimiento que nos proporcione, de forma continuada, la capacidad de identificar cuáles son las actividades locales de mejora que más se necesitan. Esta necesidad no es nueva, pero, en el entorno del mundo del valor esta información se convierte en algo casi obligatorio. Así que, vamos a explorar el tema de la mejora de procesos, pero, para hacer el tema más animado, vamos a enfocarlo de la siguiente manera:

Vamos a suponer que yo, ahora, soy un ingeniero de procesos. Antes era encargado, pero he estado estudiando, he conseguido el título, y ahora me han promocionado a ingeniero de procesos. Ya no soy un encargado,

me han promocionado. A usted no le han promocionado. Usted todavía es jefe de fábrica.

Supongamos que yo entro en su oficina mañana, usted practica la política de puertas abiertas, y le digo lo siguiente: estamos produciendo en la fábrica una pieza en grandes cantidades, es una de las más importantes. Tenemos que invertir 20 minutos de tiempo de proceso en cada unidad. Sí, 20 minutos. He tenido una idea: autoríceme 2.000 dólares para un dispositivo... bueno, puede que un poco más, pero, seguro que con 3.000 dólares será suficiente. Una vez que tengamos el dispositivo le demostraré que podemos hacer la pieza no en 20 minutos, sino en... 21 minutos.

¿Cuál sería su reacción? ¿Vuelvo a ser encargado? ¡Si tengo suerte! Supongamos que usted está dotado de una paciencia infinita y me pregunta si mejorará la calidad con este cambio. «No», le respondo. ¿Se requerirá menos material? «No, tampoco hay cambios en ese área». ¿Y bien?

¿Cuál fue mi único error? Me tomé demasiado en serio las instrucciones de la dirección. Me dijeron que *elevara las limitaciones del sistema*, incluso me dijeron que la limitación era el recurso B. «Haz algo al respecto, cada gotita cuenta. Todo minuto que se pueda ahorrar en la limitación es importante.» Cuando pregunté lo que tenía que hacer con todos los demás proyectos en los que estaba envuelto, los suprimieron directamente. «Todos giran alrededor de mejoras en los recursos no limitados, olvídate de ellos.» Eso es lo que dijeron. Me gusta mi trabajo, disfruto trabajando para la empresa. Así que me fui a casa y me exprimí el cerebro para averiguar cómo se podía elevar la limitación. Encontré la forma de hacerlo. Ahora me han despedido.

Como pueden ver, la reacción inicial ante mi sugerencia fue una reacción del mundo del coste. En el mundo del coste no hay forma de que resulte beneficioso aumentar el tiempo total de producción de un elemento. ¿Y en el mundo del valor? Mi idea era la siguiente:

Con el nuevo dispositivo se puede reducir el tiempo que tarda la limitación en procesar la pieza del medio. Este dispositivo nos permite pasar parte de la carga de trabajo del recurso B al recurso C. Ahora, en vez de 15, se tardarán catorce minutos en producir la pieza del medio en la limitación. Podemos pasar un minuto al recurso C, pero como C es más lento que B, ahora la pieza del medio tardará en C 7 minutos en vez de 5. El resultado final es que el tiempo total de proceso para la pieza del medio pasa de 20 a 21 minutos. Exactamente lo que yo dije.

Si se hubieran tomado en serio mi sugerencia, ¿cuántas semanas se habrá tardado en recuperar la inversión de los 3.000 dólares del dispositivo? Por supuesto, en el mundo del coste nunca se habría recuperado. En el mundo de JIT o de TQM se pueden justificar las inversiones etiquetándolas como mejoras. Pero tenemos que contestar en el mundo real, un mundo

donde la meta de la empresa no es reducir los costes o mejorar, sino ganar más dinero.

Gracias a este dispositivo podemos liberar un minuto de la limitación en cada unidad. Estamos produciendo 130 unidades semanales de la pieza del medio, lo que significa que tenemos en nuestras manos 130 minutos de limitación adicionales por semana. ¿A cuánto podemos vender cada minuto de limitación? Ya lo habíamos calculado antes. Pero no, la respuesta no es 3 dólares por minuto, porque ya hemos satisfecho todo el mercado potencial de P, aunque todavía tenemos demanda de Q sin satisfacer. Todavía podemos vender minutos adicionales de limitación a 2 dólares por minuto. Esto significa que la compra del dispositivo contribuirá a los ingresos netos en $130 \times 2 = 260$ dólares semanales. Como los gastos operativos permanecen iguales, este cambio en los ingresos netos se deslizará agradablemente hacia el resultado final. Necesitaremos un poco menos de 12 semanas para cubrir los 3.000 dólares. Una inversión bastante buena, ¿no es así?

Bajo la mentalidad del mundo del coste ningún ingeniero se atrevería a sugerir semejante «peora», subir el tiempo de proceso de 20 a 21 minutos. Un encargado que tuviera a su cargo ambos centros de trabajo, B y C, podría llegar a hacer algo así, pero se ocuparía muy bien de disimularlo. Si se analiza exactamente, el encargado será castigado por aumentar sus varianzas. Ha invertido en las piezas más de lo que permiten los estándares.

¿Qué datos necesitan nuestros ingenieros de procesos? Necesitan saber qué recursos están limitados, y por cuánto podemos vender cada minuto adicional que consigan sacar de ellos. Eso es todo. En vez de eso, ¿qué datos les damos? Oceános de lo que llamamos información de costes. ¿Cuál es el resultado? Otro ingeniero llamará a su puerta con la siguiente sugerencia: «Podemos mejorar la segunda operación de la pieza de la izquierda. Invirtiendo 3.000 dólares en unos utillajes, podremos reducir el tiempo de proceso de 10 a 5 minutos.» Y, probablemente, le daremos el premio al ingeniero del año. ¿Qué hemos conseguido a cambio?

No, la respuesta no es cero. El impacto negativo también existe, y no me refiero solamente a la inversión de 3.000 dólares que hemos desperdiciado. Ahora tenemos algo mucho peor; ahora tenemos un ingeniero muy orgulloso de sí mismo, orgulloso por un motivo equivocado. Hemos desviado a una persona muy especial, y muy cara, para que enfoque su atención, ahora y en el futuro, hacia puntos equivocados.

¿Cuántas empresas que tengan «programas de reducción de costes» conoce usted? Si todas estas reducciones de costes se reflejaran en los resultados, estas empresas serían extremadamente rentables. ¿Dónde desaparecen las reducciones de costes? Ahora tenemos la respuesta; para em-

pezar, no eran reducciones de costes. ¿Cómo podemor reducir los costes si no recortamos el gasto operativo? Son sólo juegos de números. Cuando los gastos operativos son básicamente fijos, la única forma de aumentar el beneficio neto es aumentando los ingresos netos.

¿Qué es información? ¿Qué datos necesitamos para deducirlo? ¿Cómo tienen que ser de exactos estos datos? Estas preguntas empiezan a tener un significado. Puede que, después de todo, no hayamos perdido el tiempo discutiendo las *nuevas filosofías globales de dirección.*

16

Demostración de que la inercia es una causa de las limitaciones de procedimiento

Ya que estamos inmersos en el caso supuesto, y nos lo sabemos de memoria, continuemos utilizándolo. Debemos recordar que todavía no hemos explorado el poder del quinto paso.

Supongamos que nuestro director de marketing ha visto lo que hemos hecho y, dándose cuenta de la validez del cuarto paso *(elevar las limitaciones del sistema)*, se sube al carro.

Dice, con razón, que hay otra limitación en la empresa además del recurso B. Una parte del potencial de mercado también es una limitación.

No sería correcto afirmar que el *potencial* de mercado es una limitación. Esto podría hacer que nuestros vendedores obtuvieran más pedidos de Q, lo que no mejoraría nuestros resultados. ¿Qué es una limitación? Si se alivia la restricción, ganaremos más dinero. Si cuando aliviamos la restricción no hay impacto en los resultados, tenemos una indicación muy clara de que no estamos tratando con una limitación.

Aquí la limitación es una demanda insuficiente del mercado para el producto P. Si conseguimos más potencial de mercado en el producto P, podremos ganar más dinero. Nuestro director de marketing nos llama la atención sobre el hecho de que nuestra empresa no vende nada en Japón, por ejemplo. Sugiere que debe ir a Japón para descubrir si hay mercado para nuestros productos.

Regresa dos semanas más tarde y nos presenta, lleno de orgullo, sus descubrimientos. Vamos a aceptar sus descubrimientos como parte del supuesto, no sólo para el caso actual, sino, también, para casos futuros. Lo que nos dice es lo siguiente: «En Japón hay un mercado maravilloso para nuestros productos. Están deseando comprar P, y están deseando comprar Q. ¡Les encantan! Además, allí el mercado es enorme. De hecho, es tan grande como nuestro mercado nacional; podemos vender hasta 50 Q por semana y/o hasta 100 P por semana. Pero, hay un pequeño problema...»

Cuando quiera que hablamos con la gente de marketing, de alguna manera, al final siempre hay un pequeño problema. Sí, su suposición es correcta. Si queremos vender en Japón tendremos que descontar el 20 por 100 de nuestro precio de venta. Marketing nos garantiza que esta reducción de precios no influirá en nuestros precios nacionales. Lo han comprobado a fondo. Podemos dar precios reducidos en Japón sin afectar a nuestros precios de aquí. Quieren una ligera modificación en nuestro producto que nos permitirá diferenciar los precios, una modificación que, en realidad, no requiere ningún esfuerzo adicional.

¿He oído la palabra *dumping*? ¿Qué es lo que no se permite en el *dumping*? ¿Vender por debajo del coste? Si no existe el coste del producto, ¿cómo podemos vender por debajo? Y, además, ¿dónde prefiere usted comprarse una cámara japonesa, en Manhattan o en Tokio? En Manhattan, ¿verdad? ¿Por qué? Porque es más barata. ¿Por qué es más barata en Manhattan? Ah, como todo el mundo sabe, los precios del transporte son negativos.

No se nos ocurriría ir a Japón a vender Q por un 20 por 100 menos. Si no vendemos todos nuestros Q aquí a 100 dólares, ¿por qué vamos a ir hasta Japón para venderlos a 80 dólares? Pero, ¿deberíamos ir o no a Japón para vender P a 72 dólares?

Una pregunta muy difícil en el mundo del coste. Una pregunta muy fácil en el mundo del valor. Estamos en una situación en la que los gastos operativos son fijos, y en la que todavía no podemos elevar el recurso limitado. ¿Cuál es la única forma de aumentar el beneficio neto? Mejorando la explotación de la limitación. Ahora mismo, lo mínimo que conseguimos por el tiempo de la limitación son 2 dólares por minuto. Si en Japón podemos conseguir más de 2 dólares por minuto, deberíamos vender allí. Sin duda, aumentará nuestro resultado. Si no es así, el único motivo para ir a Japón sería ir de visita turística.

El precio de venta en Japón es de 72 dólares. De ahí tenemos que descontar la materia prima, que supone 45 dólares. Por alguna razón, nuestros proveedores nos cobran lo mismo, sin importarles dónde vendemos el producto final. No hay patriotismo. Por tanto, el ingreso neto por unidad es 27 dólares. Todavía tenemos que invertir en él 15 minutos de la limitación, lo que hace que la relación dólares de ingreso neto por minuto de limitación es menor que 2. En estas circunstancias, no deberíamos ir a Japón.

¿Qué necesita nuestro personal de ventas suponiendo, por supuesto, que la segmentación del mercado es perfecta? El valor en dólares de la materia prima de cada producto, el número de minutos de limitación de cada producto; y un número adicional, el nivel del umbral: el mínimo que obtenemos hoy en día por cada minuto de limitación. ¿Cómo deberían

ellos determinar la información requerida? ¿A qué precio vendemos? En un mercado perfectamente segmentado deberían cobrar tanto como soporte el mercado. Si el precio de mercado nos sitúa por debajo del umbral, no deberían aceptar el pedido. ¿A que es bastante fácil? En vez de eso, ¿qué «información» les damos? Más vale no pensarlo.

Pero, ahora, vamos a ponernos más serios. Esta adivinanza no nos representa una fábrica de la vida real. ¿Cuál es la situación que se describe? Los clientes están llamando a nuestra puerta con los talonarios en la mano, queriendo comprar nuestros productos, y no podemos aceptar sus cheques sólo por una estúpida máquina. ¿Qué haríamos normalmente en un caso así? Por supuesto, ¡vamos a comprar otra máquina!

Venga, vamos a hacerlo. Vamos a comprar otra máquina B. Pero un momento, sólo tenemos una persona en toda la fábrica que sepa operar esta máquina. Y, además, esta persona ya tiene ocupado el 100 por 100 de su tiempo. Si lo que se pretende es ganar más dinero, y no sólo decorar la fábrica, tendremos que contratar a otro trabajador. Vamos a suponer que ya lo hemos encontrado. Barato. Una auténtica ganga. Sólo 400 dólares por semana, beneficios incluidos. Ahora nos enfretamos a un caso más general. Con frecuencia, la decisión de invertir afecta tanto a los gastos operativos como a los ingresos netos. Hemos subido los gastos operativos de la empresa a 6.400 dólares semanales.

¿Cuál es ahora la pregunta? Hemos comprado una máquina nueva, no nos la han regalado. Digamos que el precio de compra de la máquina es de 100.000 dólares. Para simplificar los cálculos, vamos a suponer que no hay intereses. Por supuesto, la pregunta es: ¿en cuántas semanas recuperaremos el precio de la nueva máquina?

De nuevo, insistiré en que dejen de leer en este punto y hagan los cálculos por sí mismos. No es para que se pongan a prueba, es para que aprovechen esta oportunidad para revelarse a sí mismos lo que de verdad guía sus acciones.

Muy pocas de las personas que se han enfrentado a esta pregunta han sido capaces de responderla correctamente. La «atracción» del mundo del coste se subestima demasiado. Vamos a seguir sus pasos para verificar el enfoque típico de un directivo occidental.

Hemos comprado otra máquina B, así que B ya no es la limitación. Ahora podemos proporcionar los 100 P y los 50 Q. Ya hemos comprobado que tenemos suficiente capacidad en todos los demás recursos. El mercado se ha convertido en nuestra limitación.

P nos proporciona 100 × 45 = 4.500 dólares de ingreso neto, y Q nos proporciona otros 50 × 60 = 3.000 dólares de ingreso neto. Los ingresos netos totales ascienden a 7.500 dólares. De aquí tenemos que restar los gastos operativos. Recordemos que ahora son 6.400 dólares porque hemos

contratado a otra persona. El beneficio neto es de 1.100 dólares por semana. Pero no podemos utilizar todo este dinero para recuperar la inversión que hemos hecho en la máquina. Ya teníamos beneficios anteriormente. Sólo podemos utilizar el incremento de beneficio neto que ha resultado de la compra de la nueva máquina. El incremento de beneficio neto es de 1.100 − 300 = 800 dólares por semana. Como el precio de la máquina era de 100.000 dólares, recuperaremos la inversión en 125 semanas exactamente. ¿Correcto? No.

Vamos a recordar el quinto paso de enfoque, *si en los pasos anteriores hemos roto una limitación...* Es exactamente donde nosotros estamos, ¿no? Acabamos de romper una limitación... *Hay que volver al paso número uno, pero sin permitir que la inercia cause una limitación de sistema.* Esta advertencia se hace para situaciones como la nuestra. ¿Le hemos hecho caso? Verán, en el mundo del coste casi todo es importante, por eso, si cambiamos una o dos cosas, no cambia mucho el cuadro total. Pero, en el mundo del valor no es éste el caso. Aquí hay muy pocas cosas que sean de verdad importantes. Si cambiamos una cosa importante tenemos que volver a evaluar toda la situación.

¿Por qué decidimos no ir a Japón? Porque nos habría reportado menos de 2 dólares por minuto de limitación. ¿Qué limitación? ¡Ya la hemos roto!

La compra de la nueva máquina ha generado exceso de capacidad en todos los recursos. Ahora, la única limitación es el mercado. Vayamos inmediatamente a Japón para vender nuestro exceso de capacidad. Toda la diferencia entre el precio de venta y el precio de materia prima, el ingreso neto adicional, será beneficio neto. Los gastos operativos son fijos, aunque ahora están a un nivel más alto.

Si vamos a Japón, ¿qué se convertirá en la limitación interna? ¿Cuál es el siguiente recurso en nivel de ocupación? El recurso A, por supuesto. Así que, vamos a rehacer los cálculos. La venta de 100 P en el mercado nacional nos proporcionará 4.500 dólares de ingreso neto, pero requerirá 1.500 minutos del recurso A. Cincuenta unidades de Q, en el mercado nacional, nos proporcionarán 3.000 dólares adicionales de ingreso neto, y requerirán 500 minutos adicionales del recurso A. Esto nos deja con un excedente de 400 minutos «ociosos» en el recurso A... vamos a «largarlos» en Japón.

¿Cuántos P podemos producir en 400 minutos del recurso A? Unas 26 unidades. El ingreso neto por unidad ya no es 45 dólares, estamos vendiendo en Japón. Es sólo 72 − 45 = 27 dólares. Así que, la venta de P en Japón añadirá otros 700 dólares de ingreso neto. ¿Deberíamos molestarnos por una cantidad tan pequeña? Veamos. Ahora los ingresos netos totales son 8.200 dólares, menos 6.400 dólares de gastos operativos, menos el beneficio neto anterior de 300 dólares, nos lleva a un cambio de 1.500 dólares en el beneficio neto semanal. Casi el doble de lo anterior. Por culpa de la *inercia*, nos habíamos dejado la mitad del dinero encima de la mesa.

¿Entendemos ahora el significado de la inercia? Desgraciadamente la respuesta todavía es que no. En el mundo del coste existe el concepto de coste del producto. Mientras no cambiemos el diseño del producto, o los salarios de los trabajadores, no cambiará el coste del producto. Pero, en el mundo del valor no existen ni el coste del producto ni el «beneficio del producto». No tenemos que evaluar el impacto de un producto, sino el impacto de una decisión. Esta evaluación se debe hacer a través de su impacto en las limitaciones del sistema. Por eso, el primer paso siempre es la identificación de las limitaciones del sistema. Si las limitaciones han cambiado, hay que volver a examinar las decisiones.

¿Por qué decidimos vender P en Japón? Porque teníamos la impresión de que P era el producto más rentable. Esta impresión es una extrapolación del mundo del coste. *Producto rentable* es la terminología del pasado. Sí, habíamos decidido que la venta de P sería más rentable para la empresa. ¿Por qué llegamos a esta conclusión? Porque B era una limitación. Pero ya no lo es.

Intentemos vender Q en Japón. Utilicemos los 400 minutos residuales de A para la producción de Q. Podemos fabricar 40 unidades, ya que Q sólo necesita 10 minutos de limitación por unidad. El ingreso neto de cada Q que se venda en Japón es 80 dólares (el precio de venta en Japón es un 20 por 100 más bajo), menos 40 dólares de la materia prima, lo que es igual a 40 dólares. Por tanto, la venta de Q en Japón añadirá otros 1.600 dólares a nuestro beneficio neto, en vez de los 700 dólares que ganábamos vendiendo P. Un aumento de 900 dólares en el resultado. Ya que vamos a ir a Japón, ¿por qué no vender el producto correcto? *¡Inercia!*

¿Entendemos ahora lo que significa la inercia? ¿Nos damos perfecta cuenta de sus devastadores riesgos? La respuesta todavía es *no*. ¿Por qué decidimos que primero teníamos que exprimir el mercado nacional, y, sólo después, examiar la posibilidad de exportar? Porque al principio la exportación no estaba en el cuadro. *¡Inercia!*

Vuelva al primer paso, no permita que haya inercia. Vuelva al paso 1 y mire al sistema como si nunca lo hubiera visto. Es un sistema nuevo. Recordemos una vez más que, en el mundo del coste el cambio de una o dos cosas no influye mucho. En el mundo del valor, el cambio de una limitación lo cambia todo.

Lo que tenemos que hacer es volver a examinar qué producto contribuye más, a través de la limitación. Ahora, la limitación es A, no B. Ya se ha descrito antes la forma de calcularlo, dólares de ingreso neto divididos por minutos de limitación. Hagan ustedes mismos los cálculos, les espera una sorpresa.

SEGUNDA PARTE

La arquitectura de un sistema de información

17

Escrutando la estructura inherente de un sistema de información. Primer intento

Como ya hemos visto, la información es la respuesta a la pregunta que se ha hecho. ¿Qué tipo de información estamos buscando? Sólo tenemos que escuchar las quejas de todo el mundo para poder contestar: nos estamos ahogando en mares de datos y, a pesar de todo, nos falta información. Limítese a escuchar cuidadosamente los ejemplos que siguen a este desesperado lamento...

¿Deberíamos aceptar el pedido de este cliente? ¿Deberíamos aprobar la asignación de fondos para invertir en nuevas máquinas? ¿A qué precio debemos ofertar? ¿Qué más podemos hacer para reducir la exagerada duración del diseño de nuevos productos? ¿Debemos fabricar esta pieza, o debemos seguir comprándola? ¿Cómo podemos evaluar objetivamente el rendimiento de un área local? ¿Qué proveedor debemos elegir? Etc.

Es difícil no darse cuenta de que la información que más se necesita se refiere a preguntas a las que se suponía que contestaba la contabilidad de costes. De hecho, no resulta sorprendente, ya que el proceso de decisión del mundo del coste es totalmente inapropiado para nuestra realidad del mundo del valor. Ninguna de esas preguntas se puede contestar adecuadamente con nuestros procedimientos tradicionales de toma de decisiones. Los directivos sólo han podido depender de su intuición. Faltaba el puente, el proceso de decisión, entre los datos y la información necesaria. Así, el océano de datos que les ofrecíamos no les ayudaba a justificar sus respuestas intuitivas. De hecho, lo que conseguía era borrarlas.

Vamos a digerir lo que hemos dicho, aun a riesgo de parecer algo redundantes. ¿No es esto una discusión? Parece como si la continuación de nuestra charla nos fuera a aclarar las bases para definir la naturaleza de lo que estamos buscando, como si nos encontráramos muy cerca de hallar el núcleo. La naturaleza de los *sistemas de información* que necesitamos es muy distinta de la de los *sistemas de datos* de que disponemos.

Ya nos hemos dado cuenta de que la información se dispone en una estructura jerárquica en la que la información de los niveles superiores se puede deducir de los niveles inferiores a través de un proceso de decisión. No tenemos la información necesaria porque no hemos usado los procesos de decisión adecuados. Esto implica que el sistema de información que estamos buscando debería estar orientado, principalmente, hacia los niveles superiores de la información. Es una conclusión interesante, entremos un poco más o fondo.

Cuando la pregunta se podía contestar sin necesidad de un proceso de decisión, simplemente localizando el dato necesario, ya lo hemos hecho. La mayoría de nuestros esfuerzos anteriores se han orientado en ese sentido. No es sorprendente que llamemos a nuestros sistemas actuales *sistemas de datos*. No es que no intentáramos contestar a las preguntas del nivel más alto; la lista de preguntas que hemos descrito antes siempre ha estado ante nuestros ojos, los sistemas de *coste* son un buen ejemplo de nuestros intentos anteriores. Pero, durante nuestra discusión, hemos descubierto que estos intentos no nos llevaban a ninguna parte porque se basaban en procesos de decisión equivocados.

Dada esta situación, deberíamos reservar el nombre de *sistemas de información* para los sistemas que son capaces de responder a preguntas que implican el uso de un proceso de decisión. Los sistemas que se orientan hacia la contestación de preguntas más directas se deberían llamar *sistemas de datos*. Esta última decisión implica que el sistema de información no se debería ocupar consultando datos disponibles, sino que debería suponer que existe un sistema de datos del que puede extraer los que sean necesarios. Es una conclusión atrevida, pero no va contra la intuición.

La potencia de un sistema de información se debería juzgar principalmente por la amplitud de las cuestiones a las que de verdad puede responder. Cuanta más amplitud, más potente será el sistema.

¿A dónde nos lleva esto? Parece que el primer paso debe ser desarrollar un proceso de decisión que sea adecuado para cada una de las preguntas que nos hacíamos antes, y para muchas más que no hemos mencionado.

Esto, como sin duda sospechan, nos asegura que nuestra pequeña discusión tendrá que continuar algún tiempo más, sin duda, más allá del alcance de este libro. Hasta un pequeño caso supuesto nos ha hecho ver que no nos vamos a aburrir, pero, ¿podemos permitirnos dedicarle tanto tiempo? Si seguimos este camino, ¿cuándo vamos a hablar de la estructura de un *sistema de información*?

La tentación de construir los distintos procesos de decisión es muy grande, especialmente al darnos cuenta de que estos nuevos procedimientos ni son muy complicados ni requieren una sofisticación exorbitante. Por el contrario, da un poco de vergüenza lo simples que son. No es de

extrañar, cuando pasamos de la noción de que casi todo es importante a la noción de que son pocas cosas las que de verdad influyen, las cosas se simplifican maravillosamente. Ya hemos notado que tenemos tan metida la mentalidad del mundo del coste que nos enmascara la realidad. Va a ser ciertamete divertido derribar tantos bloques mentales desarrollando esos procedimientos que hemos necesitado desesperadamente durante tanto tiempo.

Pero, no olvidemos el propósito principal de esta discusión. Era hacer un esquema de la estructura y composición de un *sistema de información* completo y fiable. ¿Podemos hacer esto antes de diseñar todos los procesos de decisión necesarios?

Por suerte, ya hemos establecido las directrices de las que se pueden derivar los procesos de decisión que hacen falta para contestar a las preguntas de dirección. Son, simplemente, los *cinco pasos de enfoque*. Puede que podamos encontrar la estructura del *sistema de información* requerido examinando solamente una o dos de esas preguntas. Puede que haya un patrón general. Si éste es el caso, podemos construir la estructura ahora e ir rellenándola después, cuando vayamos tratando con cada pregunta específica. Así, podemos conseguir grandes beneficios mucho más pronto. Además, en cualquier caso tendremos que ir introduciendo esos conceptos gradualmente en nuestras organizaciones. La transición completa desde el mundo del coste hasta el mundo del valor no se va a realizar en un solo día.

Bien, ¿a qué estamos esperando? Vamos a empezar ya.

Supongamos que usted es el director de compras. Uno de sus problemas principales es determinar el nivel de inventario de cada una de las materias primas y piezas compradas. Usted sabe que es un trabajo ingrato. Si no mantiene un inventario suficiente, los de operaciones se quedarán sin piezas y todo el mundo le echará la culpa. Si intenta acumular grandes reservas, el director financiero pedirá a su jefe que intente meterle algo de sentido común. ¿Cuánto hay que tener? Y lo que no es menos importante, ¿cómo puede usted demostrar que está manteniendo el inventario en el punto adecuado?

Examinemos su problema usando algunos números concretos. Supongamos que el precio de compra de un elemento es de 100 dólares por unidad, y que el plazo de entrega del proveedor es de seis meses. Otro elemento cuesta 10.000 dólares por unidad, y el plazo de entrega es de sólo dos meses. ¿Cuántas semanas de inventario hay que tener para cada elemento?

No se precipite con la respuesta. La percepción del mundo del coste le inducirá a creer que la respuesta es obvia, que el nivel de inventario de la primera pieza debería ser mayor que el de la segunda. En los cinco últimos

años hemos aprendido que hay otros factores mucho más dominantes que el precio y el plazo de entrega del proveedor. Vamos a considerar uno de estos factores adicionales.

Si estamos debatiendo cuál debe ser el nivel del inventario, quiere decir que estos elementos no se consumen de forma esporádica, sino continuadamente. Si fuera de otra forma, habríamos comprado justo lo necesario para cumplir pedidos específicos y no estaríamos hablando de mantener inventarios de materia prima. Así que, vamos a suponer que el primer elemento se entrega con una frecuencia bisemanal, mientras que la frecuencia del segundo es de sólo una vez al mes. ¿Cuál sería ahora su respuesta? ¿Qué elemento debe tener el nivel más alto de inventario?

Deberíamos tener unas dos semanas de inventario del primer elemento, dos semanas de consumo nuestro. Pero, para el segundo, dos semanas no es suficiente. Deberíamos tener un mes, ¿no? ¿Dónde entran en el cuadro el precio y el plazo de entrega del proveedor? ¿Son en absoluto importantes? Sí, pero en mucha menor medida de lo que habíamos supuesto. Veamos, cuando decimos un mes de consumo, ¿qué es lo que de verdad queremos decir? ¿Un mes de consumo medio? Con nuestra experiencia, no debemos caer en esa trampa. Ya hemos aprendido, dolorosamente, que Murphy es la persona más activa de nuestra empresa. Debemos mantener un mes de *consumo paranoico*. ¿Hasta qué punto debemos ser paranoicos? Por supuesto, nuestra paranoia está en función de las fluctuaciones internas, que, a su vez, están fuertemente influidas por la fluctuación en la demanda de nuestros clientes. ¿Cómo podemos cuantificar la paranoia o, alternativamente, cuantificar a Murphy? Eso es otro asunto. A pesar de ello, es obvio que nuestro grado de paranoia se verá influido por el precio unitario del elemento. Cuanto más alto sea, menos nos podremos permitir el ser paranoicos. ¿No es asombroso? Lo que en el mundo del coste considerábamos como el parámetro más importante resulta ser sólo un factor de corrección en el mundo del valor.

¿Qué ocurre con el plazo de entrega del proveedor? ¿Tiene alguna importancia? Una vez más sí, pero sólo como factor de corrección secundario. Cuando lanzamos un pedido al proveedor, algo que debemos hacer con una antelación, al menos igual que el tiempo de entrega del vendedor, no debemos considerar nuestro consumo paranoico actual, sino nuestro consumo paranoico futuro. El material que pedimos hoy sólo será inventario cuando llegue a nuestra empresa, lo que esperamos que suceda en el plazo de entrega del proveedor a partir del lanzamiento del pedido.

Por ejemplo, en el primer elemento tendremos que evaluar cuál será el consumo dentro de seis meses. Por supuesto, cuanto más largo sea el plazo de entrega del proveedor, el momento futuro que se evalúe será más remoto y, por tanto, la incertidumbre será mayor. Pero, no olvidemos que

el precio del elemento y el plazo de entrega del proveedor tienen mucha menos importancia que la frecuencia de entrega. Sólo impactan en el resultado a través del nivel de interferencia que causan nuestras fluctuaciones internas. Cuanto más estable sea nuestra empresa, menos importancia tienen.

Ya que hablamos de niveles de interferencia, debemos reconocer que nuestra empresa no es la única que está introduciendo incertidumbre en el juego. La falta de fiabilidad del proveedor también es importante. Vamos a usar el mismo caso, pero suponiendo que el primer proveedor, el que entrega cada dos semanas, es muy poco fiable. Poco fiable hasta el punto de que hay bastantes probabilidades de que falle una o dos entregas, o de que un envío sea defectuoso en su totalidad. El otro proveedor es extremadamente fiable, tanto en las fechas de entrega como en la calidad del material. ¿Qué contestamos ahora? ¿Se acuerdan de la pregunta? Era: ¿qué nivel de inventario debemos mantener para cada uno de estos dos elementos?

Podemos ver que, para determinar los niveles de inventario de materia prima, debemos considerar tres factores principales. El primero es la frecuencia de las entregas, el segundo el nivel de interferencia de nuestra propia empresa (la falta de fiabilidad en los niveles de consumo). El tercer factor es la fiabilidad del proveedor (tanto en el cumplimiento de las fechas de entrega como en la calidad de los productos). Esto último suscita otra cuestión importante: si un proveedor ofrece mejores precios, el segundo, mejor frecuencia y, el tercero, mayor fiabilidad, ¿qué proveedor elegiríamos? Es evidente que nos hace falta una evaluación numérica.

¿Qué hemos hecho, aparte de demostrar una vez más la diferencia entre pensar lógicamente en el mundo del valor o jugar con los números en el mundo del coste? ¿Podemos aprender de este último ejemplo algo que nos ayude a descubrir la estructura necesaria de un sistema de información?

18

Introducción de la necesidad de cuantificar la «protección»

Gracias al último ejemplo, podemos aprender bastante sobre la estructura del *sistema de información* que deseamos. No debemos dejar que el pánico nos arrastre. De hecho, no es tan complicado como pudiera parecer. ¿Dónde acabó nuestro análisis de la gestión de compras? La cuestión del nivel adecuado de materia prima, así como la cuestión de la selección de proveedores, dependían fuertemente de un dato necesario: nuestro nivel de consumo paranoico.

Si conociéramos este dato ilusorio podríamos haber usado los demás datos para llegar, sin gran dificultad, a un juicio numérico razonable. ¿Qué podemos hacer para conseguir este dato? La primera pregunta que debemos hacernos está bastante clara: ¿Qué es lo que determina el ritmo de consumo?

A primera vista parece que se puede contestar a esta pregunta sin ningún género de duda, al menos conceptualmente. El ritmo de consumo lo determinan las limitaciones del sistema. ¿Estamos seguros? ¿Sólo las limitaciones del sistema? Dada una serie de limitaciones, ¿existen grados adicionales de libertad en la determinación del ritmo de consumo?

Bien, ya veo lo que me van a decir. Las limitaciones deberían determinar el ritmo de consumo, pero las decisiones inadecuadas sobre cómo considerarlas pueden afectar drásticamente al resultado final. Por una parte, si no usamos adecuadamente las limitaciones, el ritmo de consumo será menor que el deseado. Por otra parte, si las ignoramos y nos concentramos en eficiencias de mano de obra artificiales, los ritmos de consumo de materia prima se inflarán innecesariamente.

Esto parece que nos suena, como si estuviéramos recorriendo un camino ya conocido y bien hollado. Lo que acabamos de expresar es una repetición de los pasos de enfoque. Primero, necesitamos *identificar* las limitaciones del sistema. Después, decidir cómo *explotarlas* y *subordinarlas*.

Tiene sentido, y es normal que lo tenga. La necesidad de *subordinar* ¿sólo entra a través de la necesidad de eliminar la tendencia a conseguir eficiencias altas? No necesariamente. No olvidemos que lo que necesitamos predecir no es el consumo, sino el consumo paranoico. ¿Se puede conseguir? Por supuesto, pero para ello tenemos que desarrollar un mecanismo adicional. Para empezar, ¿por qué estamos paranoicos? Porque nos damos perfecta cuenta de que normalmente las cosas no funcionan bien. Las perturbaciones son parte de la vida real. Así pues, necesitamos desarrollar un mecanismo que prediga numéricamente los niveles de inventario que hacen falta en toda la fábrica, considerando la estructura de limitaciones y el nivel de «ruido» de nuestra fábrica. En efecto, aquí el paso de subordinación es muy importante.

Lo que vemos, al menos en este caso, es que para contestar a una pregunta de dirección, incluso cuando el proceso de decisión está plenamente desarrollado y entendido (lo que sospecho que aún no es el caso), necesitamos datos. No podemos obtener este tipo de datos en el mundo del coste. A un nivel más bajo es información, información que los procedimientos tradicionales de decisión eran incapaces de deducir de los datos disponibles. Vemos que antes de que podamos responder fiablemente a las preguntas sobre compras, nuestro sistema de información debe llevar a cabo dos pasos preliminares.

La primera fase, que tendrá que incluir nuestro sistema de información, es el procedimiento necesario para *identificar* y *explotar* las limitaciones, subordinando a ellas todo lo demás, tanto ahora como en el futuro. Esta fase (o bloque código, si consideramos un sistema de información por ordenador) tiene que estar instalada. Además de esto hay que instituir otra fase, basada en un procedimiento que separe el «ruido» de las transacciones diarias, que ocurren, de hecho, en nuestra empresa. Este procedimiento debe ser capaz de convertir estos datos en niveles adecuados de protección, que se colocarán en los sitios más apropiados para reducir el impacto de Murphy.

Vamos a examiar otra cuestión muy corriente para ver si se repite el mismo modelo. Si nuestros *cinco pasos de enfoque* son tan genéricos como sospechamos, así debe suceder.

Supongamos que actualmente estamos comprando un cierto elemento para el que tenemos la capacidad técnica de producirlo en nuestra empresa. ¿Debemos seguir comprándolo, o debemos empezar a producirlo? La respuesta del mundo del coste es, aparentemente, ridícula. ¿Qué se supone que debemos hacer? ¿Calcular el coste de producir el elemento y compararlo con el precio de compra? Supongamos que el coste de producirlo en la empresa es de 100 dólares, de acuerdo con los cálculos tradicionales, y el coste de compra es de 80 dólares. ¿Qué hacemos?

Ya nos hemos aprendido la lección. El coste del producto sólo es un fantasma matemático, sin ninguna relevancia en la vida real. La forma más fácil de justificar esta afirmación es examinando la situación descrita arriba con los siguientes datos adicionales. Supongamos que el precio de compra de la materia prima necesaria para producir ese elemento es de 40 dólares, y que todo el trabajo interno lo realizan recursos no limitados. ¿Cuál es ahora vuestra respuesta? La compra del elemento impactará negativamente en el resultado final en 80 dólares, mientras que el impacto negativo de producirlo en la empresa será de sólo 40 dólares.

Bien, la necesidad de identificar las limitaciones del sistema es cada vez más clara. ¿Qué hay del segundo paso, el concepto de *explotación*? Puede que la decisión de fabricar la pieza en la empresa sea, de hecho, la aplicación de este concepto, aunque al revés de como lo habíamos aplicado hasta ahora. Tiene sentido. ¿Y el tercer paso, el concepto de *subordinación*? Podríamos decir que, como la decisión ya se ha tomado en el segundo paso, el tercero no tiene ninguna vigencia.

Podríamos decirlo, pero no sería correcto. Deberíamos entender el concepto de subordinación con mucha más profundidad. Recordemos que casi todo lo que hemos aprendido de la aplicación de los cinco pasos de enfoque nos ha venido, hasta ahora, del caso supuesto que hemos estudiado. Uno de los supuestos básicos sobre los que está construido el caso es la ausencia de incertidumbre, de «ruido». Podemos recordar que esto se hizo para desechar, de una vez por todas, las excusas típicas: la noción de que la incertidumbre de los datos es lo que nos impide conseguir una buena información. Aunque, por este motivo, la omisión estaba justificada, por otra parte, nos ha impedido examinar en profundidad el mecanismo de *subordinación*.

Al *subordinar*, la atención se centra en los «eslabones más fuertes» de nuestra cadena. Sí, ya nos dimos cuenta de que en cualquier cadena sólo puede haber un «eslabón más débil» pero, ¿por qué se da este fenómeno? Podemos demostrar con todo rigor que éste será el caso en cualquier sistema que contenga tanto variables dependientes como fluctuaciones estadísticas. Si en su viaje a través de la empresa un material pasa por más de una limitación física alterna, no se producirá el resultado esperado, y el inventario crecerá indefinidamente. (Veáse *Theory of Constraints Journal*, volumen 1, número 5, artículo 1.)

En otras palabras, debemos huir de las limitaciones interactivas como de la peste. Esta última conclusión nos lleva a darnos cuenta de que todos los demás «eslabones» deben tener más capacidad de la que estrictamente se necesita para la carga prevista. Resulta muy lógico y muy bonito, pero, ¿cuál es la razón sencilla, de sentido común, para que esto sea así? Si las matemáticas nos demuestran que ésta es la situación real, debemos acep-

tarlo, pero la pregunta sigue ahí. ¿Por qué se da este fenómeno? Puede que la respuesta resida simplemente en nuestro bien conocido e indeseable amigo, el Sr. Murphy.

Vamos a mirarlo a fondo, teniendo presente lo que hemos aprendido hasta ahora. Supongamos que hemos identificado la limitación de un sistema y ahora queremos explotarla. Cualquier despilfarro en la limitación perjudicará irremediablemente nuestro resultado. Pero, un momento, nos acabamos de acordar de la existencia de Murphy. Si Murphy ataca directamente a la limitación, poco podemos hacer. Lo único que podemos hacer es maldecir nuestra mala suerte y continuar. Pero, ¿y si Murphy ataca a uno de los recursos que alimentan a la limitación? ¿Tampoco podemos remediarlo?

Seguimos queriendo explotar la limitación. Y podemos hacerlo, siempre que hayamos preparado, con antelación, algo de inventario delante de nuestro recurso limitado. Parece una buena idea, el inventario que necesitemos para proteger nuestro resultado no se va a considerar como un pasivo. Mientras Murphy siga existiendo, más vale que lo tengamos.

Pero, ¿es el inventario una protección suficiente? Si Murphy ataca, y seguimos explotando la limitación, consumiremos el inventario que habíamos acumulado delante de ella. ¿Qué pasa entonces? Las visitas de Murphy a nuestras empresas no son infrecuentes. ¿Qué pasa si Murphy vuelve a atacar los recursos no limitados? La limitación se ve afectada.

Es obvio que debemos reconstruir rápidamente el inventario situado delante de la limitación, debemos reconstruirlo antes de que «Murphy ataque otra vez». Pero, para poder hacer esto, los recursos no limitados deben ser capaces de procesar el material más rápidamente de lo que sería estrictamente necesario según el ritmo de consumo de la limitación. Tienen que seguir proporcionando lo que necesita la limitación y, además, tienen que reconstruir el inventario. Esto nos lleva a la inevitable conclusión de que mientras existan las fluctuaciones estadísticas, si queremos explotar la limitación, debemos tener en los demás recursos más capacidad de la que sería estrictamente necesaria según la demanda.

Para aclararlo más, supongamos que uno de los recursos que alimenta a la limitación no tiene capacidad de sobra. ¿Qué nivel de inventario necesitaremos que haya delante de la limitación para garantizar su explotación? Supongamos que Murphy ataca precisamente a este recurso, o a uno de los que le alimentan. La limitación empezará a comerse el inventario, y ahora no hay forma de recuperarlo. El nivel de protección ha bajado irreversiblemente. Murphy ataca otra vez. Vuelve a bajar el nivel de protección. Y, así, sucesivamente... Si aceptamos que Murphy no va a desaparecer, ¿cuánto inventario debemos situar inicialmente delante de la limitación? En efecto, si tenemos otra limitación en la cadena —un recurso sin capacidad sobrante— necesitaremos un inventario infinito para poder explotar, al menos, una limitación.

Como la limitación no tiene capacidad interna de protección, debemos protegerla de Murphy con una combinación de inventario situado delante de ella y de capacidad de protección de los recursos que la alimentan. Estos dos mecanismos de protección se compensan. Una menor capacidad de protección en los recursos que la alimentan requerirá un mayor nivel de inventario delante de la limitación. Si no es así, la limitación se quedará sin piezas de vez en cuando y la empresa perderá ventas. Si la capacidad de protección de uno de los recursos que la alimentan es cero, el inventario requerido delante de la limitación debe ser infinito. Así pues, si una cadena tiene más de un eslabón más débil, su fuerza será considerablemente menor que la de estos eslabones. Una cadena así se verá destrozada por la dura realidad; los sistemas que contienen limitaciones interactivas desaparecen pronto en nuestro mundo salvaje.

Solíamos considerar que cualquier capacidad que no fuera la estrictamente necesaria para producir era un desperdicio. Ahora nos damos cuenta de que cuando examinamos la capacidad disponible no debemos dividirla en dos segmentos, sino en tres. El primero es capacidad de producción, el segmento que necesitamos para cumplir con la demanda. El segundo es capacidad de protección, que se necesita como escudo para defendernos de Murphy. El único que no tiene capacidad de protección es el recurso limitado (recordemos - *explotar*). La capacidad que quede después de considerar las capacidades de producción y de protección es, de hecho, exceso de capacidad.

Ahora, vemos que la producción de un elemento que antes se compraba puede tener efectos negativos, aunque la producción sólo se realice en recursos no limitados. Por ejemplo, si el elemento consume tiempo de un recurso no limitado que no tiene exceso de capacidad, la carga adicional reducirá, por definición, la capacidad de protección de ese recurso. Esto requerirá un incremento de los niveles de inventario en proceso y de producto acabado para proteger las limitaciones. Un aumento no sólo en el inventario de ese elemento, sino en el de todos los demás elementos que pasan por las mismas áreas.

Esta consideración es totalmente nueva, y hasta ahora no la hemos tenido en cuenta. Lo que hicimos en nuestro caso supuesto fue concentrarnos en el impacto sobre las ventas y sobre el gasto operativo. Lo que deberíamos hacer siempre es evaluar el impacto sobre las tres medidas, inventario incluido. El paso de subordinación es esencial para contestar con fiabilidad a las cuestioes sobre comprar o fabricar.

Recordemos que la razón por la que pusimos al *inventario* en el segundo lugar de la escala de importancia era por su impacto indirecto sobre las

ventas. Si queremos obtener una respuesta cuantitativa sobre el impacto de una actuación (aunque sólo sea en beneficio neto) no podemos ignorar al inventario. En muchas cuestiones puede llegar a ser el factor dominante.

Y lo que es más, este asunto de la capacidad de protección provoca inmediatamente la pregunta de cómo distinguir entre la misteriosa capacidad de protección y el exceso de capacidad. Cuando un recurso está parado, ¿cómo podemos saber si eso es correcto o es un despilfarro?

Estos dos últimos temas, tan importantes, vuelven a sacar a la luz la necesidad de que nuestro *sistema de información* no sólo *identifique* y *explote* las limitaciones, sino que, además, derive algún método para determinar el nivel de Murphy partiendo de los hechos que se dan en nuestra empresa.

Exactamente, la misma conclusión que antes. Puede que ahora debamos intentar definir la arquitectura de un sistema de información.

19

Los datos requeridos sólo se pueden conseguir a través de la programación y de la cuantificación de Murphy

Ahora deberíamos concentrarnos en esbozar un diagrama de bloques de la arquitectura global del sistema de información que deseamos.

El contenido del bloque más alto está muy claro. Cualquier sistema de información que merezca ese nombre debe ser capaz de responder al tipo de preguntas *¿qué pasaría si...?* que hemos mencionado en capítulos anteriores. ¿No es esto lo que en realidad queremos? Intenten imaginar el efecto que tendría en nuestra empresa el hecho de que tuviéramos respuestas fiables a esas preguntas. Sobre todo a la vista del hecho de que actualmente estamos derivando respuestas erróneas, casi opuestas, de forma continua.

Pero, ya en el primer examen que hicimos pudimos ver que para conseguir este fin necesitamos otros bloques. Los datos que se requieren para la fase del *¿qué pasaría si...?* no están disponibles. Es suficiente con examinar lo que llevamos hecho para hacerse una buena idea del tipo de datos que faltan. De hecho, podemos empezar a distinguir entre dos tipos de datos ausentes:

El primero salta a la vista: conocimiento de cuáles son las limitaciones de la empresa. Debemos desarrollar el procedimiento para identificar limitaciones actuales y futuras. Es más, debe tener la capacidad de identificar cuáles serán las limitaciones si elegimos una alternativa determinada, una de las alternativas que evaluamos en la fase del *¿qué pasaría si...?*

Como ya hemos mencionado, las limitaciones no son necesariamente físicas, muchas veces son las políticas las que limitan. Un *sistema de información* debería concentrarse en las limitaciones físicas, sin caer en la estupidez de instituir limitaciones políticas que podrían ser devastadoras.

¿Cómo podemos *identificar* las limitaciones del sistema? Sabemos que algunas limitaciones se pueden conocer desde el principio, como el recurso

B de nuestro caso. Hay otras limitaciones que sólo se pueden encontrar una vez terminado el paso de *explotación* sobre las limitaciones ya identificadas: el caso del potencial de mercado para el producto P. Otras limitaciones sólo se revelarán una vez hecha la subordinación. Normalmente éste es el caso de los recursos que no tienen suficiente capacidad de protección.

De lo anterior se deduce, claramente, que no se pueden encontrar de forma simultánea todas las limitaciones, no hay un paso que lo abarque todo. Lo tendremos que hacer gradualmente, quizá, incluso, recorriendo los tres primeros pasos en un proceso reiterativo, en el que se va identificando una nueva limitación en cada bucle. Como el número de limitaciones es muy pequeño, este hecho, en sí mismo, no representa una carga significativa.

Pero empezamos a darnos cuenta de otra cosa. Algunas limitaciones sólo se pueden encontrar después de haber realizado la fase de subordinación, subordinación que se refiere a la limitación que ya se ha identificado y explotado. Si éste es el caso, sólo podemos evitar el proceso de iteración en la realidad simulando sucesos hacia el futuro.

Esta es una conclusión muy importante y algo sorprendente. Lo que acabamos de decir es que, hasta para poder identificar limitaciones *actuales* necesitamos simular acciones *futuras*. Para identificar limitaciones, un sistema de información debe tener la capacidad de simular acciones futuras. Esto quiere decir que la *promoción* debe ser una parte integral y fundamental de nuestro *sistema de información*.

Parece que nos hemos encontrado con otra antigua necesidad: la capacidad de generar sistemáticamente un programa fiable. En efecto, sin duda es así. La *promoción* debe ser uno de los bloques fundamentales de un sistema de información, un prerrequisito del bloque de *¿qué pasaría si...?*

En principio, parece que esta conclusión nos hace llevar una carga adicional. Nos gustaría llegar al punto en el que podamos contestar preguntas de dirección, y ahora nos encontramos con la enorme barrera que supone construir un programa fiable para nuestros recursos. Aunque, pensándolo bien, no debería sorprendernos tanto. Durante mucho tiempo, las preguntas de dirección más preocupantes han sido: «¿qué debemos lanzar al proceso?», «¿cuándo debemos lanzarlo?», «¿en qué cantidades debemos hacer las cosas?» Resumiendo: ¿quién debe hacer qué, cuándo y en qué cantidades? —*programación*.

De hecho, la programación es una lista de respuestas a preguntas de dirección y, por tanto, es información. ¿Necesitamos un proceso de decisión para poder generar esta información? Recordemos que ésta era nuestra prueba crucial para distinguir entre lo que se debería incluir en un sistema de información y lo que se debía dejar para el sistema de datos.

Bien, ¿podemos generar un programa fiable sin un proceso de decisión?

La respuesta es, muy probablemente, que no. Consideremos el hecho de que nadie tiene un sistema fiable de programación; ciertamente no son datos que estén disponibles inmediatamente. Probablemente esto se debe a que los intentos actuales se basan en un proceso de decisión erróneo.

¿Es posible que no se disponga de información para programar, porque no se ha hecho ningún intento serio de obtenerla? Este no es el caso. Basta con darse cuenta de cuánto dinero, tiempo y esfuerzo se ha invertido, y se sigue invirtiendo, en el intento de instalar y operar los sistemas de MRP (Material Resource Planning).

La intención original de la implantación de MRP era, y todavía es, programar. Que yo sepa, no hay ninguna empresa que haya implantado MRP sólo para tener una base de datos. Hoy en día, una gran cantidad de empresas industriales han instalado (algunas, más de una vez) un sistema así. Si tenemos en cuenta no sólo el precio del programa, del sistema de ordenadores y de la carísima implantación, sino, también, el espantoso gasto permanente que supone mantener los datos medianamente actualizados, será difícil que no alcancemos una cifra de muchos miles de millones. Ciertamente, el que no haya un modo de programación fiable y sistemático no se debe a falta de esfuerzo.

Ya hemos dicho bastante. Nos guste o no, si queremos obtener respuestas fiables a las preguntas de dirección, tendremos que resolver el problema de la programación fiable. Sin duda, la *programación* es uno de los bloques básicos de un sistema de información, pero, desgraciadamente, no es el único bloque adicional que necesitamos.

Nuestras exploraciones nos han revelado la enorme importancia del papel que desempeña Murphy en nuestras empresas. Si releemos el capítulo anterior, nos daremos cuenta de que Murphy es precisamente la razón por la que necesitamos inventarlo y capacidad de protección para cumplir con las ventas previstas. ¿Cómo podemos medir a Murphy?

El intento de medir las perturbaciones locales de cada recurso individual no sólo es una tarea gigantesca, es, teóricamente, imposible. El tiempo necesario para recopilar suficientes datos estadísticos en la realidad es mucho mayor que el tiempo medio entre los cambios que se producen. Parece que la única salida es considerar el impacto agregado de todas las «acciones» de Murphy. ¿Qué es lo que necesitamos defender de Murphy? Las limitaciones del sistema. El inventario adicional se acumula delante de la limitación, de forma que ésta pueda proseguir su trabajo sin interrupciones, aunque algo vaya mal en el proceso anterior. (Recordemos que el mercado, casi siempre, es una limitación física en los sistemas que se supone que no tienen limitaciones políticas.) La capacidad de protección de los recursos no limitados debe estar disponible para que se puedan recuperar de los ataques de Murphy. El inventario de protección debe

haberse recuperado a un nivel suficiente antes de que Murphy ataque otra vez.

Ya están claramente localizados los pocos puntos en los que se agrega el impacto de todas las acciones de Murphy: son las acumulaciones de inventario que hay delante de las limitaciones, ésta es la base lógica del procedimiento que llamamos *gestión de buffer*.

Creo que aquí no debemos extendernos mucho en este procedimiento tan interesante. De hecho, mi opinión es que debemos evitar meternos en procedimientos particulares, aunque parezcan muy interesantes e importantes, hasta que establezcamos, firmemente, el diseño conceptual del sistema de información. Si no lo hacemos así corremos el riesgo de no alcanzar nuestro objetivo principal. Cuando tengamos el marco global, y no antes, podremos ampliar y describir detalladamente cada procedimiento a medida que nos vayamos encontrando con la necesidad.

A pesar de esto, no podemos abandonar el tema de la *gestión de buffer* sin aclarar, al menos, su terminología más básica. Primero, porque no es justo mencionar un procedimiento de pasada y dejar a todo el mundo cavilando. Segundo, y más importante, es que si no se define la terminología fundamental, ésta tiene la mala costumbre de surgir en los sitios más inesperados, convirtiéndolo todo en un lío indescifrable. Pero, para esto necesitamos otro capítulo.

20

Introducción del concepto de *buffer* de tiempo

Vamos a volver al tema de los *buffers*. Dijimos que necesitábamos construir *buffers* de inventario delante de las limitaciones. Pero, un momento, puede que nos hayamos precipitado. Lo que de verdad dijimos es que tenemos que proteger nuestra capacidad de explotar la limitación. ¿No es lo mismo? No necesariamente. Puede que la limitación física no sea un recurso, puede que sea el mercado o, siendo más específicos, el pedido de un cliente. ¿Y qué?

Si queremos garantizar la entrega a tiempo, a pesar de los problemas que causa Murphy, parece que la única solución es acumular inventario de producto acabado. ¿No es así? No siempre. Supongamos que los acuerdos que tenemos con nuestros clientes nos exigen entregar no más tarde de una fecha específica; no podemos enviar después de esa fecha, pero, si enviamos antes, nuestros clientes estarán felices. Sí, ya sé que éste no es siempre el caso, pero, ciertamente, es una situación muy común. En los casos en que tenemos la flexibilidad de enviar antes, ¿es el inventario de producto acabado la única forma que tenemos de proteger la puntualidad de las entregas?

Si lo preguntamos de esta forma resulta evidente que podemos proteger la fecha deseada de entrega a base de inventario o a base de tiempo. Podemos empezar a cumplimentar un pedido antes de la fecha dada, por la duración de sus procesos. Empezar antes de lo necesario nos da el tiempo suficiente para reaccionar ante las perturbaciones imprevistas y nos asegura la entrega a tiempo. Si no hay perturbaciones, terminaremos el pedido antes de la fecha prometida, pero el resultado no será un incremento del producto acabado, sino un envío más temprano.

Si volvemos a examinar lo que acabamos de decir, nos queda un cierto mal sabor de boca, parece que sólo estamos haciendo juegos de palabras. Sí, podemos proteger con inventario o, como hemos visto en el último

ejemplo, podemos proteger con tiempo, con un lanzamiento más temprano. Tiempo e inventario no son sinónimos, y, aun así, nos parece que hemos repetido lo mismo dos veces.

¿De dónde viene esta sensación de incomodidad? Puede que debamos tratar el caso en que el cliente no acepta que se le envíen antes sus pedidos; en este caso, es obvio que la única forma de proteger es a base de inventario de producto acabado. ¿Cómo vamos a construir ese inventario? Empezando las operaciones antes de lo estrictamente necesario. Esto significa que, en ambos casos, las acciones que deben ejecutarse serán exactamente iguales: conseguimos la protección empezando antes. ¿Estamos tratando con dos mecanismos de protección, o es sólo uno?

Puede que la mejor forma de averiguarlo sea examinando otro caso donde la limitación sea un recurso físico. Este es el caso en el que nuestra intuición está mejor desarrollada. Aquí está claro que tenemos que garantizar que el inventario de protección está delante de la limitación. ¿Cómo nos aseguramos de que el inventario se irá acumulando? Pensemos en una fábrica en la que muchos productos distintos pasan por la misma limitación. La forma de construir el inventario es la misma de antes, empezando la produccción de cada trabajo antes de lo que sería estrictamente necesario según los tiempos de proceso y de manipulación de materiales.

¿Cómo es que dos entidades distintas, tiempo e inventario, parecen ser la misma cuando se juzgan según las acciones que implican? Creo que esta uniformidad emana del hecho de que estos dos mecanismos de protección deben existir simultáneamente. En realidad no son dos mecanismos de protección, sino uno solo. La dualidad emana de los distintos puntos de vista con los que tendemos a ver las cosas. Podríamos decir que estamos intentando proteger dos cosas distintas. Una es la limitación, la otra es el rendimiento, el *output*, de la limitación: los pedidos de los clientes.

Si usamos la terminología de protección de la limitación, nos centramos en garantizar que la limitación no estará ociosa. La terminología que usamos de forma natural es la de *inventario;* la composición específica de ese inventario no tiene relevancia alguna. En realidad, ¿a qué llamamos *proteger la limitación*, si no es a proteger su rendimiento? Si nos centramos en el rendimiento de la limitación, los pedidos específicos de los clientes, la terminología que usaríamos es la de *tiempo.*

Como ya hemos dicho, proteger la limitación y proteger su rendimiento es básicamente lo mismo, por lo que no es sorprendente que las acciones que se derivan de ello sean iguales: lanzar antes el material. ¿Qué terminología debemos usar de aquí en adelante, inventario o tiempo? Parece que nos encontramos ante una elección, pero ya hemos aprendido, en muchas ocasiones, que las elecciones arbitrarias son sólo la consecuencia de un entendimiento insuficiente. Así pues, vamos a emplear un poco más de

tiempo para aclarar este tema, en vez de elegir al azar, con un 50 por 100 de probabilidades de arrepentirnos de ello más adelante.

¿Qué hemos dicho? Si usamos la terminología de proteger la limitación, tendemos a usar el inventario como mecanismo de protección. La composición específica de este inventario no tiene relevancia alguna. Interesante. ¿Es posible que la composición del inventario no sea relevante? Parece sospechoso. Vamos a estudiarlo con más profundidad:

Proteger la limitación. ¿De dónde ha salido esta frase? En realidad, es una versión abreviada del segundo paso de enfoque: decidir cómo *explotar* las limitaciones del sistema. ¿A que ahora está más claro? Si explotar la limitación significa tenerla trabajando continuamente, la composición del inventario no tiene importancia alguna. Pero, no es éste el caso. Explotar la limitación significa sacarle el máximo (en términos de la meta que se ha predeterminado). Nuestro famoso caso nos ha enseñado que la clave está en el contenido del trabajo con el que queremos activar la limitación. Sólo cuando trabajamos con un único producto se deteriora el significado de *explotar* a «hacerlo trabajar todo el tiempo».

Ya que la composición del trabajo de la limitación es de la máxima importancia, la protección se debe expresar como *tiempo.* Esta determinación (ya no es una elección) también está en línea con nuestra intuición cuando consideramos aplicaciones más amplias que la de producción. Proyectos, ingeniería de diseño, administración, por no mencionar los sectores de servicio, todos pertenecen al marco de nuestra discusión. Todos tratan de tareas que deben cumplir los recursos para conseguir una meta predeterminada. Pero, en esos entornos el inventario es con frecuencia invisible, mientras que el concepto *tiempo* se entiende perfectamente.

Hemos decidido usar *tiempo* como nuestra unidad de protección, y, por lo tanto, cuando nos refiramos al *buffer* nos estaremos refiriendo al *buffer de tiempo.* Por lo tanto, el *buffer* es un intervalo de tiempo: el intervalo de tiempo en el que lanzamos el trabajo antes de lo que sería necesario si asumiéramos que Murphy no existe. Los *buffers* se expresan en horas, días o meses. ¿Qué determina la longitud de un *buffer*? (Fig. 20.1).

El *buffer* de tiempo es nuestra protección contra perturbaciones desconocidas. Lo que se desconoce de ellas no es que vayan o no a ocurrir; su ocurrencia es algo casi garantizado. Lo que no sabemos es cuándo ocurrirán, dónde afectarán, ni cuánto durará la perturbación. Dada la naturaleza aleatoria de las perturbaciones, es obvio que no estamos en situación de estimar el *buffer* de tiempo con precisión. Aunque tuviéramos el tiempo y los recursos necesarios para recopilar todos los datos estadísticos, sólo conseguiríamos una curva de probabilidad. En la figura 20.2 se enseña una curva así, en la que en el 20 por 100 de los casos la perturbación dura cinco minutos, y en el 1 por 100 de los casos dura dos días.

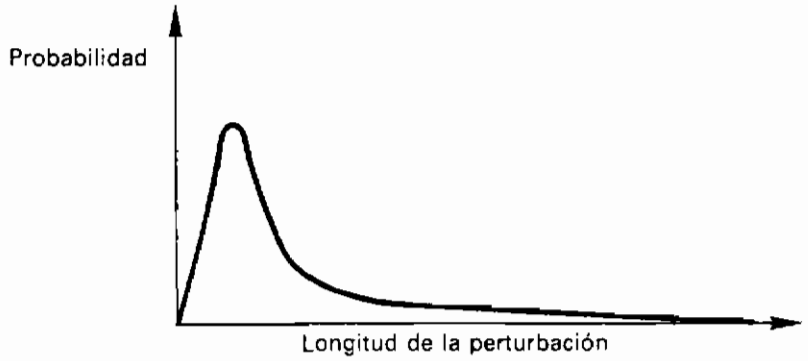


Fig. 20.1. *La probabilidad de superar una perturbación en un recurso determinado en función del tiempo que se necesita.*

Podemos aprender más de ese gráfico si miramos su integral. Esto nos dirá la probabilidad de superar la perturbación dentro de un período de tiempo dado. Pero, ganaremos mucho más si miramos el impacto que tiene sobre lo que se tarda en terminar un trabajo que pasa por varias actividades. La figura 20.2 representa la probabilidad de terminar un trabajo en función del tiempo transcurrido desde su lanzamiento. Podemos ver que esta curva nunca alcanza una probabilidad del 100 por 100. A medida que pasamos a intervalos de tiempo más y más largos aumenta la probabilidad de superar las perturbaciones, pero nunca se llega a la certeza absoluta.

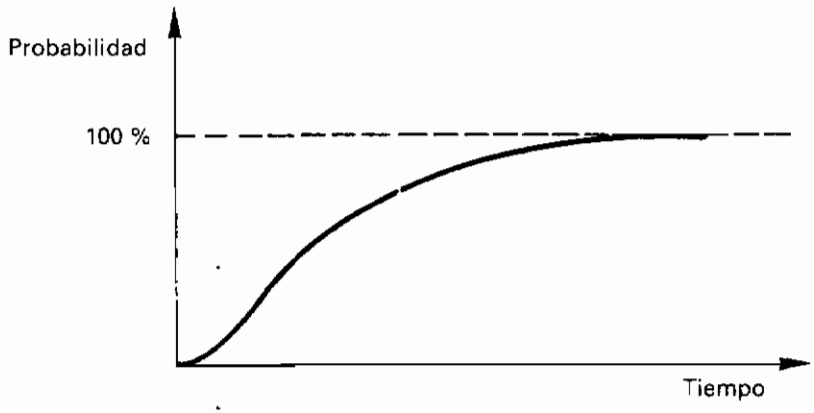


Fig. 20.2. *Probabilidad de terminar un trabajo que pasa por muchas operaciones en función del tiempo que transcurre desde el lanzamiento del trabajo.*

La determinación de la longitud del *buffer* de tiempo es una cuestión de juicio, y no es una tarea fácil ni trivial. Si queremos ser muy precavidos, y elegimos un *buffer* muy largo, podemos absorber cómodamente cualquier perturbación, pero, ¿a qué precio? Nuestro tiempo total de proceso será muy largo a priori, lanzaremos los materiales mucho antes de que podamos usarlos. Se inflarán los niveles medios de inventario en proceso y de producto acabado. Como resultado, aumenta nuestra necesidad de liquidez, se deteriora nuestra ventaja competitiva futura, y los costes financieros son más altos. Si elegimos *buffers* muy cortos, nuestro tiempo medio de respuesta será muy rápido, pero nos encontraremos con que tenemos que acelerar pedidos continuamente, y con que nuestras fechas de entrega no son demasiado fiables.

Vamos a insistir en ello otra vez. La determinación de la longitud de *buffer* implica la decisión de dirección más fundamental: la elección entre varias medidas. Cuando se eligen *buffers* largos se influye directamente en el nivel del *inventario* relacionado con tiempo (trabajo en proceso y producto acabado). Por lo tanto, influye indirectamente en las *ventas* futuras y el *gasto operativo*. La elección de *buffers* cortos influye directamente en el *gasto operativo* (aceleración de pedidos y control) y también en las *ventas* actuales y futuras (fechas de entrega poco fiables).

¿Quién debería tomar esta decisión? ¿Quién debería establecer la longitud de los *buffers* de tiempo? En la mayoría de las empresas no lo hace el director, ni siquiera el programador, lo suele hacer el operario de la carretilla.

> La decisión sobre la longitud del *buffer* debe estar en manos de las personas que son directamente responsables del resultado global de la empresa.

21

Buffer y orígenes de *buffer*

Como ya hemos visto, unos datos estadísticos más detallados no nos ayudarán a tomar mejores decisiones. Si queremos reducir la compensación entre medidas debemos tratar directamente con la causa de fondo de la necesidad de protección: Murphy. Si lo examinamos de cerca podemos ver que hay dos tipos de Murphy: uno implica cambios inesperados, como la rotura de un utillaje, el absentismo de un trabajador o el descontrol de un proceso que produce piezas defectuosas. Esta es la forma en la que normalmente representamos a Murphy. A este tipo de perturbación la llamo «Murphy puro». Pero, si nos fijamos en el flujo de un producto específico, en vez de en los recursos, nos damos cuenta de que hay otro tipo de perturbación.

Cuando un trabajo determinado llega a un recurso no limitado, a un recurso que tiene suficiente capacidad de protección, nos podemos encontrar con que el recurso está ocupado en otra tarea necesaria. El flujo de nuestro trabajo se interrumpe. El trabajo tendrá que esperar en la cola del recurso. Este fenómeno lo conocen bien todos los que hayan trabajado en una organización. Yo la llamo *disponibilidad no instantánea.*

Para entender mejor esta situación, vamos a establecer algunas relaciones generales entre los intervalos de tiempo que determinan el tiempo total de proceso *(lead-time).* Se calcula que, en la mayoría de los entornos, las perturbaciones son el factor principal en la determinación del tiempo total de proceso *(lead-time)* de un trabajo. El tiempo real de trabajo es, casi siempre, insignificante comparado con el impacto de Murphy. La mayoría de las empresas tienen datos de producción que apoyan esta afirmación.

Detengámonos, por un momento, para estimar el tiempo real de proceso que necesita una pieza de tipo medio. No, no consideremos un producto, ya que fabricamos y ensamblamos sus piezas en paralelo. Consideremos una pieza típica. ¡Otra vez! No la pieza más complicada, conside-

remos una pieza media. Y no consideremos todo un lote, sino sólo una pieza. Recordemos que, excepto en procesos muy especiales, podemos pasar una pieza a la siguiente fase del proceso antes de que el lote se haya terminado en la fase anterior. (Cuando estamos acelerando pedidos no llamamos «división y solape» de los lotes. En terminología TOC es simplemente la diferencia que hay entre el «lote de proceso» y el «lote de transferencia».) Si no somos fabricantes de piezas muy sofisticadas de la industria aeroespacial, el tiempo medio de proceso por pieza será probablemente inferior a una hora, y en industrias de semielaborados será de sólo unos segundos.

Ahora, vamos a compararlo con el tiempo medio de permanencia del inventario en nuestra fábrica. ¿Cómo vamos a calcular ese número? Es bastante fácil. Probablemente tenemos, o podemos conseguir, un cálculo bastante bueno de las vueltas de inventario del material en proceso y del producto acabado. Doce vueltas de inventario al año significa que la empresa mantiene en su poder cada pieza (desde el lanzamiento hasta su envío) durante un tiempo medio de un mes. Comparen este número, que se expresa en semanas, con el número anterior, que, probablemente, se expresa en minutos. Comparado con Murphy, el tiempo real de proceso es insignificante.

El tiempo total de proceso (*lead-time*) de los trabajos está dominado por Murphy. A efectos prácticos, el *buffer de tiempo* es el intervalo de tiempo en que adelantamos la fecha de lanzamiento de un trabajo con respecto a la fecha en la que está programado que lo consuma la limitación. En la mayoría de los casos no hace falta preocuparse de la insignificante corrección de tiempo necesaria para que se realice el trabajo en los distintos recursos.

¿Qué sucede con los dos tipos de perturbación, «Murphy puro» y «disponibilidad no instantánea»? ¿Cuál de los dos es más dominante en la determinación de la longitud del *buffer*? No lo sé. Hasta hace muy poco, debido a la falta de un sistema de información completo, no había una forma práctica de distinguirlos en la realidad. Aún, ahora, no hay suficiente experiencia práctica para responder con rigor a esta pregunta. Mi estimación personal es que deben ser más o menos iguales, pero el tiempo nos dirá.

La elección de *tiempo* como unidad básica de protección, en vez de *inventario*, les puede parecer tan natural que se estarán preguntando por qué estamos perdiendo el tiempo con algo tan obvio. Un momento, algunas de sus ramificaciones son casi contraintuitivas. Estamos acostumbrados a que los *buffers* sean algo físico. Por ejemplo, nadie se sorprenderá cuando oiga preguntas como «¿dónde están situados los *buffers*?» o, «¿cuánto inventario tenemos en el *buffer*?» Pero, ¿podemos seguir usando esa terminología? Ya no.

Si un *buffer* es un intervalo de tiempo, ya no podemos hablar en términos de localización o contenido del *buffer*. El tiempo no tiene localización ni contenido. Debemos introducir otro término que nos permita referirnos al sitio donde el inventario tenderá a acumularse, debido a lo adelantado de su lanzamiento.

¿Por qué? Por dos razones. La primera es que esta localización es muy importante porque es el sitio donde podemos empezar el seguimiento del impacto agregado de todas las perturbaciones. Esta consideración ha llevado a Eli Shragenhaim a sugerir el nombre de punto de control del *buffer*.

La segunda no se refiere al uso de los *buffers*, sino al proceso por el que los introducimos en nuestros planes. Como ya hemos dicho, el *buffer* es un intervalo de tiempo. ¿Cómo situamos este intervalo flotante de tiempo sobre el eje de tiempo?

Si lo pensamos un poco nos daremos cuenta de que sólo hay una opción. Los *buffers* existen para proteger el funcionamiento de la limitación. Así, cuando decidimos que una limitación tiene que hacer un trabajo en un momento específico, tenemos que lanzar el «material» requerido un *buffer* (intervalo de tiempo) antes que ese momento. (El «material» aparece entre comillas porque no es necesariamente un material físico. Pueden ser planos, o, incluso, un permiso para empezar el diseño.)

Ya hemos dicho que para calcular la fecha de lanzamiento del «material» tenemos que retroceder en el tiempo durante un intervalo igual al *buffer*, partiendo del momento en el que la limitación debe empezar a consumir el material. Así *fijamos* en nuestro plan de acción los intervalos de tiempo expresados por los *buffers*, o, dicho de otra manera: para calcular el programa de lanzamientos, tenemos que acoplar el extremo de los *buffers* de tiempo al programa de consumo futuro de la limitación.

En el eje de tiempo, el programa de consumo de las limitaciones es el *origen* de los *buffers* de tiempo. El *buffer* de tiempo retrocede en el tiempo a partir de este punto. Recordemos que aquí sólo estamos tratando con las limitaciones físicas. Las limitaciones políticas no deben protegerse, deben elevarse. Las limitaciones físicas, ya sean un recurso o un pedido, tienen una ubicación, y, por tanto, podemos llamar *origen de buffer* al sitio del que la limitación saca el «material» que consume. Esta terminología nos permite conectar mentalmente los *buffers* —que son intervalos de tiempo— con la posición física donde se acumulan los inventarios de protección resultantes.

Antes de abandonar este tema debemos aclarar un punto muy importante ¿cuántos tipos de *buffer* —y, por tanto, de orígenes de *buffer*— podemos tener en la empresa?

Por lo que hemos dicho hasta ahora parece obvio que tenemos más de un tipo de origen de *buffer* porque tenemos más de un tipo de limitación

física. Tenemos que proteger nuestros recursos limitados porque no queremos que se interrumpa su trabajo; para esto necesitaremos un *buffer de recursos*. El origen de *buffer* del *buffer* del recurso será el área que está delante del recurso limitado, y contiene inventario de trabajo en proceso.

También tenemos que proteger nuestras limitaciones de mercado, ya que queremos cumplir las fechas de envío. Para esto hará falta un *buffer de envíos*. El origen de *buffer* es, sencillamente, el muelle de envío o el almacén de producto acabado. Hay que hacer notar que, en los casos en los que se permite enviar con antelación, el origen de *buffer* del *buffer* de envíos no necesariamente contiene inventario de producto acabado. Contendrá la lista de pedidos que se han enviado antes de su fecha requerida.

¿Son éstos los únicos tipos de *buffer*? Sospecho que nos conviene introducir un tercer tipo, el *buffer de ensamblaje*. Para aclarar esta recomendación, vamos a examinar un caso en el que un recurso limitado alimenta una pieza a un ensamblaje. Esta operación de ensamblaje ensambla esa pieza con otras producidas por recursos no limitados. No queremos que la pieza que ha hecho el recurso limitado se quede esperando ante el ensamblaje hasta que lleguen las que han hecho los recursos no limitados. Recordemos que el concepto fundamental era *explotar* la limitación.

Permitir que el trabajo de la limitación retrase su conversión en ventas por culpa de otros recursos no limitados dista mucho de ser la forma ideal de explotación. Si queremos garantizar que no espere la pieza producida por la limitación, tendremos que conseguir que las otras lleguen antes. En otras palabras, tendremos que lanzar con antelación todas las demás piezas de los recursos no limitados. La necesidad del *buffer de ensamblaje* es casi evidente. El origen de *buffer* del *buffer* de ensamblaje sólo estará delante de los ensamblajes que usen, al menos, una pieza producida por una limitación. Este tipo de origen de *buffer* sólo contendrá piezas producidas por recursos no limitados.

Ahora que ya hemos aclarado algo de la terminología básica, vamos a describir, en términos generales, la estructura del segundo bloque fundamental de nuestro sistema de información, que trata, principalmente, sobre Murphy, sobre las perturbaciones.

22

Primer paso en la cuantificación de Murphy

Ya que hemos aclarado la terminología más relevante, deberíamos empezar a usarla para su propósito principal: la cuantificación de las perturbaciones. Puede que, una vez más, la forma de empezar sea excavando hasta las raíces. Puede que debamos empezar exponiendo la aproximación básica a Murphy, que se deriva de nuestro concepto del *buffer* de tiempo. Difiere algo de la aproximación tradicional, y tampoco coincide con la que presenta *calidad total* (TQM). Es, de alguna forma, una mezcla de las dos. En un pasado no muy lejano, la forma tradicional de tratar a Murphy era, aunque nos moleste admitirlo, aceptando su existencia y protegiendo casi todos los trabajos con tiempo e inventario. ¿Cuáles eran las explicaciones habituales sobre las acumulaciones de inventario de producto acabado? Normalmente, cualquier intento de analizar esto se encontraba con preguntas ofensivas como: ¿Qué pasa si mañana nos llega un pedido urgente por teléfono? ¿Qué hacemos? ¿Decimos al cliente que espere dos semanas? ¡Perderíamos el pedido!

También había una respuesta típica a la sugerencia de restringir la producción continua de los recursos no limitados. ¿Se acuerdan de cómo se recibía esa sugerencia? Por supuesto, una vez que los directivos se convencían de que no era una broma, de que lo estábamos sugiriendo en serio. *¿Y qué hacemos si se rompe una máquina? Sí, ahora tenemos tiempo de sobra, pero mañana puede pasar cualquier cosa.* En realidad, sabemos de sobra que todo trabajador, ingeniero, secretaria o director sólo se siente seguro si tiene un montón de trabajo esperándole.

Este tipo de tratamiento es absolutamente devastador. No es de extrañar que *calidad total* lo ataque obsesivamente. ¡No aceptéis a Murphy, no es algo inevitable, concentraos en eliminar problemas! Este es su mensaje principal. Algunos defensores de TQM han llegado a poner en su estandarte el lema: «hazlo bien a la primera». No dicen «no te equivoques nun-

ca»; lo que hacen es advertirnos en contra de la repetición continua de los mismos errores. No esperan que un prototipo funcione perfectamente la primera vez. Están intentando cambiar la actitud mental actual. Si se hace el mismo tipo de lote una y otra vez, ¿por qué debemos aceptar tranquilamente que las primeras piezas de cada lote sean siempre defectuosas?

Aunque apoyamos totalmente este mensaje tan radical, la aproximación que hemos elegido es mucho más moderada. Nuestro punto de arranque es muy distinto. Nos damos perfecta cuenta de que en la vida real no se puede eliminar a Murphy. Sí, la lucha contra Murphy es una tarea que merece la pena, pero no debemos dejarnos atrapar por nuestras propias palabras. Se puede, y se debe, luchar contra las perturbaciones, se puede reducir considerablemente a Murphy, pero, ¿se puede llegar a eliminar? Por tanto, nuestra aproximación debe consistir en intentar desarrollar un modo de operar que tenga en cuenta, en cualquier momento dado, que Murphy existe. Es más, debemos tener en cuenta que la lucha para reducir las perturbaciones no es una batalla corta y única, sino, exactamente, lo contrario. Es una guerra continua. Por tanto, deberíamos exigir que nuestro modo de operar nos guiará sabiamente en esta lucha interminable. Como punto de arranque, nos debería proporcionar una lista constante de Pareto: en qué problema nos debemos concentrar ahora, ¿cuál debemos resolver en segundo lugar?, y así sucesivamente.

Reconozcámoslo, aunque limitáramos el tema de Murphy a los problemas de calidad, ¿cuántos problemas de calidad tiene una planta? ¿Cientos? ¿Miles? Probablemente sean millones. Un esfuerzo hacia la «calidad total» sólo puede ser efectivo si se guía por una lista de Pareto bien pensada y permanentemente actualizada.

¿Cómo vamos a inventarnos un modo de operar tan deseable? Precisamente la terminología que hemos definido hasta ahora es casi suficiente para forzarnos hacia la dirección correcta. Tenemos que estar bien sintonizados con esta nueva terminología, y, además, debemos tener cuidado de que nuestra inercia no nos retrase; una inercia que procede de antiguos hábitos y de nuevos (aunque no muy fundados) y entusiastas movimientos.

Buffers, la misma palabra que tantas veces hemos repetido en los dos últimos capítulos, indica que nuestra aproximación reconoce la existencia de perturbaciones. Pero, ¿estamos intentando proteger todas y cada una de las tareas? ¡Por supuesto que no! El término *origen de buffer* muestra claramente que somos muy selectivos en lo que tratamos de proteger. De hecho, hemos elegido una aproximación muy pragmática.

¿Cuál es una de las formas más populares de describir a Murphy? ¿Cual es la probabilidad de que una rebanada de pan caiga con la mantequilla hacia abajo? Es directamente proporcional al precio de la alfombra sobre la que cae. ¿Intentamos proteger todas nuestras alfombras? No. Muchas de ellas se pueden limpiar con facilidad: los recursos no limitados.

Dejando aparte las bromas, ¿cuál es la aproximación fundamental inherente a la terminología del *buffer* y del *origen de buffer*? Por un lado, reconocemos, sin lugar a dudas, la existencia de Murphy, si no, no hacen falta los *buffers*. Al mismo tiempo, el término origen de *buffer* revela que somos muy tacaños a la hora de permitir un *buffer*. En otras palabras, nos damos perfecta cuenta de que la protección tiene un precio; inflar el inventario también es perjudicial. Por tanto, elegimos la protección sólo cuando hay riesgo de perder algo más importante: ventas.

Esta aproximación nos lleva a intentar reducir, aún más, el precio que pagamos por la protección. Recordarán que la determinación de la longitud de los *buffers* de tiempo era una cuestión de juicio. ¿Cómo verificamos si la longitud elegida representa de verdad el equilibrio que teníamos in mente? Tiene que hacer algún mecanismo para comprobar si estamos demasiado expuestos por haber elegido un *buffer* corto, o si, en vez de estar algo paranoicos, como debe ser, estamos histéricos.

Al explicar la naturaleza probabilística del tiempo total de proceso (*lead-time*) hemos dado una indicación clara de la técnica que deberíamos usar. ¿Qué se demostró en la figura 22.2? Determinamos la longitud del *buffer* de tiempo esperando que un cierto porcentaje predeterminado de las tareas iba a estar en el origen de *buffer* en el momento necesario o antes. Por supuesto, suponíamos que se iba a lanzar un *buffer* de tiempo antes de la fecha de consumo. Lo que deberíamos hacer ahora es comprobar la situación en la realidad.

Si cumplimos las fechas de lanzamiento, y si nuestra estimación del nivel de perturbación es más o menos correcta, deberíamos encontrarnos con lo que esperábamos. Pero si encontramos en el origen de *buffer* un porcentaje de tareas más alto del esperado, tenemos una clara indicación de que sobreestimamos la longitud del *buffer* y, por tanto, debemos reducirla. Si ocurre lo contrario, un porcentaje mayor de lo esperado, no está llegando al origen de *buffer* ni siquiera en las fechas previstas de consumo, deberíamos incrementar la longitud del *buffer*. Sí, no nos gusta tener que hacerlo, pero, mientras Murphy siga activo en nuestra organización, tal y como nos están indicando los retrasos, ése es el precio que tenemos que pagar para proteger las ventas. Esto nos lleva directamente al siguiente tema.

¿Cómo podemos reducir el precio de la protección? Todo el que haya estado algún tiempo en una empresa sabe que el tiempo total de proceso (*lead-time*) de un trabajo es una entidad muy flexible. Aunque normalmente se tarde una semana en hacer un trabajo, si es urgente y nos ocupamos personalmente de él, lo podemos hacer pasar por las operaciones en me-

nos de un día. Es cierto que, si tratamos de acelerarlo todo, al mismo tiempo creamos el caos. Pero, ¿podemos acelerar selectivamente para reducir la longitud de los *buffers?*

No es sorprendente que la respuesta sea *sí;* podemos usar esta aceleración selectiva para reducir el tiempo total del proceso. Volvamos a examinar la figura 22.3, el gráfico que muestra la probabilidad de duración del tiempo total del proceso de un trabajo, para ver cómo se puede hacer sistemáticamente. Para proteger adecuadamente las limitaciones, tenemos que elegir unos *buffers* de tiempo con longitud suficiente para garantizar una alta probabilidad de que los trabajos lleguen a tiempo al origen de *buffer*. La figura 22.3 muestra claramente la naturaleza gradual del incremento de probabilidades en este rango alto de porcentajes. Para aumentar la probabilidad del 90 al 98 por 100 necesitamos, casi, doblar la longitud del *buffer* de tiempo. Vamos a concentrarnos en los trabajos que ya han atravesado el 90 por 100 de probabilidad de llegar al origen de *buffer*. De éstos elegimos los que todavía no han llegado y les echamos una mano, los aceleramos.

Este incremento tan gradual es la propiedad que invita al uso de la aceleración. Supongamos que decidimos participar activamente en el proceso: no sólo nos conformamos con lanzar los trabajos un *buffer* de tiempo antes de su consumo programado, sino que, además, aceleramos trabajos selectivamente. Concentrémonos en los trabajos que ya han cruzado el 90 por 100 de probabilidad de llegar al origen de *buffer*. De ellos, aceleramos, o ayudamos, a los que todavía no han llegado. Como ya hemos dicho, cuando aceleramos un trabajo reducimos considerablemente su tiempo total de proceso. Por tanto, los trabajos que aceleramos no necesitarán tantísimo tiempo adicional para llegar al origen de *buffer;* llegarán en un espacio de tiempo relativamente corto.

Al acelerar hemos manipulado el extremo del gráfico, haciéndolo más inclinado. Con esta forma de operar necesitaremos un *buffer* de tiempo relativamente corto para garantizar una probabilidad alta de que el trabajo llegue. ¿Cuántos trabajos tendremos que acelerar? Si seguimos con los números algo arbitrarios que hemos usado, tendremos que acelerar un 10 por 100 de los trabajos, un esfuerzo bastante soportable. Si operamos de esta forma, deberíamos llamar *zona de aceleración* al intervalo de tiempo en el que aceleramos trabajos.

Por supuesto que, una vez más, nos enfrentamos a una elección. Si queremos acelerar menos trabajos, tendremos que empezar a hacerlo más tarde y, por tanto, necesitaremos un *buffer* más largo. Es una elección entre inventario y gasto operativo. Tengan en cuenta que el cambio a esta forma de operar reducirá drásticamente los esfuerzos de gestión que son necesarios hoy en día para tratar con los «fuegos imprevistos». Normalmente ya

disponemos de los recursos de gestión necesarios para llevar a cabo la «aceleración planificada», y, por tanto, no se trata realmente de una elección. En la mayoría de los casos es sólo una reducción neta del precio que se paga para proteger las ventas. ¿Cómo podemos reducir, aún más, ese precio? Hasta que a alguien se le ocurra algo más brillante, parece que lo único que podemos hacer es atacar de frente a Murphy. Pero, un momento, no sigamos haciéndolo a escopetazos. Hemos desperdiciado nuestro tiempo y esfuerzos con demasiada frecuencia atacando el problema que sabíamos resolver, para encontrarnos con que ni siquiera habíamos empezado a tratar el problema que deberíamos haber resuelto. Si hemos conseguido concentrar la protección donde de verdad hace falta, tiene que haber una forma de concentrar los esfuerzos que hacemos para reducir la necesidad de protección.

Es posible que, si seguimos insistiendo, encontremos una forma de identificar los problemas que deberíamos resolver. Ya estamos haciendo el trabajo de seguimiento del origen de *buffer* para controlar la longitud del *buffer*. Es en este punto donde aparece el impacto acumulado de todas las perturbaciones. Así pues, parece razonable suponer que debe de haber una forma de utilizar este mismo trabajo como trampolín de nuestro ataque a Murphy.

23

Gestión de los esfuerzos de mejora de procesos locales

Ya nos hemos dado cuenta de que si queremos reducir el precio que pagamos por la protección, tenemos que concentrarnos en los trabajos que llegan más retrasados al origen de *buffer*. Conseguir que llegue antes un trabajo, que ya iba a llegar a tiempo en todo caso, no nos ayuda en absoluto a mejorar el rendimiento global. Hemos tratado los trabajos retrasados individualmente: puede que consigamos mucho más si atacamos las causas más comunes de estos retrasos.

Vamos a examinar esta idea tan interesante: utilizar el esfuerzo realizado individualmente en los trabajos para determinar las causas más generales de sus retrasos. ¿Cuál es la secuencia de acciones que llevamos a cabo para acelerar trabajos? Primero determinamos qué trabajos se supone que debe de haber en el origen de *buffer*. «Se supone que debe de haber» significa con una probabilidad más alta que, digamos, un 90 por 100. Después, comprobamos si, de hecho, están en el origen de *buffer*. Si no encontramos allí alguno de estos trabajos, empezamos el proceso de aceleración. La primera acción es encontrar dónde se ha atascado el trabajo que está retrasado. Una vez encontrado, actuamos para que siga adelante inmediatamente. Pero, ahora vamos a añadir otro pequeño detalle; vamos a registrar dónde (delante de qué recurso) hemos encontrado el trabajo atrasado. La repetición de este proceso con cada trabajo que se retrase resultará en una lista de recursos, y algunos de ellos aparecerán en la lista muchas veces. ¿Cuál es el significado real de esa lista?

Supongamos que hay un problema en un centro de trabajo; es bastante probable que este problema afecte a la mayoría, si no a todos, de los trabajos que hace ese recurso. Es más, si un problema de un recurso afecta a todos los trabajos, y otro problema en otro recurso sólo afecta a un trabajo, ¿qué problema es más importante? Esta línea de razonamiento, que reconoce que muchos problemas finales tienen una causa común, nos

lleva a reconocer que los recursos que aparecen con frecuecia en nuestra lista no lo hacen por capricho estadístico.

Si un recurso aparece con frecuencia es porque contiene la causa de un problema que es común a muchos trabajos; puede ser un proceso defectuoso, puede ser un ajuste poco fiable, puede ser una falta de capacidad de protección, o puede ser que el recurso simplemente esté mal gestionado. En cualquier caso, si tratamos con el problema de fondo en el nivel del recurso, en vez de en el nivel del trabajo, no tendremos que acelerar una y otra vez. Elimiraremos, hasta cierto punto, las razones de la aceleración. Si hacemos esto, si guiamos nuestros esfuerzos de mejora de JIT y TQM por esta lista de recursos problemáticos, podremos ir reduciendo la longitud de los *buffers* de tiempo gradual, pero constantemente.

¿Significa esto que la longitud de los *buffers* de tiempo siempre se reducirá? No necesariamente. Algunas veces (y debido a la reducción del tiempo total de proceso —*lead-time*— es muy probable) las ventas aumentarán. El aumento en las ventas conducirá a un aumento de las cargas de producción que absorberá parte de la capacidad de protección disponible y, por tanto, para compensar hará falta aumentar la longitud del *buffer* de tiempo. Si se hace correctamente —sin perder de vista el objetivo de ganar más dinero— este proceso llevará a oscilaciones controladas de los niveles de inventario.

Puede que nos convenga ampliar nuestros esfuerzos para que nuestra lista de recursos problemáticos resulte más fiable. Recordemos que el tiempo que se tarda en encontrar un problema de fondo suele ser más corto que el tiempo necesario para eliminarlo. Así pues, no deberíamos conformarnos con los datos que reunimos durante nuestras actividades de aceleración de trabajos, deberíamos ampliar nuestro seguimiento para cubrir tareas que aún no pretendemos acelerar, tareas que todavía tienen tiempo de sobra, pero que aún no han llegado al origen de *buffer* aunque haya una probabilidad razonable de que lleguen. Para mantener nuestros esfuerzos en un nivel razonable, vamos a suponer que hacemos un seguimiento (pero sin expeditar todavía) a todo trabajo cuya probabilidad de llegada al origen de *buffer* haya superado el nivel del 60 por 100. El recurso donde se encuentre el trabajo se añadirá a la lista que generamos cuando estábamos acelerando trabajos.

Probablemente, esta ampliación del trabajo de seguimiento no sólo enriquecerá la estadística, sino que, además, la mejorará. Verán, si empezamos el seguimiento muy tarde, sólo cuando la probabilidad de llegar ha superado el 90 por 100, habrá muchos trabajos que ya no se encuentren en el recurso que causó el retraso. Puede que para entonces ya hayan pasado el recurso problemático, y, por tanto, será raro que nuestra aceleración de trabajos llegue a detectar los recursos problemáticos del principio del

proceso. En cualquier caso, el seguimiento de dónde quedan retenidos los trabajos que se atrasan (no sólo los urgentes) mejorará nuestra estadística considerablemente. No debemos llegar hasta el punto de seguir todos los trabajos desde su lanzamiento; no siempre es mejor hacer más. Si empezamos el seguimiento inmediatamente después del lanzamiento nos encontraremos con más del doble de trabajo, y, además, la validez de la lista resultante se difuminará.

Resumiendo, la gestación de los *buffers* nos proporciona varios beneficios. Nos permite determinar mejor la longitud requerida de acuerdo con el nivel existente de perturbaciones: «cuantificar el ruido». Nos permite acelerar tareas, sistemática y metódicamente, para reducir el tiempo total de proceso *(lead-time)*. Después, si localizamos los sitios donde se encuentran los trabajos retrasados, y asignamos prioridades de acuerdo con el número de veces que aparece en esa lista cada recurso (probablemente con un factor de ponderación que sea adecuado), tendremos una lista de Pareto que debe guiar nuestros esfuerzos de «mejora de la productividad». Pero, además, hay otro beneficio que probablemente es aún más importante.

Lo que debemos tener presente es que, cuando abordemos un recurso problemático, un recurso que aparezca en la lista con frecuencia, nos podemos encontrar con que sus procesos están en perfectas condiciones. Este recurso no aparece en nuestra lista por tener problemas de proceso, sino porque no tiene suficiente capacidad de protección. Así pues, la gestión del *buffer* nos proporciona el único mecanismo conocido que permite calcular la capacidad de protección que necesitan nuestros recursos.

Cuantificar Murphy es cuantificar la longitud del «buffer» y la cantidad de capacidad de protección que se necesita.

En este punto es necesario hacer una advertencia. Todo lo que se ha dicho hasta ahora se puede hacer fácilmente de forma manual, excepto, quizá, los extensos trabajos de seguimiento, que probablemente deberían hacerse con un informe de recursos más riguroso sobre las transacciones reales. Pero, hay otro punto importante que probablemente no se pueda hacer de forma totalmente manual. Es el tema de superar la necesidad de la parte fluctuante de la capacidad de protección ajustando la longitud de los *buffers* de tiempo. Vamos a aclarar este tema tan delicado. Cuando estamos tratando con el asunto de la capacidad de protección, tenemos que acordarnos de que una de las principales razones del tiempo total de proceso de los trabajos es la «disponibilidad no instantánea» de los recursos. El otro lado de la moneda es que las fluctuaciones en la mezcla de producto pueden afectar significativamente a la necesidad de capacidad de protección, y, por tanto, un recurso puede aparecer frecuentemente en nuestra lista de seguimiento debido a esas fluctuaciones. Lo que debemos

tener presente es que este problema no lo causa la protección de las limitaciones, lo que pasa es que se revela gracias a nuestra forma sistemática de trabajar. Pero, quizá ahora, debamos aclarar por qué decimos que es un problema. Es un problema porque la amplitud de la capacidad flexible de protección es bastante limitada, ya que normalmente se reduce a las horas extras que haya disponibles. Hoy en día abordamos esto añadiendo más capacidad permanente, aunque esta capacidad adicional sólo es útil, por definición, durante una parte del tiempo. De hecho, nos vemos obligados a añadir capacidad sobrante. A primera vista parece que estamos condenados a sufrir la naturaleza estocástica de nuestro entorno, y, de hecho, éste es el caso, al menos en lo que se refiere a los cálculos manuales. Pero hay una salida, siempre que contemos con la enorme paciencia de un ordenador para realizar cálculos voluminosos sin morirnos de aburrimiento.

La forma de vencer la necesidad de añadir capacidad permanente, para hacer frente a los cambios frecuentes de la mezcla de producto, emana directamente de la relación que existe entre la capacidad de protección y la longitud de los *buffers*. Pero esto requeriría programar de acuerdo con *buffer* dinámicos, y, por tanto, es mejor posponer todo este tema hasta el momento que tratemos del bloque de PROGRAMACION.

Si repasamos lo que hemos dicho en este capítulo, parece que hemos conseguido mucho más de lo que pretendíamos al empezar. Puede que debamos continuar y seguir empujando un poco en la misma dirección. Nos disponíamos a «cuantificar Murphy», y lo conseguimos con el mecanismo de control de la longitud de los *buffers*. Pero, hemos conseguido más, hemos encontrado un mecanismo sistemático para controlar la aceleración de trabajos, no cuando ya se ha hecho el daño, ni a base de «apagar fuegos», sino de forma constructiva, de forma que no apunta a un trabajo urgente específico, sino que se dirige a reducir el *lead-time* global de todos los trabajos. De hecho, hemos conseguido algo que podríamos empezar a llamar *control*.

El seguimiento de los puntos que están retrasando los trabajos nos proporciona un mecanismo para construir una lista de Pareto, tanto para guiar nuestros esfuerzos de mejora local como para cuantificar la cantidad de capacidad de protección que se necesita en cada recurso. Como ahora estamos hablando de hacer el seguimiento de un número considerable de tareas atrasadas, podemos hacerlo de forma más efectiva con un informe sobre transacciones, en vez de con el método, más engorroso, de hacer el seguimiento a lo largo de la explosión de los trabajos.

El informe de transacciones acerca aún más el tema al área de control y, al mismo tiempo, hace surgir el bien conocido y preocupante tema de la exactitud de las transacciones, o quizá, debamos decir de la inexactitud de

las transacciones. Normalmente esta inexactitud reduce la utilidad de los datos que hay en los informes de transacciones, pero quizá podamos matar unos cuantos pájaros de un solo tiro. ¿Podemos mejorar significativamente la exactitud y puntualidad de los informes de transacciones y, al mismo tiempo, contestar otra pregunta de dirección que todavía está pendiente?

Reconozcámoslo, la forma de mejorar sustancialmente la puntualidad y la exactitud de los informes de transacciones es convirtiéndolo, de alguna forma, en el principal interés de las personas que tienen que informar. ¿Cuál es el principal interés de una persona sino sus propias medidas? Ya estamos a este nivel, buscando datos sobre los pasos que avanza cada trabajo individual. ¿Por qué no damos otro pequeño paso e intentamos averiguar cómo transformar este esfuerzo en la respuesta al viejo problema de las medidas de rendimiento local?

No olvidemos que la cuestión de cómo medir, objetiva y constructivamente, los rendimientos locales obtenidos es una de las cuestiones más candentes de la dirección. Si conseguimos contestar satisfactoriamente a esta pregunta, podremos llamar *control*, justificadamente, a esta parte de nuestro sistema de información.

24

Medidas del rendimiento local

¿Hacia dónde nos dirigimos? Hacia el hecho de que, además de instituir la gestión del *buffer* y, además, de enfocar nuestros esfuerzos en la mejora de los procesos físicos, debemos instituir unas medidas adecuadas del rendimiento local. Unas medidas que, por el mero hecho de existir, induzcan a las áreas que se miden a hacer lo correcto para el bien global de la empresa. Para poder beneficiarnos de la fuerza más potente que hay en la empresa —la intuición humana— necesitamos introducir un sistema mucho mejor para medir el rendimiento. La medida del rendimiento local, a base de eficiencias y varianzas, lo único que consigue es que nuestros trabajadores hagan exactamente lo contrario de lo que queremos. Hoy en día se conoce bien la necesidad de unas medidas mejores del rendimiento local. Desgraciadamente, hay quien está intentando satisfacer esta necesidad candente a base de «medidas no financieras», como defectos por millón o porcentaje de pedidos que se han enviado tarde. Como ya hemos dicho, mientras la meta de la empresa sea ganar dinero, las unidades de medida deben incluir —por definición— la unidad de dinero, el signo del dólar. Tenemos que profundizar más para revelar la forma adecuada de diseñar las medidas locales, debemos hacerlo. Recordemos que el tema de las medidas es, probablemente, el tema más sensible de cualquier organización:

Dime cómo me mides y te diré cómo me comporto.

Yo no dudo en llamar «control» a la medida del rendimiento. Me doy cuenta de las connotaciones negativas que la palabra conlleva, por culpa de sistemas distorsionados que están a la orden del día, pero, aun así, las medidas son control, autocontrol, tanto como control del sistema.

Vamos a insistir otra vez, porque la palabra *control* es una de las palabras que peor se han utilizado. Por ejemplo, ¿a qué nos referimos cuando

usamos el término «control de inventario»? A la capacidad de conocer dónde está el inventario. Esto no es control en absoluto, sólo es capacidad de reunir datos. Para mí, y probablemente para ustedes, control significa saber dónde están las cosas con respecto a dónde deberían estar, y quién es el responsable de las desviaciones que se produzcan. Y no de forma esporádica, caso a caso, sino con un procedimiento que asigne continuamente un valor numérico a cada una de las áreas responsables de la ejecución.

Esta es la única forma de dar a la gente la muy necesaria retroinformación sobre el resultado final de sus acciones. Esto es aún más importante en empresas donde hay mucha gente implicada en todo el proceso, y donde el resultado final se suele obtener en algún departamento remoto. Pero esta forma de atacar el tema de las medidas del rendimiento local nos revela un aspecto muy interesante. Las medidas de rendimiento local no deberían enjuiciar el resultado final, más bien deberían enjuiciar la influencia que tiene el área que se mide sobre el resultado final. Las medidas de rendimiento local deberían enjuiciar la calidad de ejecución de un plan, y este juicio debería ser completamente independiente del juicio sobre el plan en sí.

Esto es especialmente importante en nuestras organizaciones, donde es frecuente que el nivel responsable de la ejecución tenga muy poco que ver con el desarrollo previo del plan. Si no tenemos la precaución de juzgar por separado el plan y su ejecución podríamos recompensar a un departamento por un rendimiento magnífico cuando, en realidad, el rendimiento fue bajo, pero el plan era lo suficientemente bueno como para cubrir el bajo rendimiento. O aún peor, podríamos condenar a un departamento, cuyo rendimiento fue excelente, por culpa de unos problemas que, en realidad, estaban en el plan original.

Estas advertencias nos llevan directamente a la conclusión de que cuando enjuiciamos adecuadamente el rendimiento local, de hecho, estamos enjuiciando desviaciones, desviaciones en la ejecución de un plan predeterminado. La calidad del plan en sí debe juzgarse por las medidas que hemos estado usando: ingresos netos, inventario y gasto operativo. Pero, ¿cuáles son las medidas adecuadas para las desviaciones?

Lo primero que tenemos que reconocer es que las desviaciones del plan pueden adoptar dos formas distintas. El tipo de desviación más directa es *«no hacer lo que se debe hacer»*. Este tipo de desviación, que es a la que, con razón, se le presta más atención hoy en día, sólo influye en una de las medidas. Si no hacemos lo que se supone que debemos hacer, los ingresos netos disminuirán. Pero hay un segundo tipo de desviación. Sí, en efecto, es *«hacer lo que no se debe hacer»*. ¿En cuál de las medidas tendrá un impacto adverso este tipo de desviación? En el inventario, por supuesto.

¿Cómo podemos convertir sistemáticamente en un conjunto de medi-

das sólido y bien definido estas observaciones tan obvias? Puede que primero tengamos que aclararnos a nosotros mismos la unidad de medida con la que estamos tratando aquí. Las *desviaciones* son, sin duda alguna, un tipo de pasivo, seguro que nadie dirá que las desviaciones son un activo. ¿Cuál es la unidad de medida de un pasivo? ¿Existe, de hecho, una unidad genérica? Intentemos averiguarlo. Tomemos un ejemplo claro de pasivo, un préstamo bancario. Esto, sin duda, es un pasivo. ¿Cuál es la unidad de medida de un crédito? ¿Qué medida usamos para medir el perjuicio de ese pasivo?

Cuando recibimos un crédito de un banco el perjuicio en el que incurrimos (el interés que tenemos que pagar) no es sólo función de la cantidad de dólares que nos prestan. También nos preguntan por el tiempo que vamos a tener el crédito en nuestro poder. La cantidad absoluta de dinero que tenemos que pagar como intereses es una función de la multiplicación de los dólares que nos prestan por el tiempo que tenemos el préstamo en nuestro poder. La unidad de medida de un préstamo es *dólares* multiplicando por *días*, abreviándolo, *dólares/día*. ¿Podría ser que cada vez que tratamos genéricamente con pasivos, y especialmente con las desviaciones, la unidad de medida adecuada fuera *dólares/día*? Para examinar esta especulación tan atrevida vamos a empezar por considerar toda la planta como la unidad local que queremos medir. A nivel de planta, ¿cuáles son los resultados finales de las desviaciones del primer tipo, no hacer lo que se debe? La respuesta es obvia: el resultado será que no enviaremos a tiempo. ¿Cuál es la medida adecuada para los fallos en envíos?

Como en una empresa normal los pedidos varían considerablemente en valor monetario, y también varían los precios de venta de los productos individuales, no podemos usar como unidad de medida el número de pedidos o el número de unidades que no se han enviado a tiempo (aunque, sorprendentemente, los porcentajes basados en esas unidades son bastante comunes). Es más, parece razonable que tengamos en cuenta, de alguna forma, el número de días que se ha retrasado el pedido. Un retraso de un día en un pedido no se puede tratar de la misma forma que un retraso de un mes en ese mismo pedido.

Una forma lógica de medir los retrasos en las fechas requeridas de entrega es la siguiente: para cada pedido retrasado deberíamos multiplicar el valor en dólares de su precio de venta por el número de días que se esté retrasando el pedido. Si sumamos las multiplicaciones de todos los pedidos retrasados tendremos una buena medida de la magnitud de la desviación de la planta con respecto a su obligación de enviar los pedidos a tiempo. No es de extrañar que la unidad de medida resulte ser *dólares/día*, donde los dólares reflejan los precios de venta y los días reflejan los períodos de retraso de los pedidos.

¿Podemos usar la misma técnica del primer tipo en las secciones de una organización, tales como departamentos o, incluso, recursos individuales? No veo ninguna razón por la que no se pueda hacer. A nivel de departamento, ¿qué podemos usar como valor en dólares? No hay ninguna razón para no seguir usando el precio de venta final del pedido. Al final, es probable que no podamos enviar a tiempo el pedido debido a la desviación de este departamento. Sí, este departamento sólo tiene una pieza, pero, ¿cuáles son las verdaderas ramificaciones? ¿Cuál es el daño potencial resultante para la empresa? No es el precio de esa sola pieza, es el retraso en recaudar el dinero de todo el pedido.

¿Qué pasa si resulta que dos departamentos están retrasando el mismo pedido, porque cada uno se atrasa en un trabajo distinto? ¿Por qué no usamos el precio total de venta del pedido para cada uno de los departamentos? No tenemos que repartir las culpas, estamos intentando mostrar a cada departamento el perjuicio que sufrirá la empresa debido a las desviaciones del departamento. Recordemos que no vamos a poder enviar debido a las desviaciones de cada departamento por separado. Es suficiente con que sólo uno de ellos no consiga recuperarse para que no podamos enviar a tiempo. Además, no tenemos ninguna intención de sumar todas las desviaciones de los departamentos para llegar a una medida global de la empresa. Por lo tanto, no se va a introducir ninguna distorsión por el hecho de usar el precio total de venta del pedido en cada departamento que esté retrasando ese pedido.

¿Qué vamos a usar como punto de referencia de los días? ¿Desde qué fecha vamos a considerar que un departamento está retrasando un trabajo? No parece muy correcto esperar hasta la fecha de envío del pedido, sería incurrir en un riesgo demasiado grande. Debemos recordar que ya que las medidas de rendimiento local se basan en las desviaciones, deberían avisar al área que se está midiendo de que algo está yendo mal. Si esperamos a esa fecha tan tardía para avisar, tendremos garantizado el daño. En ese momento lo único que se puede hacer es intentar minimizar los daños. Esto no parece casar bien con nuestro último descubrimiento de que debemos realizar el 100 por 100 de las entregas a tiempo. ¿Qué deberíamos escoger como punto de arranque para señalar y, por lo tanto, cuantificar y empezar la acumulación de las desviaciones? Quizá, sería una elección razonable empezar a señalarlas cuando la desviación de un departamento ya haya provocado una acción correctiva de la organización. Ya hemos definido esos puntos en el tiempo. ¿Recuerdan la zona de aceleración? Vamos a repasar el significado de esa zona. A veces, una desviación en un departamento local provoca una acción de la organización. Esto sucede siempre que un trabajo no llega a su origen de *buffer*, a pesar de que ha pasado el tiempo suficiente desde su lanzamiento, para que esperemos,

con una probabilidad bastante alta (digamos más del 90 por 100), la llegada del trabajo. Por lo tanto, podemos empezar a contar los días desde el momento en que el trabajo entra en la zona de aceleración, en vez de usar la fecha de entrega del pedido. Esto nos dará algún tiempo para corregir la situación antes de que el perjuicio para la empresa sea un *fait accompli*.

Puede que esto no sea suficiente. Puede que tengamos que empezar a contar antes. Podemos decir que, en realidad, se ha provocado una acción de la organización (no un acción del departamento) mucho antes de que empezaran las actividades de aceleración. Empezó cuando, debido a las desviaciones, empezamos a hacer el seguimiento de la situación de este trabajo. ¿No deberíamos contar los días desde ese punto? Puede ser.

Así pues, vamos a multiplicar el precio de venta del pedido por el número de días que han pasado desde que empezó una acción correctiva de la organización. ¿Qué vamos a hacer con el número resultante? ¿A quién se lo vamos a asignar? Al causante de que el trabajo se retrasara. ¿Cómo vamos a averiguar quién es el verdadero causante? ¿Abriendo una agencia de investigación? Tendrá que ser mucho más grande que el FBI, por no mencionar el tipo de ambiente polémico que introduciríamos en la organización si lo hiciéramos. Recordemos que el tema más sensible de una organización es el de las medidas de rendimiento individual.

¿Qué piensan ustedes de la atrevida sugerencia de asignar la responsabilidad, los dólares/día resultantes, al departamento donde esté el trabajo ahora mismo? Asignarlo basándonos en los hechos actuales, sin considerar en absoluto qué departamento fue el que de verdad provocó la desviación, parece injusto. Puede que el trabajo retrasado haya llegado ahora mismo, hace un minuto, a este departamento, y, ahora, vamos a poner toda la carga de los dólares/día resultantes sobre los hombros de alguien que claramente no tiene nada que ver con el retraso.

Puede que a primera vista parezca injusta esta sugerencia tan directa, pero, un momento, ¿qué es lo que de verdad estamos intentando conseguir? No olvidemos lo que nos disponíamos a hacer. Estamos intentando medir el rendimiento local. ¿Con qué propósito? Para motivar a las entidades locales a hacer lo que es bueno para la empresa como un todo. Examinándolo desde esta perspectiva, ¿cuál creen que será la reacción de un departamento al que se acaba de cargar con un trabajo retrasado que arrastra consigo un número considerable de dólares/día? Todos sabemos cuál será la reacción, por supuesto, después de maldecir al departamento que empezó todo el lío.

Este departamento que ahora tiene el trabajo retrasado llevará a cabo todas las acciones posibles para librarse de la «multa», entregando ese trabajo retrasado al siguiente departamento lo más rápidamente posible.

Al pasar el trabajo al siguiente departamento, pasará con él la penalización de los dólares/día. Si por cualquier motivo no se actúa, la penalización crecerá rápidamente. Recordemos que con cada día que pasa aumenta la penalización de dólares/día. De hecho, conseguimos exactamente el tipo de actuación que deseábamos: un trabajo retrasado pasará rápidamente de un departamento a otro, como si fuera una patata caliente. La medida en sí está provocando la autoaceleración.

¿Qué hay del hecho de que hacer esto sea simplemente injusto? ¿Qué pasa con las repercusiones sociales resultantes? Cuando examinamos las medidas a lo largo del eje de tiempo, que es como debe hacerse, en vez de en un punto, descubrimos que esta técnica de fuerza bruta es, en realidad, una medida muy justa. Veamos, por ejemplo, los siguientes tres gráficos que representan la medida de los *dólares/día* de *ingresos netos*, en función del tiempo, para tres departamentos distintos.

¿Qué podemos deducir del departamento que muestra sus resultados en la figura 24.1? Los picos nos cuentan toda la historia. Seguro que este departamento no es la fuente de las desviaciones. Es un departamento que las recibe y hace un trabajo excelente pasando rápidamente al siguiente departamento los trabajos retrasados. El resultado final es un rendimiento muy bueno, el promedio de dólares/día es muy bajo.

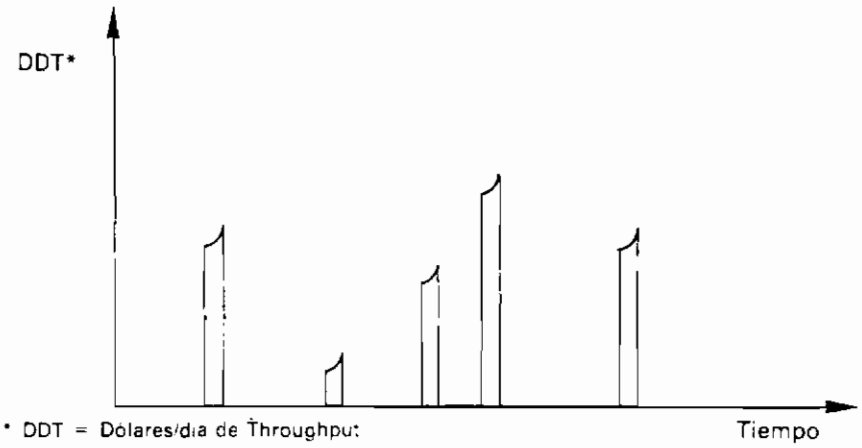


Fig. 24.1. *Los dólares/día de Throughput de un departamento en particular como función del tiempo.*

El segundo departamento muestra (en Fig. 24.2) una historia bastante distinta. Ciertamente, no es el origen de las desviaciones; cuando llegan los trabajos ya están retrasados. A pesar de ello, su forma de gestionar los

trabajos retrasados es muy torpe; se quedan mucho tiempo en el departamento, aumentando la urgencia. Como resultado, vemos que, por supuesto, los dólares/día aumentan.

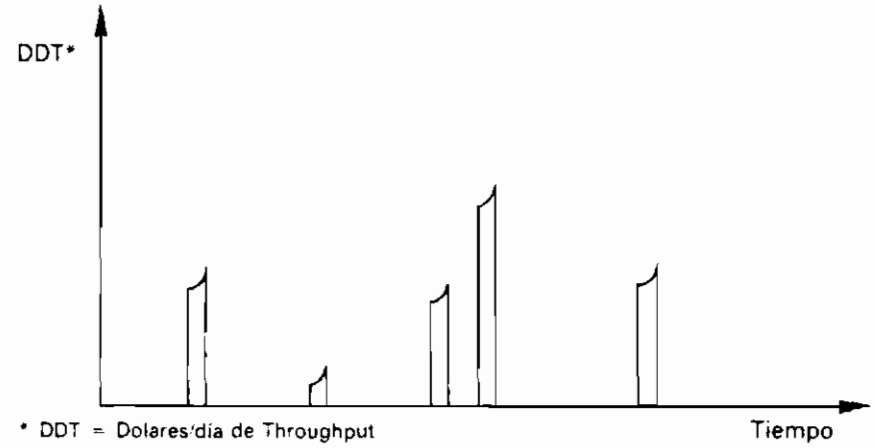


Fig. 24.2. *Los dólares/día de Throughput de otro departamento como función del tiempo.*

El tercer departamento es, sin duda, la fuente de las desviaciones. La figura 24.3 muestra claramente el crecimiento de los dólares/día, desde cero hasta que el trabajo sale por fin de ese departamento. El promedio de dólares/día es el más alto de los tres departamentos, tal y como debe ser. Ahora que hemos examinado estos tres modelos, ¿todavía piensan que es

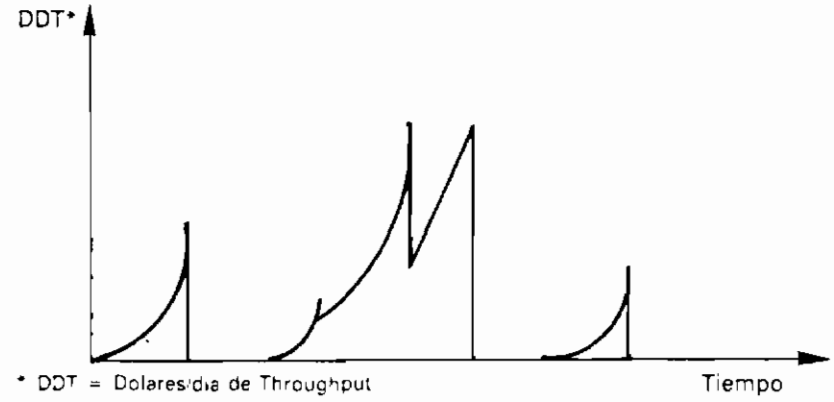


Fig. 24.3. *Los dólares/día de Throughput de los departamento más como función del tiempo.*

injusta la atribución de los dólares/día al departamento donde esté el trabajo? Sin duda, se habrán dado cuenta de que esta medida es perfectamente aplicable a cualquier departamento de nuestra organización, sea éste ingeniería, envíos o facturación. La pregunta clave es siempre la misma: en este momento, ¿quién está retrasando el progreso del pedido? Por ejemplo, no es difícil imaginar que bajo esta medida un departamento de ingeniería no se relajará mientras esté pendiente de entregar a producción un solo plano entre los cientos necesarios para fabricar algún sistema complejo.

Pero empieza a insinuarse otra ocurrencia perturbadora. Este tipo de medida puede fomentar el trabajo mal hecho. Bajo la presión creciente de los dólares/día, ¿no puede elegir un departamento el camino más fácil entregando a los siguientes departamentos un trabajo que no esté bien hecho, entregando el trabajo a medias para pasar la bola, para pasar la «culpa» a los hombros de otro? Si éste es el caso, todo lo que hemos ganado fomentando la autoaceleración se convertirá en una situación que no deseamos.

Vamos a examinarlo tranquilamente. Supongamos que un departamento ha entregado un trabajo de baja calidad. Es inevitable que esta baja calidad se revele algún día, ya sea por una de las siguientes operaciones o por un cliente muy insatisfecho. Si éste es el caso, asignemos los dólares/día correspondientes al departamento de control de calidad. Sí, al departamento de control de calidad.

Espero que a estas alturas ya estén usando extensivamente en sus empresas el control estadístico de procesos (SPC). Aunque esta técnica es muy útil para determinar cuándo es mala la calidad de una pieza, su mayor fuerza reside en su capacidad de determinar la causa de fondo de la baja calidad, de señalar el proceso y, ciertamente, el departamento que produjo el defecto. En cuanto se hace esto, los dólares/día se vuelven a asignar al departamento que causó el problema de calidad.

No es de esperar que este departamento se quede encantado con este fantasma del pasado. Recordemos que, a estas alturas, probablemente, estamos hablando de un pedido muy retrasado (especialmente en el caso de una devolución del cliente); el número de días es considerable y también lo es la penalización en dólares/día resultante. No hay duda de que cualquier departamento preferirá comprobar la calidad muchas veces antes de que el trabajo traspase los límites del departamento. Cualquier problema de calidad que se detecte en las operaciones siguientes causará al departamento descuidado unos dolores de cabeza mucho más grandes. Asombroso. Nos temíamos que esta medida nos llevaría a la mala calidad, y resulta que nos lleva, directamente, al modo de trabajar tan querido por TQM: *calidad en el origen*.

¿Qué pasa con el departamento de control de calidad? ¿Por qué tienen que sufrir mientras tanto? Dejemos de usar esta terminología tan negativa: culpa, sufrir... Lo que estamos intentando hacer es enviar las señales adecuadas para que las personas sepan en qué se tienen que concentrar para ayudar a la totalidad de la empresa. Vamos a coger el caso que acabamos de mencionar, el departamento de control de calidad, ¿cuál es de verdad su función principal? ¿Declarar que unas piezas son defectuosas? ¿Eso es todo? O, aún peor, ¿retener las piezas intentando averiguar si deberían ser desechadas o no, y mientras nadie sabe si deberíamos lanzar inmediatamente una reposición, o si las piezas retenidas por control de calidad resultarán estar bien al final?

Su verdadero trabajo es localizar el origen del problema de calidad, para que los recursos correspondientes puedan actuar eliminando el problema de una vez por todas. Parece que la asignación al departamento de control de calidad de esos dólares/día que resultaron de los problemas de calidad no sólo los lleva a hacer su trabajo verdaderamente importante, sino que, además, les proporciona la necesaria lista de Pareto.

Miren hasta dónde hemos llegado. Es evidente que no hemos terminado de explorar todas las repercusiones del uso de esta idea tan interesante, los *dólares/día de ingresos netos*. Todavía tenemos que abordar el segundo tipo de desviaciones, las que llevan a la inflación del inventario, y, también, necesitamos abordar la medida local que se relaciona con el tercer elemento, el gasto operativo. Estas desviaciones también se tienen que controlar. ¿Qué vamos a hacer? ¡Este capítulo ya es el más largo de todo el libro!

Parece como si estuviéramos cayendo en la trampa de la que hemos intentado avisar: perder de vista lo que queremos conseguir. No olvidemos que estamos intentando idear la composición y estructura de un *sistema de información*. A este paso no vamos a llegar nunca, volvamos a nuestras directrices: limitarnos a la estructura conceptual del sistema de información y, cuando sea necesario, abrir una nueva caja de Pandora, no dejarnos absorber totalmente en esos nuevos e interesantes temas. En nuestro contexto sólo debemos proporcionar las directrices, no un análisis completo.

Así que, para el que quiera profundizar más en el tema de las medidas del rendimiento local, sólo podemos decir que hay un poco más escrito sobre el tema en el *Theory of Constraints Journal*, volumen 1, número 3.

25

Un sistema de información debe estar compuesto por módulos de programación, control y simulación

Como hemos estado tratando de tantas materias, puede que sea hora de ver dónde estamos. Decidimos cortar por lo sano en la confusión existente entre datos e información, definiendo lo que nosotros entendíamos por esas palabras. Para nosotros, un dato es «cualquier conjunto de caracteres que describa algo sobre la realidad». Decidimos que la información era «la respuesta a lo que se ha preguntado». Llamamos «datos requeridos» a la parte de los datos que es necesaria para deducir la información requerida. A partir de estas definiciones, nos encontramos con situaciones en las que la información no está inmediatamente disponible, sino que tiene que deducirse de los datos requeridos. Estas situaciones nos obligaron a darnos cuenta de que el proceso de deducción no es algo externo al sistema de información, y que, para muchos tipos de información, el proceso de decisión en sí debe ser parte integral del sistema de información.

Debido a esto, y reconociendo los sistemas que hay disponbibles actualmente, decidimos llamar «sistemas de datos» a los sistemas que proporcionan información inmediatamente disponible, reservando el nombre de «sistemas de información» para aquellos que proporcionan información que no se puede obtener sino es a través de un proceso de decisión.

Al examinar diversas preguntas tipicas de dirección no pudimos dejar de darnos cuenta de que, a veces, la información tiene una estructura jerárquica, donde lo que para un nivel es un dato requerido, puede ser información para otro nivel. Es más, nos encontramos con preguntas muy importantes para las que no había disponibilidad inmediata de los datos, se tenían que deducir usando un proceso de decisión. Esos casos nos hicieron comprender que un sistema de información que sea completo deberá construirse sobre una estructura jerárquica.

En los sistemas de información industriales está muy claro que, en la

cumbre de la pirámide informativa, sabemos contestar a preguntas de dirección que tratan principalmente de elevar limitaciones o de prevenir la creación innecesaria de nuevas limitaciones. A este nivel pertenecen las preguntas relacionadas con justificación de inversiones, decisiones de comprar/fabricar, cuestiones de compras y, por supuesto, dilemas entre el diseño del producto y las ventas/marketing. Decidimos llamar a esta parte superior del sistema de información la fase de *simulación* (¿qué pasaría si...?

Resultó que, para poder siquiera empezar a contestar a esas preguntas, primero nos teníamos que dedicar a generar los datos requeridos, datos que, en sí mismos, no están inmediatamente disponibles, y que son la respuesta de otros dos tipos de preguntas de la dirección.

El elemento más fundamental resulta ser la identificación de las limitaciones actuales del sistema. Nuestro análisis nos mostró claramente que, para identificar las limitaciones actuales del sistema (por no mencionar la necesidad de identificar las limitaciones resultantes que se derivan de la alternativa que se está evaluando), no teníamos más remedio que resolver el viejo problema de cómo programar las operaciones de una empresa. Así pues, la fase más básica de un sistema de información es la fase de *programación*.

Decidimos dejar para el final la discusión detallada de cómo programar fiablemente las operaciones de una empresa, y nos sumergimos en la aclaración del otro tipo de dato requerido, el tipo de dato que gira alrededor de la cuantificación de las perturbaciones que se acumulan en la organización. Como este tema es un territorio relativamente virgen, tuvimos que emplear bastante tiempo en aclarar la terminología básica adecuada. Durante este proceso nos fuimos convenciendo, cada vez más, de que esta fase del sistema de información, de hecho, está tratando con preguntas de la dirección que giran alrededor del tema de *control*.

La fase de *control* tiene la capacidad de cuantificar a Murphy, lo que es esencial para entender la magnitud de la relación que existe entre el inventario y la capacidad de protección y, por tanto, para ser capaces de contestar fiablemente a cualquier pregunta del tipo *¿qué pasaría si...?* Es más, este mismo mecanismo es necesario para proporcionar otras dos clases de información necesaria. Una es la respuesta a la pregunta de dónde concentrar nuestros esfuerzos para reducir Murphy: dónde enfocar nuestros esfuerzos de mejora de procesos. La segunda es el tema, enormemente necesario, de las medidas del rendimiento local.

Hasta ahora hemos establecido que el sistema de información debe estar compuesto por tres fases o bloques: los bloques de *simulación*, de *programación* y de *control*. También hemos establecido que el bloque de *simulación* no se puede hacer antes de que los otros dos sean ya operativos,

ya que la información que proporcionan estos dos bloques son los datos que necesita el bloque de *simulación*. ¿Cuál es la relación entre los bloques de *programación* y *control*? ¿Son completamente independientes, o uno es prerrequisito del otro?

Es suficiente un examen superficial para revelar que el bloque de *control* no se puede usar hasta que esté implantado el bloque de *programación*. ¿Qué dijimos que significaba control? Saber dónde están las cosas con respecto a *dónde deberían estar*. Debemos controlar las desviaciones con respecto a un plan predeterminado, por tanto, la programación —la planificación— debe estar establecida antes de que podamos empezar con el *control*. Pero, vamos a examinar conceptualmente, con más detalle, lo que estamos intentando controlar, para que podamos exponer los requerimientos que deberá cumplir el bloque de *programación*.

Las desviaciones que influyen en los ingresos netos y las desviaciones que influyen en el inventario son desviaciones de planes realistas. Esto supone que nuestro plan —programa— debe tener en cuenta la existencia de Murphy, si no no podrá ser un plan realista. Es obligatorio que el bloque de *programación* proporcione un plan de acuerdo con algún cálculo predeterminado del nivel de Murphy que existe en la planta. Sólo así podrá proporcionar programas realistas el bloque de *programación*, y el bloque de *control* tendrá sentido en su esfuerzo por reducir el impacto de Murphy.

Por tanto, debemos dar a la fase de *programación* unos cálculos aproximados de los *buffers* de tiempo y de los niveles requeridos de capacidad de protección, cálculos que después se afinarán en la fase de *control*. El intento de usar la fase de *simulación* antes de que se hayan determinado unos niveles fiables de capacidad de protección con el uso de las otras dos fases es, desde mi punto de vista, una pérdida de tiempo.

Ya tenemos el plan de acción. La primera fase que debemos construir del plan de acción es la fase de *programación*. El único dato necesario, además de los que hoy en día se usan normalmente, es un cálculo aproximado del impacto del Murphy existente. Una vez que esta fase sea operativa se puede poner en práctica la siguiente, la fase de *control*. Y solamente después de haberlas usado durante algún tiempo podremos alcanzar el objetivo último del sistema de información: la capacidad de contestar a nuestras preguntas de *simulación*.

Ahora necesitamos empezar a tratar de la estructura de la primera fase —la *programación*—, pero, no como antes. Acabamos de llegar a los cimientos y, por tanto, ya no podemos dejar temas abiertos, quedándonos satisfechos con sólo mostrar la dirección conceptual. De ahora en adelante cada detalle debe quedar muy bien establecido.

TERCERA PARTE
Programación

26

Acelerando el proceso

Así que, ahora, nos encontramos con la poco envidiable tarea de describir la aproximación a la fase de programación. ¿Será fácil, sólo una aburrida tarea técnica? Probablemente ninguna de las dos cosas. No olvidemos que el objetivo principal de MRP era programar; y, a pesar de todo, después de treinta años de monstruosos esfuerzos, ¿dónde hemos llegado? Todos los expertos coinciden en que, a pesar de tanto esfuerzo, MRP no es un programador, sino un banco de datos muy necesario.

Esto no es de extrañar, porque los que originaron el MRP no disponían del proceso de decisión que se basa en los ingresos netos. Como ya hemos indicado antes, ésa es la razón por la que casi todos los esfuerzos adicionales de los últimos treinta años se han dirigido hacia el infructuoso intento de ampliar la disponibilidad de los datos. Al menos, ahora nos podremos beneficiar de esos esfuerzos; al desarrollar la fase de programación no nos tendremos que preocupar de la disponibilidad de la mayoría de los datos básicos; todo el mundo los tiene ya capturados de una forma u otra. De hecho, normalmente se tienen en más de una forma, no es raro encontrar empresas que tienen tres registros distintos de la lista de materiales para, exactamente, el mismo producto.

Aun así, parece que no nos podemos librar de volver a examinar la forma en que los ordenadores tratan actualmente los datos. Verán, nuestra experiencia con el MRP nos ha enseñado otra lección bastante desagradable. El tiempo necesario para generar un programa, incluso uno erróneo y con poco detalle, es bastante exorbitante. En plantas pequeñas estamos hablando normalmente de horas de ordenador y, en organizaciones grandes y complicadas, hay veces que no es suficiente con un fin de semana. ¿Es esto una preocupación real, un ingrediente esencial con el que tenemos que tratar ahora? ¿O, sólo estamos cayendo en la trampa de intentar no enfrentarnos a la necesidad de desarrollar una solución «suficientemente

buena» a base de refinar prematuramente una solución que todavía no se ha descrito?

En otras palabras, el tiempo necesario para generar un programa, ¿es un factor decisivo que nos impedirá totalmente el uso del sistema de información, o sólo es una molestia? A primera vista, parece que la situación requiere que se realice un programa nada más. El que queramos tenerlo tan pronto como sea posible no importa mucho, siempre que, de verdad, lleguemos a tenerlo. Pero ésta no puede ser la respuesta general.

Es fácil demostrarlo. Imaginemos que el tiempo necesario para generar un programa de una semana es más de una semana. Si ésta es la situación, el método, ciertamente, no es práctico. El tiempo de generación no es una trivialidad, es un ingrediente esencial que se debe considerar mientras se desarrolla la solución. Tiene que haber un límite máximo para el tiempo de generación del programa, un límite que, si se supera, convierte al método definitivamente en inútil.

Bueno y, ¿por qué estamos perdiendo el tiempo en cosas que son obvias? Porque puede que nuestra impaciencia provenga de la falsa premisa de que la mayoría de las soluciones factibles para los problemas de programación entrarán dentro de los límites tolerables sin ningún esfuerzo adicional. ¿De verdad es éste el caso? Parece que lo es, sin duda. Nuestra experiencia en programación ya nos da un cálculo aproximado de ese límite superior. Sabemos que podemos tolerar un tiempo de generación del programa de muchas horas, incluso de un fin de semana completo... Entonces, ¿por qué tenemos que preocuparnos de limitar el tiempo de generación? Un momento, no tan deprisa. Nuestra experiencia procede de una realidad en la que considerábamos la programación como un fin en sí misma, no como un paso necesario para otra cosa. ¿Es posible que cuando evaluemos la programación como una fase de un proceso más grande, el proceso de simulación, nuestra intuición nos pida que reduzcamos considerablemente el límite superior? ¿Reducirlo hasta el extremo de que no sean aceptables las muchas horas necesarias para generar un programa detallado de varias semanas?

Para aclarar la situación intenten imaginar que ustedes, como directores, están intentando decidir entre varias alternativas. No olvidemos que la mayoría de las cuestiones de dirección no sólo tienen influencia en el futuro inmediato, sino, también, a medio plazo. Examinen, por ejemplo, la lista de preguntas que aparece al principio del capítulo 17. ¿Cuál es el horizonte de tiempo que implican? Varía desde algunos meses hasta varios años. Como ya hemos dicho, si queremos evaluar fiablemente cada una de las alternativas, tendremos que descubrir el impacto que tienen en las limitaciones del sistema y, por lo tanto, debemos tener un programa que abarque todo el horizonte de tiempo correspondiente, un programa detallado.

¿Cuánto tardaremos si usamos el método tradicional? Seguro que muchas horas, quizás días. Si tienen que emplear tanto tiempo en evaluar siquiera una de las muchas alternativas, ¿de verdad creen que usarían el sistema de información? La conclusión inevitable es que debemos exigir que la fase de programación sea capaz de realizar una programación detallada de un plazo largo para toda la empresa en menos de una o dos horas. Es una exigencia tremenda, pero, probablemente, es inevitable.

Para reducir drásticamente el tiempo necesario para programar, parece que no tenemos más remedio que volver a examinar cada paso que consuma tiempo; incluso la forma en que actualmente tratan los datos los sistemas MRP convencionales. Tenemos que examinarlo críticamente, para encontrar dónde se consume la mayoría del tiempo que tardan. Una vez más, cuando nos encontramos con un problema que existe desde hace mucho, no tenemos más remedio que profundizar hasta las raíces escondidas del problema.

¿Por qué tarda tanto una ejecución de MRP? Los ordenadores tienen fama de ser extremadamente rápidos. Como sabemos algo sobre ordenadores, sabemos que tienen dos velocidades diferentes: la velocidad de cálculo y la velocidad de archivo y recuperación de datos. Ambas velocidades son notables, pero son drásticamente distintas.

La velocidad de cálculo del ordenador, usando los que llamamos la unidad central de proceso (CPU) y su memoria *on-line*, es extraordinaria. Hasta los ordenadores personales pueden multiplicar dos números grandes en menos de una millonésima de segundo. Pero la velocidad del ordenador es muy distinta cuando recupera o escribe un bloque de datos en su soporte de almacenamiento de datos: los discos. Llamamos a este modo de operación el modo de entrada/salida (*Input/Output*, I/O). Aun considerando los discos más rápidos de uso más común que hay en el mercado, estamos hablando de tiempos mayores que una centésima de segundo. Una velocidad notable, pero es como el paso de un caracol si lo comparamos con la velocidad de cálculo.

De la forma en que se han escrito los paquetes de programación disponibles actualmente destaca, sin duda, el hecho de que, la mayoría del tiempo, el ordenador está ocupado recuperando y escribiendo datos, desde y hacia los discos. Cualquier profesional les dirá, sin ninguna duda, que MRP es «totalmente I/O». Que quiere decir, en palabras más simples, que la gran mayoría del tiempo no se consume haciendo cálculos, sino moviendo datos internamente. Hasta el pequeño porcentaje de tiempo que el ordenador declara como tiempo de la CPU resulta ser, si lo examinamos bien, el tiempo de la CPU que tiene que invertir el ordenador para tratar los datos. Sólo se dedica a los cálculos una fracción de un porcentaje del

tiempo total que tenemos que esperar para que el ordenador genere el programa.

¿Tiene que ser así, o podemos hacer algo para mejorar la situación existente? Resulta que la mentalidad que usamos para codificar el ordenador todavía tiene una fuerte influencia de las limitaciones técnicas que había en el pasado —limitaciones que hoy en día están totalmente superadas, incluso en el más pequeño y corriente de los ordenadores personales (PCs).

Verán, hace menos de quince años, un programador no tenía un ordenador potente a sus órdenes, un ordenador con más de un millón de *bytes* para el tratamiento de datos *on-line* (memoria *on-line*). Todos los programadores con experiencia, y, ciertamente, todos los libros de texto sobre programación, se basan en la experiencia conseguida en un entorno donde tenían que luchar contra la rigidez de no tener suficiente memoria disponible para sus programas.

Incluso cuando llegaron los grandes ordenadores centrales, trayendo con ellos memorias *on-line* de millones de *bytes*, los programadores sabían que si su programa necesitaba una gran parte de esa memoria, tendría que esperar en la cola durante muchas horas. Recordemos que esos ordenadores centrales los utilizaban muchos usuarios, en contraste drástico con nuestro mundo de ordenadores personales. Con esas restricciones de memoria, no quedaba más remedio que almacenar y después recuperar los valores intermedios (especialmente si recordamos que el tiempo promedio entre fallos era de unas pocas horas).

La inercia que generaron las restricciones pasadas ha enmascarado la gran diferencia que hay entre las preferencias de una persona y de un ordenador. Supongamos que tienen ustedes dos listas de 50 elementos cada una. En una lista están los precios por unidad y, en la otra, las cantidades que se van a comprar de cada elemento. Les gustaría saber la cantidad total de dinero que tendrán que pagar. Uno de los caminos posibles es multiplicar cada número de la primera lista con su correspondiente de la segunda, y sumar los cincuenta resultados. O, simplemente, podemos pasar la página y leer el resultado, está escrito allí. ¿Qué preferirían hacer? Sin duda, pasar la página. Pero el ordenador no.

Para el ordenador «pasar la página» supone acceder al disco y traer a la memoria un conjunto de datos. En esta operación tardará algunas centésimas de segundo. Si hace las 50 multiplicaciones y, después, suma los resultados, tardará algunas millonésimas de segundo. El ordenador prefiere hacer el cálculo completo mil veces antes que tener que recuperar del disco el resultado.

En general, los paquetes que se encuentran en el mercado no utilizan esta enorme capacidad de cálculo. En vez de eso, van y vienen al disco

para recuperar datos que podrían recalcularse internamente muchísimo más deprisa. Por supuesto, este último método requiere mantener más datos en la memoria *on-line*, pero, como ya hemos dicho, las limitaciones, en cuanto a memoria disponible, se han reducido drásticamente. Si prestamos atención a esta diferencia entre la velocidad de cálculo y la velocidad de recuperación, podemos reducir en más de mil veces el tiempo total de ejecución del ordenador.

Esto es básicamente lo que tenemos que hacer aquí, pero, aunque disponemos de mucha memoria, si la desperdiciamos puede llegar a convertirse en una auténtica limitación. Por lo tanto, deberíamos examinar la forma en que vamos a almacenar y tratar los datos, que es lo que nos va a dictar los tiempos globales de ejecución, y, por lo tanto, la viabilidad de nuestro sistema de información.

27

Reduciendo algo más la inercia: reordenación de la estructura de los datos

El proceso de explosión es la parte más tediosa de la programación, y la que más tiempo consume. Explosión quiere decir empezar en el nivel de los pedidos e ir profundizando por la estructura del producto para determinar las necesidades, tanto en cantidad como en tiempo, en los niveles inferiores: ensamblaje, producción de componentes y compras de material.

Hoy en día la estructura del producto normalmente no está contenida en una sola entidad, en vez de eso está dividida en dos categorías separadas, la *lista de materiales* (*bill of materials*, BOM) y las rutas. Esta separación obliga a ir y venir entre dos segmentos distintos de la base de datos, lo que, naturalmente, hace que aumente drásticamente el tiempo de respuesta del ordenador. Pero pensándolo bien, tanto el BOM como las rutas son descripciones del «viaje» que tiene que realizar el material a través de las operaciones para convertirse en algo que satisfaga las necesidades del cliente. Así pues, ¿por qué hace falta esta segmentación tan incómoda?

Es asombroso contemplar la respuesta de los diseñadores de sistemas cuando se les enfrenta a la pregunta ¿por qué se segmentó la estructura de producto en dos ficheros, el BOM y las rutas? Normalmente escucharemos, o bien una avalancha de jerga técnica que hará que nos dé vueltas la cabeza, o alguna extraña argumentación filosófica, casi metafísica. Con lo que, de hecho, nos estamos encontrando es con nuestra vieja, y no muy buena amiga, la inercia. Sí, había una razón muy práctica, casi obligatoria, para esta segmentación. Había una razón, pero ya ha desaparecido.

Vamos a hacer un poco de excavación arqueológica para revelar la verdadera razón. Empecemos desde el origen, a principios de los sesenta, cuando se hizo el principal esfuerzo para construir el diseño conceptual de MRP. En aquel tiempo sólo se disponía de cintas magnéticas para almace-

nar cantidades grandes de datos. En las cintas sólo se pueden almacenar cantidades grandes de datos. En las cintas sólo se pueden almacenar y recuperar los datos de forma secuencial; esta limitación técnica enfrentó a los diseñadores de MRP con un enorme problema.

Examinemos la situación a la que se enfrentaban cuando la estructura del producto contenía un subensamblaje complejo, necesario para distintos productos finales. Veamos, por ejemplo, la figura 27.1.

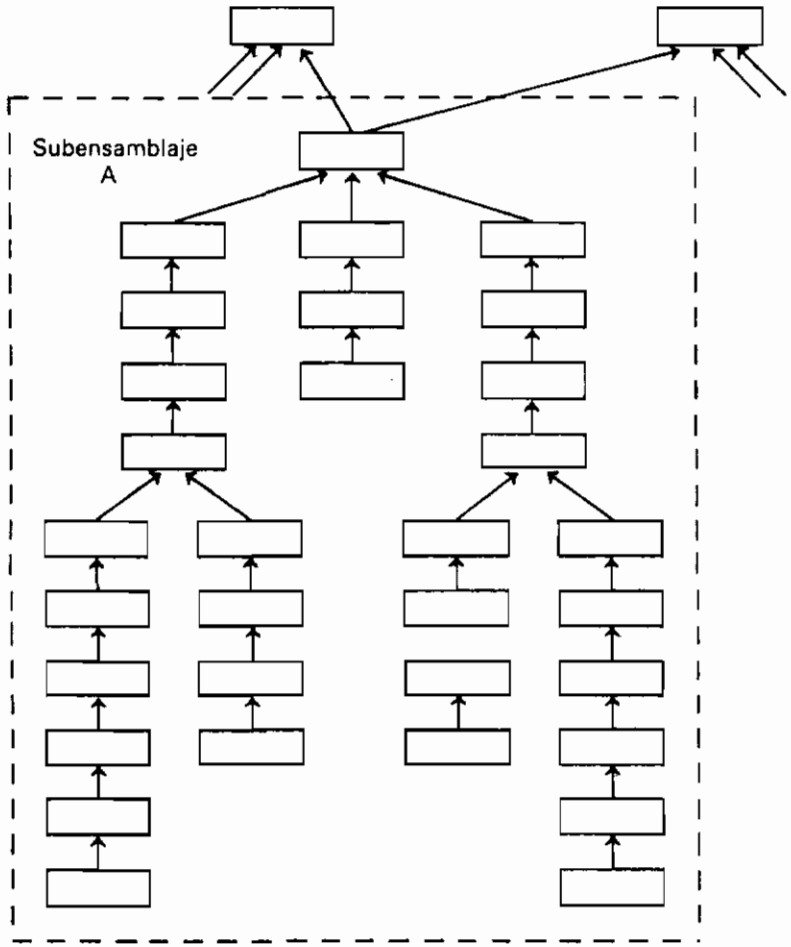


Fig. 27.1. *Un subensamblaje mayor A, requerido para dos productos diferentes.*

Una descripción así, aunque es natural, es imposible cuando el medio de almacenamiento es a base de cintas. Lo más aproximado es detallar el

subensamblaje dentro del primer producto, y sólo mencionarlo en los otros productos (como se muestra en la figura 27.2).

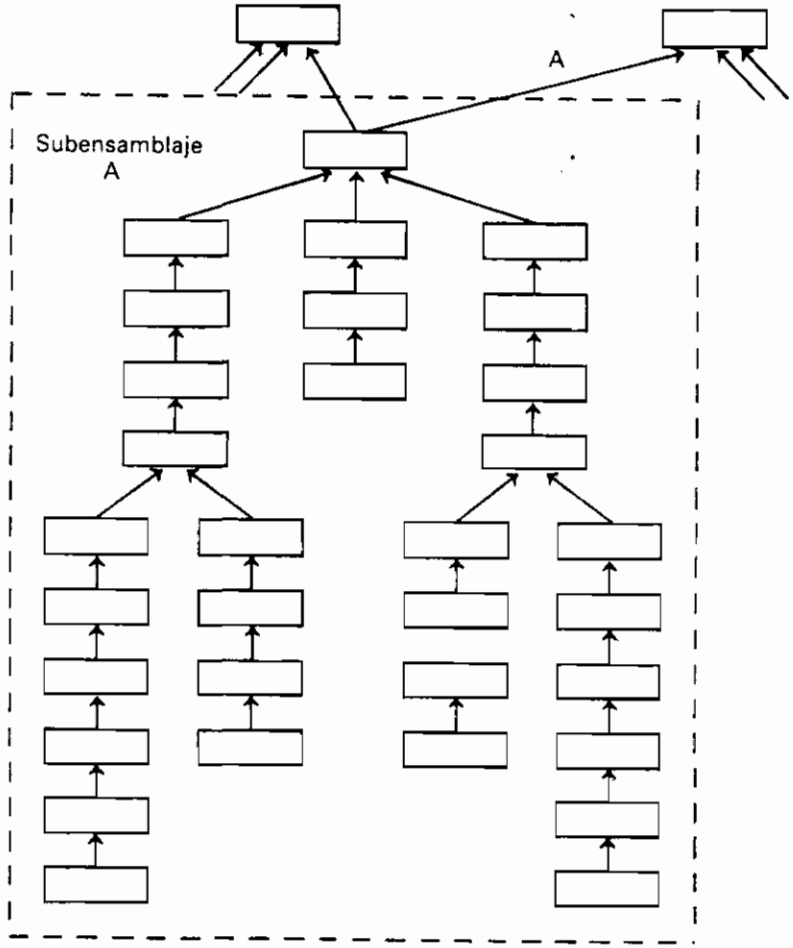


Fig. 27.2. *Los detalles de un subensamblaje mayor se presentan completos sólo en un producto y nada más se indican en los demás.*

Pero, ¿qué habría pasado si hubieran elegido esa descripción? Cada vez que hiciera falta *explosionar* uno de los otros productos tendríamos que rebobinar la cinta para conseguir los detalles del ensamblaje. ¿Saben cuánto se tarda en rebobinar una cinta? Ya no son fracciones de segundo, son minutos, aparte de que harían falta tantos rebobinados que, probablemente, la cinta se rompería antes de terminar la ejecución.

La otra alternativa que tenían, también mala, era detallar los datos del submensaje en todos y cada uno de los productos que lo necesitaran (como se muestra en la figura 27.3).

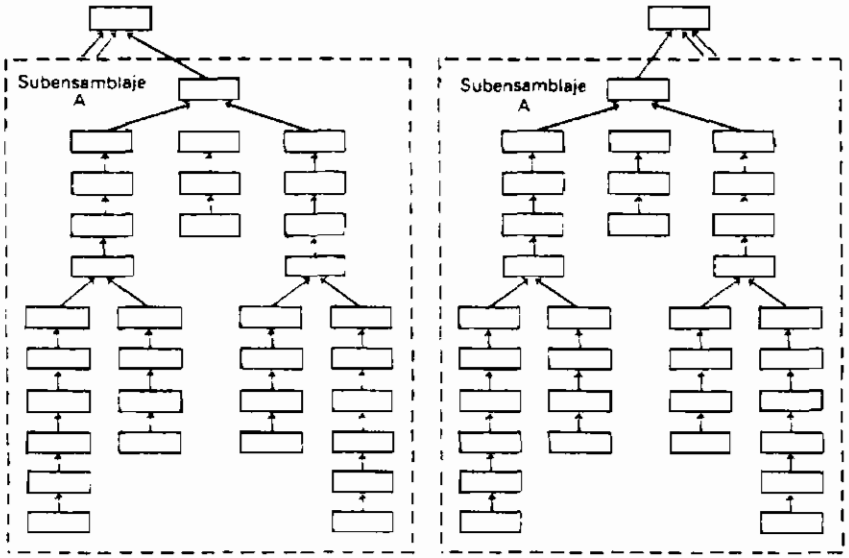


Fig. 27.3. *Un ensamblaje mayor A, se presenta completo en cada producto que lo requiere.*

El problema no reside en la cantidad de datos que es necesario almacenar para hacer la *explosión* de necesidades. El problema se revela después del esfuerzo inicial, cuando se llega a la fase de mantenimiento de los datos. Se hacen cambios en este subensamblaje. Se actualiza en la mayoría de los sitios, pero, no en todos: siempre se olvida alguno. No pasa mucho tiempo hasta que las discrepancias llegan a un nivel en el que el sistema ya no es utilizable.

Encontrándose ante la devastadora elección entre una alternativa horrible y otra aún peor, los diseñadores de MRP decidieron llegar a un compromiso. Crearon los conceptos de BOM y de rutas. De hecho, lo que hicieron fue repetir en todas partes sólo los principales detalles estructurales del subensamblaje —lo que hoy en día llamamos BOM— y almacenar sólo una vez la gran mayoría de los datos detallados —lo que llamamos rutas. El resultado final se muestra en la figura 27.4. ¿Era una solución perfecta? Ni de lejos, pero funcionaba.

Desde entonces, los discos han sustituido a las cintas como medio de almacenamiento. Los discos permiten el acceso directo en vez del secuencial; moviendo la cabeza del disco, como si fuera un disco fonográfico, podemos alcanzar directamente cualquier punto de su superficie. Los discos eliminaron la limitación técnica que originó el problema, pero ya era demasiado tarde. BOM y rutas ya habían enraizado hasta el punto de adquirir vida propia. Nadie se atrevió a cuestionar esta sagrada segmentación.

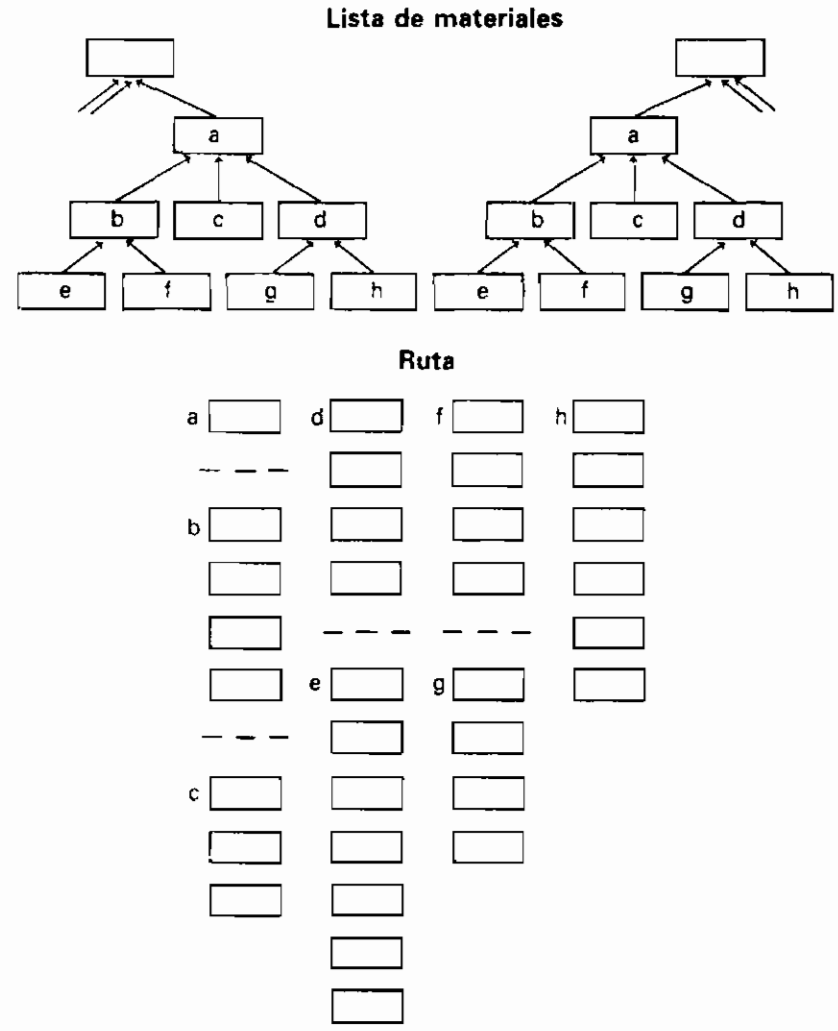


Fig. 27.4. *La solución de compromiso de la lista de materiales y las rutas.*

Si nos damos cuenta de la capacidad actual de los ordenadores, deberíamos usar una estructura de producto que no diferencie entre BOM y

rutas, representando cada entrada una fase del «viaje». Básicamente hemos vuelto al cuadro más fundamental, el que se muestra en la figura 27.1. Cada paso del «viaje» del material es equivalente a un código de pieza/código de operación. Un cambio trivial.

A una persona que no haya pasado muchos años en el entorno de MRP ni siquiera le parecerá un cambio, le parecerá una elección natural. Pero el impacto en el tiempo de ejecución del ordenador es bastante profundo. Esta fusión entre el BOM y las rutas convencionales reduce drásticamente el número de veces que necesitamos acceder al disco, y, por lo tanto, acelera todo el proceso en varios órdenes de magnitud. ¡Qué precio tan alto hemos pagado por culpa de nuestra inercia!

Pero no nos detengamos aquí, vamos a fundir más ficheros de datos en nuestra estructura del producto. Hoy en día la mayoría de los sistemas mantienen separados otros dos ficheros. Uno se suele llamar «inventario de almacén» y el otro «inventario de trabajo en proceso (*work-in-process:* WIP)». El inventario de almacén básicamente detalla la cantidad disponible de piezas terminadas, subensamblajes y materiales, mientras que el WIP contiene el inventario de piezas parcialmente procesadas. ¿Cuál es el motivo de esta separación que, una vez más, dilata el tiempo de la explosión?

Cuando consideramos desde el punto de vista de diseño del sistema algunas piezas que están procesadas parcialmente, nos encontramos con la necesidad de codificarlas, de asignar un código a estas unidades. De hecho, tenemos dos alternativas para este caso. Podemos codificar el inventario según el código de la última operación por la que han pasado las unidades, o podemos codificarlo según el código de la siguiente operación que deben pasar. Ambas opciones son equivalentes. No es de extrañar que a principios de los sesenta los diseñadores del sistema decidieran elegir de acuerdo con la costumbre de las fábricas. ¿Dónde situamos los elementos que están a medio procesar? En la cola que hay delante de la siguiente máquina. En otras palabras, asignamos físicamente el inventario WIP de acuerdo con la siguiente operación por la que debe pasar. Era una elección natural.

Esta elección, tan natural para piezas semiprocesadas, es totalmente inadecuada cuando estamos tratando con piezas acabadas. Aquí no tenemos la opción de codificar el inventario de acuerdo con la siguiente operación, la siguiente operación es el ensamblaje. Si le asignamos el código del ensamblaje damos la falsa impresión de que también están disponibles los demás componentes que hacen falta para ese ensamblaje.

Para las piezas acabadas, la única alternativa posible es codificarlas de acuerdo con la última operación que se les ha hecho. Esta diferencia hace necesaria la separación de los dos ficheros de inventario, una separación que no molestó demasiado a los diseñadores, ya que reflejaba la separación que ya existía entre los ficheros BOM y de rutas. Nosotros tendremos que ser más coherentes, ya que hemos fundido los dos ficheros de la estructura del producto. Siempre nos referiremos al inventario según la última operación por la que haya pasado. Esto no sólo nos permitirá fundir en uno los dos ficheros de inventario, sino que, además, permitirá la convergencia de los datos de inventario en un solo campo, que residirá en el fichero de la estructura del producto.

Esta reducción del número de ficheros abre la posibilidad de mantener en la memoria todos los datos necesarios. Esto sólo se podrá conseguir si no inflamos los datos necesarios para programar con otros detalles que no tienen nada que ver, como la dirección de un proveedor o el ángulo al que se debe cortar el material. Igualmente, debemos tener la precaución de no almacenar demasiados resultados de cálculo intermedios. Si lo hacemos, se «comerá» rápidamente la memoria disponible, y nos veremos obligados a recurrir a los discos. Recordemos que repetir un cálculo largo es mucho más rápido que ir a buscarlo al disco.

La capacidad de mantener en la memoria todos los datos necesarios permitirá que el ordenador sólo tenga que acceder al disco al principio de la ejecución de la programación —para copiar los datos en la memoria— y, al final —para escribir la programación. La disponibilidad de memoria que existe hoy en día, que alcanza los *megabytes* incluso en un PC, nos permite operar de esta forma tan deseable. No es de extrañar que el resultado sea un recorte drástico del tiempo de ejecución. La programación detallada de una fábrica grande, para un horizonte de muchos meses o, incluso, de años, puede conseguirse ahora, en los potentes PC actuales, en bastante menos de una hora.

Por supuesto que el problema de la conversión empieza a levantar su fea cabeza. ¿Cómo vamos a pasar de la estructura multiarchivo actual a la «red de estructura-tarea» uniforme que se sugiere? Este problema parece aún más complicado cuando recordamos que en nuestros ficheros actuales hay metidos muchos más datos, muchos más de los estrictamente necesarios para la simulación de las acciones futuras de la empresa. Esos datos adicionales ¿no van a convertir el plan que hemos sugerido en otra pesadilla distinta, pero no menos complicada?

Puede que sí, puede que no. Pero la pregunta que de verdad nos debemos hacer es si tenemos o no que convertir toda nuestra base de datos. ¿De verdad que nos tenemos que preocupar de esto? Aquí estamos tratando de construir un sistema de información, no estamos intentando mejorar nuestros sistemas actuales de datos. En cualquier caso, ya hemos llegado a la conclusión de que un sistema de información no sustituye a un sistema de datos; es mejor que nuestro sistema de información absorba los datos que necesite de un sistema de datos.

Es más, ya hemos llegado al acuerdo de que el propósito principal de un sistema de información es tratar con situaciones hipotéticas, con preguntas de simulación. Al mismo tiempo, nos damos perfecta cuenta de que a los directivos de distintos niveles y distintas funciones les preocupan distintas preguntas de simulación. Por lo tanto, cada directivo tendrá que ajustar la base de datos para crear las situaciones «hipotéticas» únicas que desean examinar. La conclusión es casi inevitable: un sistema de información y, por lo tanto, sus propios datos específicos requeridos, debe tener una distribución muy amplia y accesible. ¿Debemos tratar de la misma forma a nuestros sistemas de datos? En mi opinión, esto sería una receta para el desastre. Si mantenemos nuestros sistemas de datos en una red descentralizada tenemos la garantía de que, con el tiempo, las discrepancias en los datos empezarán a surgir como setas. Intenten imaginar el caos que se crearía si distintos directivos, que se supone que están todos trabajando con la misma base de datos, empiezan a tomar decisiones que, de hecho, están basadas en datos distintos. No, el acalorado debate actual entre los oponentes de procesos centralizados y descentralizados procede de la confusión que existe entre sistemas de datos y sistemas de información. Un sistema de datos debe estar centralizado. Los sistemas de información, alimentados. ¿No es obvio?

Este descubrimiento es también una buena directriz para la necesidad de conversión del formato de los datos. No hace falta reestructurar los bancos de datos que existen. En lo que tenemos que concentrarnos es en la estructura del subconjunto de datos que debe estar disponible para el sistema de información. Como casi todas las empresas almacenan sus datos en un formato y con una distribución de ficheros distinta, no tiene sentido intentar desarrollar un paquete de conversión genérico. En vez de eso, deberíamos intentar reducir, al mínimo absoluto, la parte del trabajo de programación que debe adaptarse a los formatos actuales de los sistemas de datos existentes, asegurándonos de que este trabajo de programación sea lo bastante fácil como para que lo pueda hacer cualquier programador. La parte más sofisticada, la parte que se ocupa de fundir los ficheros esporádicos en una red uniforme, puede, entonces, estandarizarse con un paquete.

Aunque pueda parecer, a primera vista, que encontrar la segmentación adecuada es un problema difícil, en realidad es una simpleza. Sólo con ignorar, momentáneamente, nuestro ardiente deseo de complicar las cosas, de hacerlas más «sofisticadas», la solución resulta obvia. Todo programador sabe cómo extraer de una base de datos un fichero secuencial que sólo contenga, en cada registro, un subconjunto de campos predeterminados. De hecho, esto es una utilidad estándar en la mayoría de los sistemas de datos. Hagámoslo separadamente con cada uno de los ficheros: el fichero

de la lista de materiales (BOM), el de rutas, el de inventario, el de trabajo en proceso, el de disponibilidad de recursos, el de clientes y el de previsión de consumo externo. Ya estamos en una estructura específica de la base de datos existente, y, por lo tanto, podemos continuar con la fase más complicada de la conversión utilizando una interfaz estándar.

¿Qué más necesitamos? El calendario de la organización, o varios calendarios si hay departamentos que usan calendarios distintos, como es el caso de una división que tenga sectores en distintos países. ¿Qué más? Oh, algunos parámetros como la longitud de los diversos *buffers* de tiempo y una estimación del nivel necesario de capacidad de protección.

Pero dejemos de entretenernos; es obvio que ya hemos dejado atrás los problemas del tiempo de ejecución y de la conversión de datos, al menos, a nivel conceptual. Ya es hora de que dirijamos nuestra atención al verdadero problema: ¿cómo deberíamos programar, de acuerdo con lo que hemos descubierto sobre el mundo del valor?

28

Establecimiento de los criterios para un programa aceptable

¡Programad las tareas que debe realizar una organización! Es fácil de hacer, pero ¿por dónde empezamos? ¿Debemos empezar con los pedidos de los clientes y *explosionarlos* hasta el nivel de las tareas? Si lo hacemos así, ¿qué pedido debemos coger primero? ¿El de mayor facturación? ¿El que representa una carga mayor? ¿El que ya estamos sospechando que vamos a entregar tarde? O puede que debamos intentar un ataque completamente distinto, como podría ser empezar con los trabajos que están atascados ahora mismo: limpiar el sistema no parece mala idea. Pero hay una vocecita que nos susurra desde el fondo que, ya que hemos discutido tanto el problema de la capacidad limitada, lo mejor sería empezar por secuenciar las tareas de uno de los recursos más cargados, y continuar a partir de ahí. Hay tantas posibilidades... Ninguna de ellas parece muy prometedora, especialmente cuando descubrimos que cada una de ellas se ha intentado ya más de una vez, y que han llevado a programas que ciertamente no eran nada extraordinarios.

Bueno, en algún sitio tendremos que empezar. Pero, antes de luchar con la pregunta de «dónde empezar», es mejor que aclaremos lo que de verdad queremos conseguir. ¡Un programa! ¿No está claro? No necesariamente. ¿Qué tipo de programa queremos construir? ¿Qué programa nos resultará satisfactorio?

Empezamos a darnos cuenta de que el primer paso para desarrollar un programa es definir los criterios que debe cumplir un «buen» programa. El primer criterio es muy obvio: el programa debe ser realista. Pero, un momento... ¿Qué queremos decir con la palabra realista? No nos conformemos con usar palabras de significado demasiado amplio, palabras que no nos facilitan un entendimiento seguro de lo que en realidad debemos, o no debemos, hacer. ¿Qué puede hacer que un programa se considere como no realista?

Si reflexionamos sobre esta pregunta, aparecen dos cosas distintas que pueden hacer que consideremos un programa como no realista. La primera es generar un programa imposible de llevar a cabo por nuestro sistema. Esto puede suceder si el programa ignora las limitaciones inherentes a nuestro sistema. ¿Qué nombre pusimos a lo que limita el rendimiento de nuestro sistema? Limitaciones. Así pues, un programa realista deberá empezar por reconocer las limitaciones del sistema. No es una conclusión abrumadora, ya hemos afirmado más de una vez que el primer paso (después de definir la meta y las medidas) siempre es *identificar las limitaciones del sistema.*

¿La identificación de las limitaciones es suficiente para garantizar un programa realista? No necesariamente, podemos encontrarnos con alguna situación en la que surge un conflicto entre limitaciones. Por ejemplo, una empresa promete entregas que sobrepasan su capacidad de cumplir las fechas prometidas. ¿En qué momento hay que resolver estos conflictos? ¿Vamos a permitir que sea la realidad la que los arregle? Esta forma de funcionar nos llevará, sin duda alguna, hacia sorpresas desagradables. Es mejor que examinemos cuidadosamente esos conflictos futuros, y que los resolvamos antes de que la realidad se encargue de resolverlos por defecto. Es evidente que un programa realista no debería albergar ningún conflicto entre las limitaciones del sistema.

Esta última afirmación arroja algo de luz sobre el proceso con el que se debería construir un programa. Ya nos hemos dado cuenta de que la identificación de las limitacioes y, en consecuencia, la creación de un programa, tendrá que seguir un proceso de iteración, repitiendo una y otra vez los pasos de *identificar, explotar* y *subordinar*, encontrando cada vez una limitación adicional. Ya hemos dicho que este proceso tendrá que continuar hasta que no se encuentren violaciones al final del paso *subordinar*. Ahora nos damos cuenta de que siempre que se encuentre una limitación adicional, tendremos que comprobar concienzudamente si existen conflictos entre las limitaciones que se han identificado.

Es más, como esos conflictos no se habían detectado antes (si no, ya se habrían eliminado), es bastante posible que los datos necesarios para resolver los conflictos no se hayan especificado claramente, que estos datos existan sobre todo en la parte más intuitiva del saber hacer. Por ejemplo, si la única forma de resolver un conflicto es atrasar pedidos de clientes, el conocimiento de qué pedido es el que podemos retrasar sin generar un conflicto demasiado grande con el cliente no suele estar escrito en ninguna parte ¿Deberíamos exigir, como diseñadores del sistema, que este tipo de dato necesario se registre completa y explícitamente? Tal y como yo lo veo, sería una exigencia incumplible. Si esta exigencia resulta ser obligatoria, más vale que hagamos las maletas y dejemos el tema.

¿Qué más queda por hacer cuando se revela un conflicto entre limitaciones? La respuesta debe ser que el sistema de información debe concentrarse en revelar los conflictos, destacando en cada caso el mínimo óptimo de acciones que deben tomarse para resolver el conflicto; pero, a no ser que se hayan establecido unas directrices muy precisas, el sistema de información deberá parar en ese punto y pedir al usuario que tome la decisión.

Esta conclusión es de un gran alcance. Comparémosla con los otros métodos de programación que se han usado en las últimas décadas. Por ejemplo, con la programación lineal, una técnica que se atreve a presentar al usuario un resultado final en el que se han «resuelto» los conflictos, ignorando el hecho de que no se han considerado muchos de los datos necesarios para la resolución adecuada de esos conflictos. Es más, ni siquiera se molesta en destacar dónde se han encontrado los conflictos ni qué supuestos se han considerado para su resolución. Para el usuario es sólo una caja negra que, de forma arrogante, pone un conjunto de ecuaciones matemáticas por encima de la superior capacidad de la mente humana para encontrar soluciones amigables con un simple cambio de los supuestos del problema, cuando se encuentra con una situación sin salida. No es de extrañar que la programación lineal (o su hermana mayor y más sofisticada, la programación dinámica) no haya conseguido encontrar una aplicación general, a pesar del hecho de que, sin razón aparente, ha sido la piedra angular de la investigación operativa durante los últimos veinte años.

Comparemos nuestra conclusión con el método de programación de «Just-in-time», el sistema *Kanban*, un método que decidió descargar todo el tratamiento de los conflictos de la fase de programación a la fase de ejecución: casi la antítesis de la programación lineal. No se han establecido directrices coherentes para guiarnos en la decisión sobre el número y contenido de las diversas tarjetas que deben situarse entre los centros de trabajo: todo el peso de la ejecución del programa recae sobre el personal de planta.

Si miramos el método MRP, vemos que ha desistido, a priori, de todo intento de ser realista, como se puede entender claramente gracias a su lema, *capacidad infinita*. Sí, MRP intenta corregir la situación introduciendo el concepto de «bucle cerrado», un concepto que tiene en cuenta la capacidad finita de los recursos de la empresa. Pero hasta un simple vistazo superficial nos revela que el bucle cerrado es un proceso iterativo que *nunca converge*. En nombre de los procedimientos realistas, MRP ha presentado un procedimiento no realista.

Es bastante evidente que la pretensión de conseguir programas realistas no es trivial en absoluto. Si nuestro sistema de información tiene que ser efectivo contestando a preguntas de gestión del tipo «que pasaría si...», nos

tenemos que tomar mucho más en serio el requerimiento de que los productos sean realistas.

Antes de hacer una pausa para seguir avanzando, debemos recordar que teníamos dos requerimientos para que un programa fuera realista. Uno era la resolución de conflictos entre limitaciones. ¿Cuál era el segundo? ¡Ah, sí! El segundo requerimiento que debe cumplir un programa es que sea inmune a las perturbaciones. Para nosotros este requerimiento no es una novedad, ya que lo hemos tratado bastante extensamente en los capítulos anteriores, pero no como parte del programa en sí. Hemos tratado de este tema como parte del control. ¿El concepto de inmunidad tiene que ver con la programación? No podemos dejar de darnos cuenta de que, sin duda, es así.

Un programa, por definición, se refiere a un intervalo de tiempo. Si el propósito del programa es la identificación de limitaciones futuras, el programa deberá ser absolutamente realista para ese futuro intervalo de tiempo. Si cualquier perturbación, cualquier acción de Murphy, provoca una reprogramación, es la mejor indicación de que el programa que se ha generado no es, en absoluto, realista para el futuro intervalo de tiempo requerido. Parece que la inmunidad es un requerimiento obligatorio para un programa.

¿Los métodos antiguos han tenido en cuenta la inmunidad? No del todo. La programación lineal no dudaba en presentar «soluciones» que se basaban en limitaciones interactivas. Esto se hacía a pesar del hecho de que todos los análisis de sensibilidad (que son parte integral de la programación lineal) indicaban, claramente, en los numerosos casos de soluciones a limitaciones interactivas, que los programas resultantes eran inestables, y que cualquier perturbación invalidaría totalmente los programas que no sugerían. No, a pesar de que teníamos todas las indicaciones, preferimos ignorar la inmunidad como requerimiento válido.

¿Y qué hay de JIT y del MRP? Estos métodos eran mucho más prácticos, pero no lo suficiente. Ambos métodos se daban cuenta de que la inmunidad contra las perturbaciones era esencial para un programa, tanto que han llegado al extremo de asignar más tiempo a la inmunización que a la realización de las tareas en sí. A pesar de ello, los programas que resultan de ambos métodos están lejos de ser inmunes, lo que resulta evidente cuando vemos que la responsabilidad de resolver las «sorpresas» recae sobre los hombros del personal de planta. El problema principal lo provoca el hecho de que ambos métodos han intentado inmunizar el programa en sí, en vez de inmunizar el resultado del programa. Este enfoque los ha obligado a intentar proteger cada instrucción individual, un método que se destruye a sí mismo. El resultado inevitable de querer proteger cada operación es una sobreextensión del *lead-time* necesario para

cumplir con los requerimientos del cliente. Para vencer este obstáculo los «tiempos disponibles» individuales (ya fuera el número de tarjetas *Kanban* o los tiempos de colas y esperas) tuvieron que reducirse tanto que, una vez más, nos encontramos con programas inestables.

Lo que se necesita es inmunidad en el programa hasta el punto de que las predicciones del programa sean realistas: la predicción sobre el funcionamiento que se espera de la organización, y la predicción sobre las limitaciones de la organización.

¿Es eso todo? ¿Ya hemos terminado de definir los criterios por los que deberíamos juzgar un programa? Todavía no hemos mencionado el más importante. ¿Para qué necesitamos un programa? No nos olvidemos de lo que estamos tratando. Cuando originamos un programa nuestro objetivo es, como siempre, intentar alcanzar la meta. Así pues, el programa, una vez que sea realista, debe juzgarse con las mismas medidas que usamos para juzgar los resultados: valor generado, inventario y gasto operativo.

Debemos tener presente la escala de importancia de estas medidas. Cuando nos encontremos en una situación en la que se pueda perjudicar el valor generado, una situación que podríamos corregir aumentando el inventario o el gasto operativo, debemos tomar las medidas de corrección adecuadas sin titubear. Pero aquí es necesaria una advertencia. Siempre que aumentemos el inventario o el gasto operativo debemos tener mucho cuidado de no chocar contra las condiciones necesarias.

Por ejemplo, si para proteger el valor generado debemos aumentar el inventario de material (lanzando el material antes de lo que es normalmente necesario), esto debe hacerse a través del sistema de información. Pero si necesitamos aumentar capacidad para proteger el valor generado, ya sea comprando una nueva máquina o autorizando algunas horas extras, el sistema de información deberá ser mucho más cuidadoso, la compra de una nueva máquina podría violar la condición necesaria de disponibilidad de caja. Es más, faltan muchos de los datos necesarios para tomar una decisión así, como el tiempo de entrega de la máquina nueva o las nuevas capacidades que se ofrecen hoy en día para esa tecnología. De la misma manera, cuando se necesitan más horas extras de las que ya se han autorizado, el sistema de información no debe autorizar las horas extras adicionales. No debemos ignorar el hecho de que muchos de los datos necesarios para tomar una decisión así (como la voluntad de hacer más horas por parte de los implicados, o su efectividad cuando las horas extras se usan en demasía) no están disponibles para el sistema. Ni siquiera debemos pretender que el sistema disponga de estos datos. En la mayoría de los casos es un volumen enorme, y normalmente existen sólo a nivel intuitivo. No, en estos casos nuestro sistema de información debe destacar la situación, dejando que la decisión específica la tome el usuario.

En resumen, un sistema de información debe generar un programa que sea, ante todo, realista, que no contenga ningún conflicto entre las limitaciones de la organización, y que sea inmune a un nivel razonable de perturbación.

Los programas realistas resultantes se juzgan con las medidas habituales. En otras palabras, el rendimiento final que indica el programa se juzga con respecto a su consecución del máximo valor generado (máximo en términos de explotación de las limitaciones de la empresa). Lo segundo en importancia es el nivel de inventario de material que se va a necesitar en función del tiempo. Sólo debe haber inventario de material en cantidad suficiente para garantizar el valor generado; lo demás se debe considerar como un exceso de inventario, y el programa se deberá juzgar de acuerdo con esto.

En lo que se refiere al gasto operativo, sólo se permite que el sistema de información asigne horas extras dentro de los parámetros que se le hayan especificado, y en esos casos la única razón válida para emplear horas extras es la protección del valor generado por la empresa. Cualquier otro aumento del inventario (como máquinas o dispositivos), o de los gastos operativos (como horas extras especiales o personal adicional) deben ser dictados por el usuario y, por tanto, aunque deben ser parte de las medidas usadas para juzgar el programa final, no deberían usarse para juzgar el sistema de información en sí mismo.

29

Identificación de las primeras limitaciones

Ahora resulta obvio dónde debemos empezar a trabajar para generar un programa: identificación de las limitaciones de la empresa. ¿Qué podría ser una limitación? Recordemos que hemos acordado que nuestro sistema de información no iba a tener en cuenta las limitaciones políticas. Las limitaciones políticas no deben explotarse, deben elevarse. Lo que quedan son las limitaciones físicas: limitaciones de mercado, de recursos y de proveedores. Esto todavía es demasiado amplio como para usarlo de punto de partida, vamos a intentar centrarlo más. ¿Cómo?

¿Nos limitamos a declarar que algo es una limitación, para tener un punto de arranque definitivo? Esto podría funcionar si se cumplen dos condiciones. La primera es que, al final, tengamos una buena indicación de si nuestra elección fue buena o mala. Recordemos que estamos haciendo el programa, sobre todo, para identificar las limitaciones. La segunda condición, no menos importante, es que tengamos garantía de que el programa será aceptable, aunque resulte que nos equivocamos en nuestra elección y que lo que elegimos no era una limitación.

Desgraciadamente, si consideramos todo lo que hemos discutido hasta ahora con respecto a la realización de un programa, resulta evidente que no se cumplen las dos condiciones anteriores. Parece que el único mecanismo que puede señalarnos claramente que la limitación que escogimos no es una limitación es la «gestión del *buffer*». Pero, si nos basamos en esta técnica, sólo sabremos que nuestra suposición era errónea cuando hayamos observado, en la realidad, durante un cierto tiempo, la ejecución del programa dudoso. Es demasiado tarde. No, debemos empezar por identificar algo que sea, sin duda, una limitación.

¿Debemos identificar todas las limitaciones antes de que podamos pasar al paso siguiente para intentar explotarlas? No hace falta. Como ya hemos dicho, la identificación de las limitaciones es un proceso reiterativo;

tendremos que recorrer los pasos de *identificar, explotar* y *subordinar* una y otra vez, añadiendo, cada vez, una nueva limitación, hasta que lleguemos al punto en el que al final de la subordinación no se encuentren violaciones de la realidad. Sólo entonces habremos terminado.

Esto implica que siempre que tengamos dudas sobre si algo es o no una limitación es mucho más seguro suponer que no lo es. Si no es una limitación, y nosotros declaramos que lo es, vamos a tener que cargar con ella. Pero, si la verdad es una limitación, y en esta fase decidimos ignorarla, no habremos hecho ningún daño, porque llegaremos a cogerla en una de las vueltas siguientes. Así que la cuestión es: ¿qué podemos declarar como limitación desde el principio y con un alto grado de probabilidad?

Aparentemente, empezar eligiendo un proveedor o, incluso, un material específico, es demasiado arriesgado. Sólo podemos identificar a un material como limitación cuando comparemos su disponibilidad actual y futura con los plazos y cantidades del consumo requerido. Esto implica que disponemos, desde el principio, de un conocimiento detallado de lo que, de hecho, estamos intentando generar con este proceso: el programa. No, la decisión de empezar con la limitación en el proveedor no es una buena elección, sobre todo cuando recordamos muchos casos en los que no había limitación en los proveedores.

¿Podemos empezar con una limitación en los recursos? Podría ser, pero recordemos que la mayoría de las limitaciones de recursos con las que nos encontramos en la realidad no son cuellos de botella, sino recursos que no tienen suficiente capacidad de protección. La «marca» de un recurso con capacidad de protección insuficiente es que ese recurso, aunque tiene suficiente capacidad media, no es capaz de absorber sus picos de carga. Así pues, sólo lo podremos identificar analizando con detalle los requerimientos en función del tiempo. Volvemos a encontrarnos con una situación en la que para identificar la limitación necesitamos tener el programa. No, no podemos construir un procedimiento general basado en la poco realista suposición de que en toda situación hay, al menos, un recurso que no tiene capacidad suficiente para satisfacer la demanda del mercado.

Bueno, las vueltas que hemos dado nos han llevado a algo: la única opción que nos queda es empezar desde las limitaciones del mercado: los pedidos de los clientes. Pero antes de lanzarnos es mejor que comprobemos si es correcto suponer, con una probabilidad alta, que los pedidos de los clientes siempre se pueden tomar como limitaciones. El mero hecho de decir que no queda otra alternativa equivale a suponer que hemos incluido en nuestro análisis todas las posibilidades. Esta suposición, además de ser arrogante, es muy peligrosa.

Así que vamos a ignorar el hecho de que ya hemos descalificado todas las otras posibilidades que conocemos, y nos vamos a concentrar en el

examen de si podemos o no suponer con seguridad que los pedidos de los clientes siempre son una limitación de nuestra organización. (Por cierto, esto subrayará el hecho de que debemos cambiar de actitud hacia las limitaciones; si los pedidos de los clientes son limitaciones, entonces, las limitaciones no son necesariamente algo malo.)

Si no hay limitaciones internas en nuestra organización, entonces, la limitación está en la demanda del mercado (ignoremos por el momento la posibilidad de la limitación en los proveedores). Si internamente tenemos exceso de todo, lo único que nos limita para ganar más dinero es la demanda del mercado. ¿Qué pasa con los casos en los que sí tenemos limitaciones internas de capacidad? ¿No es arriesgado suponer que en estos casos el mercado sigue siendo la limitación? Recordemos que hemos supuesto que no hay limitaciones políticas en lo que respecta a nuestro sistema de información. Vamos a distinguir entre dos casos, cuando de hecho tenemos cuellos de botella, y cuando sólo tenemos limitaciones de capacidad debidas a una insuficiente capacidad de protección. Este último caso es más fácil de tratar, así que empezaremos por él.

La cantidad de capacidad necesaria de protección que tiene que tener un recurso está en función de la longitud del *buffer* de tiempo. Si no se especifica un plazo de entrega (implícita o explícitamente), la longitud del *buffer* de tiempo no tiene límite y, por lo tanto, no hace falta ninguna capacidad de protección. Así pues, siempre que tratemos con un caso en el que exista un recurso limitado debido a una falta de capacidad de protección, estaremos, por definición, tratando un caso en el que la limitación principal será la demanda del mercado.

Lo que nos queda por examinar es el caso del cuello de botella, un recurso que no tiene suficiente capacidad disponible como para satisfacer estrictamente la demanda (como el siguiente análisis del caso del cuello de botella es idéntico al análisis requerido para el caso de una limitación en el proveedor o en un material específico, éste no se detallará separadamente). Nuestro pequeño caso de «Q» y «P» nos enseñó que, cuando un cuello de botella participa en la realización de más de un producto, la demanda del mercado sigue siendo la limitación del sistema para todos los productos, excepto uno. Es más, a diferencia del caso, normalmente tenemos fechas de entrega; tenemos que entregar a nuestro cliente no más tarde de una fecha determinada. En un caso así, el pedido específico será una limitación.

Resulta que el único caso en el que no podemos considerar al mercado como limitación es cuando no tenemos que comprometernos con fechas de entrega a clientes. Tenemos un recurso limitado fabricando un solo producto, y cada unidad que ofrecemos la absorbe el mercado inmediatamente. Reconozcamos que es un caso muy raro, así que, si tenemos pedidos

con sus correspondientes fechas de entrega, no es arriesgado suponer que las limitaciones son los pedidos de los clientes.

Por fin sabemos, sin ninguna duda, dónde debemos empezar con nuestros esfuerzos de programación. ¿Y ahora qué? Deberíamos examinar cómo explotar las limitaciones. Muy complicado. Si las limitaciones son los pedidos de los clientes, explotar la limitación significa, simplemente, cumplir las fechas de entrega requeridas; el paso de explotación se consigue por defecto. Ahora podemos continuar con el siguiente paso de subordinar todo lo demás a los pedidos de los clientes, lo que no resultará tan fácil. Tendremos que emplear un tiempo y un esfuerzo mental considerables para determinar precisamente cómo hacerlo.

Mi problema es que al final de la subordinación cualquier conflicto en los datos indicará la existencia de una limitación adicional. Tendremos que localizarla, en cuyo caso tenemos garantizado que el paso de subordinación lo será todo menos trivial. El seguir este camino representa un riesgo pedagógico; dedicar primero mucho tiempo a la subordinación, y después a la explotación, puede grabar en nuestra memoria una secuencia distorsionada. Así pues, y sólo para que yo me quede tranquilo, vamos a seguir buscando limitaciones reales antes de pasar al siguiente paso de explotar lo que ya hemos encontrado. Esto nos obligará a dedicarnos, primero, al paso de explotación, para pasar, después, a la subordinación.

¿Podemos encontrar en esta fase inicial, aunque sólo sea a nivel teórico, alguna otra limitación? Sí, a veces nos podemos encontrar con una situación en la que, de hecho, existe un cuello de botella. Recordemos que podríamos simplemente continuar, ya que, sin duda, lo encontraremos al final de la primera vuelta cuando la subordinación nos indique que hay conflictos en los datos. Pero vamos a intentar tratarlo ahora mismo. El resultado final va a ser el mismo, pero esto ayudará a clarificar el proceso.

¿Cómo podemos identificar un cuello de botella en esta fase inicial? Para eso tendremos que proporcionar al sistema otro parámetro. Un cuello de botella es un recurso que no tiene suficiente capacidad productiva. ¿En qué plazo de tiempo? Si no se especifica el plazo de tiempo, un plazo ilimitado con una demanda de mercado finita implica que todos los recursos tienen suficiente capacidad. Así pues, cuando la demanda del mercado se concrete en un conjunto de pedidos dados, sólo tendrá sentido hablar de cuellos de botella cuando también se especifique el plazo de tiempo.

¿Debemos elegir como plazo de tiempo el intervalo que va desde hoy hasta la fecha de entrega más remota de los pedidos que tenemos? Esto casi nos garantiza que, en la vida real, acabaremos con las manos vacías en nuestro intento de encontrar un cuello de botella. No olvidemos que, normalmente, nuestros conocimientos sobre el futuro se degradan a medida que miramos más y más lejos: la cantidad de pedidos que tenemos en

firme no engloba todos los pedidos que recibiremos en el futuro. Normalmente, cuando examinamos los pedidos en firme, tenderemos a ver una gran concentración de pedidos para el futuro inmediato, una concentración que se va diluyendo cuando cambiamos nuestro enfoque hacia horizontes de tiempo más remotos. Por supuesto que lo que se considera futuro inmediato y horizonte de tiempo más remoto depende, en gran medida, del tipo de industria con el que estemos tratando: en las industria de armamento un año se considera como un futuro muy próximo, mientras que, en un taller pequeño, incluso un mes puede llegar a verse como demasiado remoto. Así pues, si insistimos en encontrar un recurso limitado antes de intentar explotar y subordinarnos a las limitaciones del mercado, el único camino practicable que nos queda es pedir al usuario que especifique una fecha de corte, una fecha a la que vamos a denominar *horizonte de programación*.

Para comprobar si tenemos o no un cuello de botella, primero tenemos que calcular la carga total que se ha puesto sobre cada uno de los tipos de recursos, la carga que han generado los pedidos en los que se debe trabajar durante el horizonte de programación. ¿Cuáles son esos pedidos? Desde luego, aquellos cuya fecha de entrega sea anterior a la fecha del horizonte de programación. Pero esto no es suficiente. Consideremos, por ejemplo, un pedido que se debe entregar un día después de la fecha del horizonte de programación: ¿de verdad queremos decir que todo el trabajo necesario para satisfacer ese pedido se debe hacer sólo el último día? Vemos que incluso un pedido que se tiene que entregar después del horizonte de programación puede colocar cargas dentro del horizonte. ¿Cuánto nos tendremos que adentrar en el futuro? ¿Cuáles son los criterios? Desgraciadamente, parece que el nuevo parámetro —«horizonte de programación»— no nos ha ayudado, después de todo.

¿Para qué necesitamos este quebradero de cabeza? Sabemos que si tenemos recursos limitados, acabaremos encontrándolos antes o después gracias a nuestro proceso de reiteración. ¿Por qué no nos quedamos con la limitación de mercado ya identificada y continuamos a partir de ahí? Es más, estamos empezando a tener la muy errónea impresión de que sólo se deben tener en cuenta los pedidos en firme. No olvidemos que la mayoría de las preguntas de la dirección ¿qué pasaría si...? deben tener un horizonte más largo y, por lo tanto, nuestro análisis también se debe basar en las «previsiones de ventas». Así que, cuando hablemos de pedidos, siempre queremos decir pedidos en firme más previsiones. Si repasamos esta última página, ¿no sería mucho mejor descartar todo este fútil ejercicio pedagógico? Parece que sólo nos lleva a callejones sin salida. ¡Un momento! Un pequeño obstáculo no debería causar tanta agitación. Puede que un minuto más de reflexión sea suficiente para enseñarnos claramente el camino.

Entonces, ¿qué pedidos con fecha de entrega posterior al horizonte de programación debemos considerar al calcular las cargas? Aquellos en los que estemos seguros de que el trabajo necesario para realizarlos debe hacerse dentro del horizonte. Bien, pero, ¿cómo los identificamos? De hecho, no es muy difícil, sólo tenemos que recordar lo que dijimos hace algunos capítulos. Toda limitación debe protegerse, y la protegemos con tiempo. Una limitación de mercado debe estar protegida con un «*buffer* de envío». ¿Recuerdan todos estos detalles? Dijimos que el «material» se debe lanzar un intervalo de tiempo (que llamamos «*buffer* de envío») antes de la fecha de entrega del pedido. También dijimos que el tiempo real de trabajo es despreciable con respecto al tiempo total transcurrido, consumido principalmente por Murphy. Así pues, cualquier pedido cuya fecha de entrega sea anterior al horizonte de programación más el *buffer* de envío, situará cargas en los recursos de la empresa dentro del horizonte.

Ya ha quedado claro que debemos considerar todos los pedidos cuyas fechas de entrega sean anteriores al horizonte de programación más el *buffer* de envío. También está claro que debemos deducir todos los trabajos que se hayan hecho ya en esos pedidos. Esto implica que el mecanismo para calcular la carga relevante deberá *explosionar* desde los pedidos hacia las estructuras de los productos correspondientes, teniendo en cuenta todos los inventarios que existan, tanto de producto acabado como de piezas en proceso.

¿Y los tiempos de cambio de modelo? ¿No hace falta tenerlos en cuenta? Sí, pero en esta fase inicial debemos tener el cuidado de coger cada cambio una sola vez. Vamos a aclarar este tema un poco más detalladamente. Como *explosionamos* desde cada pedido, es posible que nos encontremos más de una vez exactamente en la misma operación y en la misma pieza. Puede que distintos pedidos tengan fechas distintas, pero este hecho todavía no implica que vayamos a producir por separado los componentes de cada pedido. A efectos de ahorrar tiempos de cambio, podemos decidir combinar muchos lotes en uno solo, satisfaciendo muchos pedidos de fecha diferente, pero activando el recurso una sola vez para esa pieza/operación específica. El número de cambios en el que vamos a incurrir, y, por lo tanto, el tamaño de los lotes, es función de la propia construcción del programa y, por ello, no se puede predeterminar.

Lo máximo que podemos hacer en esta fase inicial es considerar un solo cambio por pieza/operación, independientemente del número de pedidos que necesiten de ese trabajo en particular (suponiendo que haya, al menos, un pedido que requiera ese trabajo). De hecho, ni siquiera esta precaución me deja satisfecho. Todos sabemos que los tiempos de proceso no son muy exactos, pero las estimaciones de los tiempos de cambio son, en la mayoría de los casos, fantasías bienintencionadas. ¿Podemos hacer algo al respecto?

Como veremos muy pronto, en esta fase tendremos que elegir, en todo caso, un cuello de botella como máximo. Es preferible encontrarlo sin tener que depender de los datos sobre los tiempos de cambio. Esto quiere decir que sólo en último extremo consideraremos los cambios como parte de la identificación de la limitación.

Una vez calculada la carga para cada tipo de recurso, tendremos que calcular su disponibilidad durante el mismo intervalo de tiempo futuro (desde ahora hasta el horizonte de programación, sin el *buffer* de envío), de acuerdo con los calendarios dados. Por supuesto, cuando se calcule la disponibilidad, habrá que tener en cuenta el número de unidades de cada tipo de recurso que tiene la empresa. Si la carga que se asigna a un recurso es mayor que su disponibilidad, tenemos, al menos, un cuello de botella.

30

Cómo trabajar con datos muy inexactos

Supongamos que cuando comparamos las dos listas, la que recoge la carga por tipo de recurso con la que nos muestra la disponibilidad de recursos, nos encontramos con que todos tienen suficiente capacidad. No importa. No podremos llegar a ninguna conclusión con respecto a la existencia de recursos limitados, pero tampoco se habrá hecho ningún daño. Recordemos que como todos los datos requeridos están en la memoria, este trabajo de *explosión* de necesidades (que básicamente equivale a ejecutar los requerimientos de toda la red) se habrá hecho en un tiempo insignificante. Sin embargo, ¿qué hacemos si, no sólo uno, sino varios recursos resultan tener menos capacidad disponible de la que es necesaria para cumplir los pedidos?

En un caso así, ciertamente, tenemos algún recurso limitado, pero, ¿cuántos tenemos? ¿Debemos considerar como limitación a cada uno de esos recursos problemáticos? ¡Desde luego que no! En esa lista podemos encontrar que tenemos una limitación, pero solamente una. ¡Recordemos lo peligroso que es considerar como limitación un recurso que no lo es! Debemos tener mucho cuidado. Primero vamos a aclarar cómo es posible que un recurso pueda no ser una limitación, aunque el cálculo nos indique claramente que no tiene suficiente capacidad para cumplir con toda la demanda.

Supongamos que tenemos un caso elemental en el que sólo tenemos que entregar un producto. Los pedidos nos exigen entregar 100 unidades diarias durante los próximos diez días. El proceso de producción de este producto es muy simple, cada unidad pasa por diez operaciones distintas que se hacen en serie (una tras otra), y cada operación requiere un tipo distinto de recurso. En nuestra empresa sólo hay una unidad de cada tipo de recurso, y están disponibles veinticuatro horas al día. Supongamos que cada operación tarda en realizarse, al menos, una hora, y la más larga

tarda dos. Ahora vamos a calcular la carga que suponen esos pedidos para cada recurso, comparándola con su disponibilidad. En nuestro caso, la comparación nos indica claramente que todos y cada uno de los recursos no tiene suficiente capacidad.

¿Significa esto que todos son limitaciones, que todos los recursos están limitando la capacidad de nuestra empresa para ganar más dinero? En absoluto. Si no hay forma de que consigamos capacidad adicional, el recurso que dictará el resultado final será el recurso que necesita emplear dos horas en cada unidad. El resto de los recursos no tendrá influencia alguna sobre el resultado final. Si no podemos conseguir más capacidad para este recurso específico, podemos llegar a doblar la capacidad de los demás sin que cambie el resultado final. Así pues, podemos ver que puede que sólo haya una limitación, aunque tengamos más de un recurso sin capacidad suficiente para atender a la demanda del mercado.

Por tanto, de todos los recursos que no tienen suficiente capacidad, en este punto sólo podemos declarar como presunto recurso limitado al que tenga menos capacidad de todos. Hemos dicho presunto. ¿Por qué no lo podemos considerar definitivamente como limitación? Porque este análisis se basa en los datos disponibles, y sabemos que este tipo de datos puede ser enormemente inexacto.

¿En qué situación nos encontramos ahora? Hemos identificado una limitación en los pedidos, y tenemos razones para creer que, además de eso, tenemos un recurso limitado. Si resulta que estamos en lo cierto, que tenemos un recurso limitado y somos incapaces de satisfacer la limitación del mercado por falta de capacidad, quiere decir que nos encontramos ante un conflicto entre las limitaciones de la empresa. Cualquier solución a este conflicto dará como resultado una degradación de los resultados de la empresa. Antes de comprometer estos resultados, ¿no deberíamos, al menos, comprobar si la situación está basada en datos erróneos? Pero tenemos que reconocer que, después de intentar mantener unos registros exactos durante más de dos décadas, casi desesperadamente, hemos aprendido que es virtualmente imposible mantener todos los tiempos de proceso con exactitud. Entonces, ¿qué hacemos?

Recordemos que, aunque no es posible garantizar la exactitud de todos los tiempos de proceso, es bastante fácil verificar la validez de unos pocos.

¿Qué datos nos han llevado a sospechar la existencia de un recurso limitado en particular? Vamos a examinarlos, teniendo presente que distintos tipos de datos pueden tener distintas probabilidades de ser inexactos. Nuestra estimación se ha basado en el cálculo de la disponibilidad de cada tipo de recurso y en las cargas que se les asignan. A su vez, el cálculo de disponibilidad se basaba en el calendario y en el número de unidades disponibles de este tipo de recurso. Normalmente, el calendario es común

a toda la organización, o, al menos, a gran parte de ella, y, por tanto, suele estar más que comprobado. La probabilidad de encontrar un error aquí es muy pequeña.

Pero no es éste el caso cuando consideramos la exactitud del número de unidades del recurso. Por extraño que parezca, este tipo de dato suele estar sujeto a grandes errores. No es demasiado sorprendente cuando consideramos que el número de unidades del recurso no sirve para calcular los requerimientos netos ni para hacer cálculos de costes, así que hoy en día nadie se molesta en actualizar estos números. Es más, es bastante común encontrar en nuestra lista recursos que simplemente no existen. Son sólo fantasmas que se han introducido para que el personal de sistemas pueda burlar la rigidez de sus propios sistemas.

Por supuesto, si nos encontramos con errores así de grandes los corregimos y volvemos a examinar nuestra lista. Supongamos que ya hemos comprobado los datos de disponibilidad del presunto recurso limitado y hemos encontrado que son exactos. ¿Qué debemos comprobar ahora? Los datos que hemos usado para calcular las cargas. ¿Qué parte de esos datos nos preocupa más? Los tiempos de proceso, por supuesto. ¿Todos ellos? Ciertamente no, sólo los tiempos de proceso de ese recurso en particular. ¿Lo podemos reducir todavía más? Sí, el tiempo de proceso de los trabajos que hace este recurso en particular, para los que haya, al menos, un pedido dentro del plazo de tiempo que el usuario esté considerando. ¿Son todos igual de importantes? No, algunos trabajos absorben una gran cantidad de tiempo, mientras que otros sólo requieren algunas horas. ¿De qué depende? Del tiempo necesario para procesar una unidad y de las cantidades requeridas por los pedidos.

Esta línea de razonamiento parece muy obvia, casi infantil, ¿por qué nos molestamos en desarrollarla tanto? Porque, desgraciadamente, la situación actual es que, siempre que sospechamos que nuestros datos no están en buena forma y nos están llevando a conclusiones erróneas, lo normal es declarar que «hay que limpiar la base de datos», normalmente con la recomendación adicional: «por lo menos hasta un nivel del 95 por 100» y dejar el problema al usuario. Cualquier sistema que aborde de esta forma el problema de la exactitud de los datos está ignorando una de sus funciones más importantes: destacar, de forma clara y precisa, qué datos de toda la maraña son los que hay que comprobar, reduciéndolo hasta el punto de que resulte factible el trabajo de verificación de la exactitud de esos datos. Lo que estamos haciendo ahora es demostrar que esto lo puede hacer un sistema, siempre que los diseñadores de ese sistema den a este tema la importancia que merece.

Así que ¿dónde estamos ahora? Está bastante claro que el sistema debe mostrar al usuario un gráfico que detalle el porcentaje de disponibilidad

de nuestra presunta limitación que absorbe cada trabajo. Aquí estamos tratando con variables interconectadas débilmente, y, por tanto, es razonable esperar que muy pocos trabajos estén creando la mayoría de la carga. Los tiempos de proceso de ésos trabajos, y solamente esos, deben ser verificados por el usuario. Normalmente tendremos que comprobar unos cinco tiempos de proceso.

Recordemos que siempre es preferible comprobarlo con la gente que está realizando el trabajo, no con los que lo diseñaron. Los consultores expertos en MRP tienen la siguiente regla: «comprueba siempre con el encargado, no con el ingeniero. Es probable que el encargado te engañe, quizá en un 30 por 100. Pero sabrás exactamente en qué dirección te está engañando. El ingeniero te puede despistar, incluso hasta en un 200 por 100, y no tendrás la menor idea de en qué dirección».

Una vez comprobados los tiempos de proceso relevantes (y suponiendo que no se hayan encontrado errores) es el momento de verificar, sólo para esos «trabajos de carga alta», los pedidos correspondientes. No es raro encontrar en la cantidad del pedido algún error, un cero errante mal tecleado por algún administrativo aburrido.

¿Cuál es la forma recomendada para que el sistema de información destile esos datos? La tendencia normal es a almacenar todos estos datos a medida que el sistema lleva a cabo las explosiones necesarias para calcular las cargas. Es cierto que en esta fase se «visitan» todos los datos y se pueden registrar fácilmente. Pero esto es lo que no se debe hacer. Preguntémonos lo siguiente: si capturamos los datos en la fase de explosión, ¿cuántos datos habrá que almacenar, y qué mínima parte de ellos hará falta mostrar eventualmente?

He aquí un ejemplo muy bueno de cómo conseguir que el sistema responda a paso de caracol. Recordemos la enorme diferencia que hay entre la velocidad de archivo/recuperación y la velocidad de cálculo. Si almacenamos con cada recurso todos los trabajos relevantes, junto con sus tiempos de proceso, y, además, almacenamos, para cada uno de esos trabajos, todos los códigos de los pedidos correspondientes, entonces, no hay duda de que, incluso en una empresa relativamente pequeña, no tendremos suficiente memoria *on-line*. Dieciséis «megabytes» de memoria *on line* (el máximo actual disponible para un PC) no serán suficientes. Si intentamos almacenarlo todo en los discos, el tiempo consumido será demasiado largo.

La respuesta está en la dirección que ya hemos apuntado. Como todos los datos relevantes ya están en la memoria, no tiene sentido almacenar resultados intermedios. Simplemente se vuelven a calcular. En lo que respecta al presunto recurso limitado, vamos a subir por las estructuras de productos desde todos sus trabajos (código de pieza/operación que pasa por este recurso en particular) hacia el nivel de los pedidos. Este paso establecerá las conexiones del producto entre nuestro recurso limitado y las correspondientes limitaciones de mercado. Para hacerlo más simple, llamaremos «líneas rojas» a esas conexiones del producto, como si pintáramos con una pintura roja imaginaria esas secciones de la estructura del producto.

Ahora, podemos bajar por las estructuras de producto desde las órdenes correspondientes, pero sólo por las líneas rojas, almacenando las cargas (no los datos de los pedidos) que necesitamos para el gráfico de cargas que hemos mencionado antes. Esto supone una cantidad de datos relativamente pequeña, que no representará una carga significativa para la memoria *on-line* disponible. Sólo cuando el usuario desee ver los detalles de los pedidos correspondientes al trabajo específico en el que esté interesado, subiremos por las estructuras de producto, sólo para ese trabajo, para encontrar y mostrar el contenido de los pedidos. Pueden parecer muchas subidas y bajadas repetitivas, pero recordemos que esta actividad sólo tarda unos segundos, o, incluso, menos. Es cierto que requerirá un mayor esfuerzo de programación, pero, ¿qué es más importante, el trabajo inicial del programador, o el consumo continuado de la paciencia y el tiempo del usuario?

Así que, una vez que hemos pasado la fase de verificación de los pocos datos que nos hicieron sospechar la existencia de un recurso limitado, sólo entonces estaremos en la poco envidiable situación de tener que resolver un conflicto aparente entre las limitaciones de la empresa. ¿Cómo se supone que debe tratar esto un sistema de información? ¿Dejándolo en manos del usuario? ¿Qué haría éste entonces? No, aquí llega una verdadera prueba para nuestro sistema de información, reducir el conflicto a sus raíces más básicas, para que nosotros, los usuarios, nos encontremos ante alternativas muy claras. Tan claras que no tengamos ninguna dificultad para elegir entre ellas usando solamente nuestra intuición.

31

Localización de los conflictos entre las limitaciones identificadas

La identificación de un cuello de botella implica que no podemos satisfacer todos los pedidos en sus fechas requeridas; no tenemos suficiente capacidad disponible, algo tendrá que ceder. También es evidente que el sistema de información no tiene los datos necesarios para tomar la decisión, ni siquiera en el caso de que, por algún milagro, consiguiéramos meter en el sistema las rutinas necesarias.

¿Qué cliente se enfadará menos si no entregamos a tiempo? ¿Podemos aliviar la presión haciendo un envío parcial? Si es así, ¿a quién y en qué cantidad? ¿Hay alguna forma de ejecución alternativa? Nada de esto es deseable normalmente, pero es peor perder a un cliente. Sí, es imposible volver a pedir a la gente que trabaje el fin de semana, ni siquiera lo vamos a considerar, excepto si..., etc. No, no es realista en absoluto pretender que este conjunto de datos, tan voluminoso y tan indefinido, se pueda meter en el sistema.

Lo que necesitamos es un método para que el sistema de información nos pueda enfocar sobre el conflicto, para que la dirección responsable, y no el sistema, pueda tomar este tipo de decisiones con facilidad.

¿Cómo lo vamos a hacer? Vamos a usar nuestro viejo y bien probado método: los cinco pasos de enfoque. ¿Pero no hemos identificado ya las limitaciones? ¿No tenemos ya un conflicto entre las limitaciones identificadas? No importa. El hecho es que estamos confusos, y éste es el momento más adecuado para seguir los cinco pasos de enfoque.

Vamos a empezar con la primera limitación que identificamos, la limitación del mercado; y vamos a pasar al segundo paso, *explotar*. Esto significa, simplemente, que nos gustaría cumplir con todas las fechas de entrega requeridas. Bien, ahora vamos a subordinar. Al final de este paso, se supone que debemos comprobar si se ha generado algún conflicto. Ya conocemos la existencia de uno, el conflicto con la capacidad limitada de

uno de los recursos. Así que, en vez de intentar subordinar toda la empresa a la decisión de explotación, sólo vamos a intentar subordinar las acciones de ese recurso específico. Vamos a buscar los detalles del conflicto resultante.

¿Qué significa subordinar un recurso? Significa ignorar sus propias limitaciones y concentrarse en conocer exactamente lo que nos gustaría que hiciera ese recurso para satisfacer la limitación. Muy bonito. Así que, dado un pedido específico, ¿qué nos gustaría que hiciera nuestro recurso para satisfacerlo? Realizar el trabajo necesario, producir el número de unidades necesarias para cumplir con el pedido. Bien, ¿cuántas unidades? Depende.

¡Vaya hombre! Si yo soy vuestro obediente recurso, estoy deseando hacer lo que me digáis, así que, por favor, decídmelo. No a base de pistas e indicios, decídmelo directamente, usando números concretos siempre que sea posible.

Vale, nos hemos enterado. De repente, estamos hablando con un ordenador, un tonto completamente obediente. Probablemente eso es lo que debemos suponer cuando estamos intentando desarrollar un procedimiento muy riguroso. ¿Cuántas unidades tiene que producir el recurso? Eso es bastante fácil de calcular. Empezamos con la cantidad requerida por el pedido, hacemos la *explosión* de necesidades a través de la estructura del producto, y traducimos este número a unidades que se piden al recurso. No olvidemos que, en la estructura del producto, la «cantidad por», escrita en las flechas de conexión, puede ser distinta de uno. Por ejemplo, si el pedido es de coches y nuestro recurso tiene que producir las ruedas, un pedido de 100 coches se traducirá en un requerimiento para producir 400 ruedas. O, si un pedido es por veinte tornillos, puede traducirse en la necesidad de preparar un documento de facturación.

Eso no es suficiente en sí mismo. Puede que tengamos en la estructura del producto algunos datos relativos a la expectativa de piezas defectuosas. Así, nos podemos encontrar con que, para entregar al cliente 100 unidades, tendremos que pedir al recurso que haga 110, y no sólo 100. ¿Ya estamos contentos con la respuesta?

No del todo. No podemos suponer que empezamos con la pizarra en blanco, con las líneas vacías. ¿Qué pasa con las tareas que ya están entre nuestro recurso y los pedidos en el momento de hacer la programación? Necesitaremos deducir estos inventarios, que es exactamente lo que hicimos cuando calculamos las cargas de los recursos.

Sí, pero, ahora, nos encontramos con otro problema. ¿Cómo vamos a asignar esos inventarios? Esto no tenía importancia cuando considerábamos la carga total, pero, ahora estamos intentando determinar las instrucciones detalladas. ¿A qué pedido debemos asignar una cantidad determinada de inventario si ésta es suficiente para cubrir algunos de los pedidos pero no todos? Este es un ejemplo excelente de cómo coger un gatito doméstico y presentarlo como un tigre salvaje, dando una enorme importancia a una serie de procedimientos técnicos triviales y bien establecidos. ¿Qué estamos intentando conseguir? ¿Impresionar a todo el mundo con nuestra competencia en asuntos técnicos? Ya estábamos de acuerdo en que nuestra discusión pretendía descubrir cómo se puede construir un sistema de información. ¿Se nos ha olvidado nuestro objetivo original?

Lo de arriba también es un buen ejemplo de lo fácil que es ahogarse en los detalles, ahogarse hasta el punto de que no sólo queda borrado el cuadro global, sino que, además, se abren las puertas para la supersofisticación; una sofisticación que puede arruinar la facilidad de uso del paquete. ¿Cómo asignamos el inventario? De acuerdo con las fechas de entrega requeridas en los pedidos. El que tenga una fecha de entrega más temprana gana al que la tenga más tardía. No tiene sentido usar reglas más sofisticadas. No intentemos decidir qué pedido es más importante. En cualquier caso, este tipo de datos nunca está rápidamente disponible para un sistema mecanizado. No intentéis considerar el *lead-time* desde el inventario hasta el pedido; en todo caso, esos *lead-time* estarán en función de nuestro querido y viejo Murphy. No intentemos pasarnos de listos, el resultado final nos demostrará que nos equivocábamos. Nosotros, por lo menos, vamos a seguir con la sencillez. «Al que llega primero se le sirve primero». Es una regla simple y buena.

¡Qué tomadura de pelo! Por favor, un poco de calma. Sólo quería encontrar honradamente cómo subordinar el recurso limitado a la decisión de explotar la limitación del mercado. Puede que para ustedes resulte fácil, pero no debemos ignorar el hecho de que es la primera vez que yo hago estas cosas. Ya resulta bastante difícil de entender lo que de verdad se quiere decir con subir y bajar por las estructuras de producto. No hace falta que lo hagan más difícil aún mostrando su impaciencia.

Hemos calculado la cantidad que tiene que hacer nuestro recurso para satisfacer un pedido específico. Desde ahí, usando el tiempo necesario para fabricar una unidad y el tiempo de cambio, podemos deducir fácilmente la cantidad de carga que se está asignando al recurso. Ahora lo que nos falta es encontrar dónde situar esta carga en el eje de tiempo, determinar *cuándo* tiene que trabajar el recurso para satisfacer ese pedido, suponiendo que el recurso en sí no tiene limitaciones.

Es bastante fácil contestar a esta pregunta. Tenemos la fecha requerida del pedido, el usuario nos ha dado la longitud del *buffer* de envío, la cantidad de tiempo que tenemos que asignar para que el trabajo llegue a su destino con seguridad. Así que debemos «soltar» el trabajo desde el recurso limitado con una antelación sobre la fecha de entrega igual a la duración del

buffer de envío. En otras palabras, idealmente, el recurso limitado debe terminar su trabajo un *buffer* de envío antes de la fecha de entrega.

Lo que tenemos que hacer ahora es repetir el mismo cálculo para todos los pedidos que requieran trabajo de nuestro recurso limitado, colocando los «bloques» de trabajo resultantes sobre el eje de tiempo del recurso. Como tenemos que realizar la asignación de inventarios intermedios y generar los bloques para cada uno de los pedidos, podemos combinar estas dos tareas y hacerlas en un solo paso. Como decidimos asignar según la fecha de entrega de los pedidos, la forma más fácil de hacerlo es empezar con el pedido más cercano en el tiempo y, moviéndonos hacia delante, continuar con cada uno de los demás. Tendremos que generar el bloque de trabajo que se requiere del recurso limitado para cada pedido que no tenga suficiente inventario en la línea.

Por supuesto que estos bloques pueden quedar apilados uno encima del otro, ya que los estamos colocando sin considerar las limitaciones propias del recurso. Probablemente, obtengamos un cuadro como el de la figura 31.1, semejante a unas «ruinas». Un escenario totalmente impracticable. No es realista pretender que un recurso trabaje simultáneamente en más de un bloque, así que, si tenemos, por ejemplo, una sola unidad de ese recurso, no podemos trabajar con una pila de bloques.

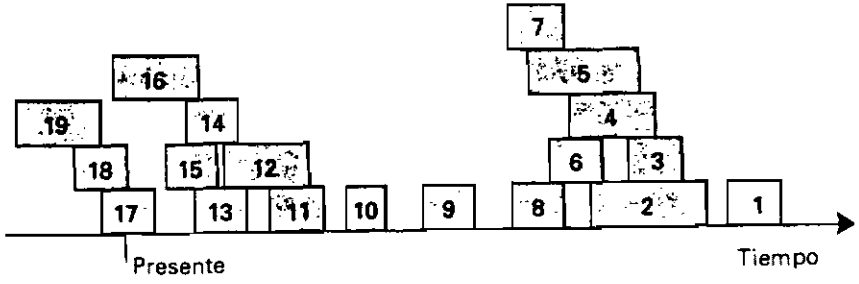


Fig. 31.1. *«Las ruinas».*

Nos encontramos ante un conflicto. No nos sorprende, lo estábamos esperando, y ahora es el momento de intentar resolverlo. Debemos «nivelar las ruinas». Debemos asegurarnos de que la cantidad de bloques que se necesita hacer en el mismo momento nunca excederá del número de unidades disponibles del recurso limitado. Esto quiere decir que cada vez que encontremos una acumulación, tendremos que desplazar los bloques superiores. ¿Hacia qué lado de las ruinas debemos desplazarlos? ¿Debemos posponer la ejecución? Obviamente no, sidesplazamos un bloque hacia adelante en el tiempo, ponemos en peligro la fecha de entrega del pedido correspondiente. Así que vamos a nivelar las ruinas desplazando

los bloques superiores hacia la izquierda, situándolos antes de lo que sería estrictamente necesario según las fechas de entrega de sus pedidos. En realidad, esto significa un aumento de inventario, pero eso es mejor que perder valor generado.

Así que vamos a imaginarnos una gran máquina niveladora a la derecha de nuestro cuadro de ruinas. Ajustemos su cuchilla a una altura igual al número de unidades disponibles del recurso. Ahora, decimos a la niveladora que se mueva hacia la izquierda, nivelando el suelo. Nuestra niveladora es un instrumento muy especial. Mantiene el orden (la secuencia) de los bloques; un bloque cuyo lado derecho aparece detrás (más tarde) del lado derecho de otro bloque, mantendrá su posición relativa, incluso después de que la niveladora haya nivelado el suelo. Es muy fácil convertir este cuadro figurado en un código de ordenador, así que, no hace falta que nos extendamos más en detalles técnicos. Un brote de impaciencia ha sido suficiente.

Pero, un momento, no hemos terminado. No hemos eliminado el conflicto, sólo lo hemos cambiado de sitio. Como estamos tratando con un recurso que no tiene bastante capacidad, es inevitable que el resultado final del trabajo de nuestra máquina niveladora sea que algunos bloques quedan en el lado incorrecto de nuestro eje de tiempo. Inevitablemente, algunos bloques de trabajo se desplazarán hacia el pasado. Es evidente que no podemos pedir a un recurso que haga un trabajo ayer. Es más, nuestra técnica de nivelación empeorará las cosas. Dependiendo de la distribución de las fechas de entrega de los pedidos, puede que deje huecos hacia la derecha, empeorando las cosas donde de verdad importa: a la izquierda del cuadro.

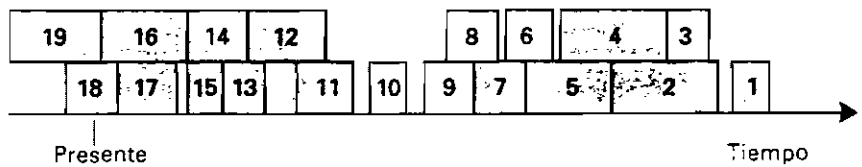


Fig. 31.2. *Nivelación de las ruinas de acuerdo con la disponibilidad de dos unidades del recurso.*

¿Peor? No exactamente, es sólo una representación más precisa del conflicto. Llegados a este punto parece inevitable que nos enfrentemos a la realidad: no podemos hacer cosas en el pasado. Algunas de las fechas de entrega tendrán que ceder. Vamos a coger nuestra niveladora. Ahora mismo está aparcada a la izquierda de nuestro cuadro, le damos la vuelta y bajamos la cuchilla para que actúe como una pared de acero. Ahora le

ordenamos que se mueva, que empuje los bloques hacia la derecha, hasta el punto en el que no quede ningún trabajo en el pasado. Con este último movimiento hemos evitado la posibilidad de chocar con la realidad, pero ¿qué pasa con las fechas de entrega de los pedidos?

Por supuesto que la manipulación de los bloques con la niveladora, moviéndolos primero hacia el pasado y después hacia el futuro, los deja en posiciones distintas de las originales. Como hemos empezado con un recurso limitado, es inevitable que algunos bloques queden más tarde (hacia la derecha) de lo que estaban originalmente. Esto quiere decir que los pedidos que les corresponden están en peligro. Si el desplazamiento resultante es de más de medio *buffer*, no es ya que la empresa tenga una probabilidad alta de no cumplir la fecha de entrega, es que podemos estar seguros de que Murphy lo convertirá en una certeza. Por tanto, en vez de llenar de datos la pantalla, vamos a limitarnos a destacar en color rojo los bloques peligrosos. Un sistema que haga todo lo antedicho proporcionará al usuario un cuadro muy preciso y muy concentrado del conflicto que hay entre las limitaciones de la empresa.

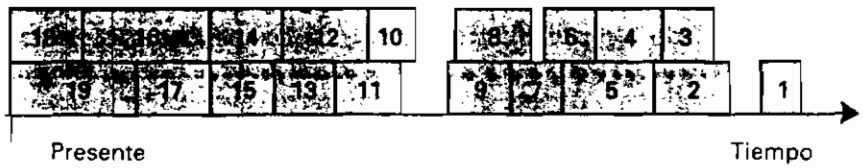


Fig. 31.3. *Nivelación de las cargas en los recursos limitados, teniendo en cuenta que no podemos hacer cosas en el pasado.*

¿Lo debemos dejar así, esperando que intervenga el usuario? Todavía no. Todavía tenemos algo que hacer, pero antes de continuar vamos a tomarnos un tiempo para digerir el interesante mecanismo que hemos descrito. No se parece a ninguna otra técnica de programación que hayamos visto. Ni siquiera se puede clasificar dentro de los dos grupos principales de métodos de programación que había antes: los métodos que iban hacia atrás o hacia adelente en el tiempo. De hecho, por lo que se ha descrito hasta ahora, es obvio que este método ni siquiera usa el eje de tiempo como guía principal de la programación. Esta técnica, hasta ahora, es simplemente una derivación directa de los cinco pasos de enfoque, que son, a su vez, una deducción lógica directa de la elección que hicimos del *valor generado* como medida número uno.

Vamos a dedicar algún tiempo a estudiar los gráficos correspondientes: la figura 31.1 nos enseña las «ruinas» originales; la figura 31.2 nos enseña el resultado de la primera pasada de la niveladora, y la figura 31.3 nos

enseña el resultado final hasta su momento. ¿Pueden imaginarse cuál sería el resultado si, en vez de tratar con un cuello de botella, estuviéramos tratando con un recurso que tiene suficiente capacidad media, pero que se queda corto en intervalos específicos de tiempo? Parece que ya tenemos un método genérico para tratar con cualquier tipo de recurso limitado. Por supuesto, todavía tenemos que explorar el procedimiento general de cómo subordinar no sólo el recurso limitado, sino, también, todos los demás, pero nos estamos acercando.

32

Comienzo de la resolución de conflictos: la relación sistema-usuario

¿Dónde hemos dejado al usuario? ¡Ah, sí! Mirando una pantalla que muestra los bloques de trabajo de la limitación, algunos pintados, como advertencia, de rojo, lo que indica que sus pedidos correspondientes van a llegar tarde. No podemos dejar así la situación, y el camino más fácil, decir al recurso que empiece a trabajar antes, no se puede usar en este caso. No hay ningún bloque rojo, si no hemos cometido ningún error lógico o de cálculo, que se pueda mover hacia la izquierda. La forma en que hemos secuenciado las tareas nos garantiza que nuestro recurso está totalmente ocupado desde el presente hasta el primer bloque rojo, haciendo otros bloques aún más urgentes. Parece que la única forma de entregar a tiempo este pedido consiste en violar alguna otra cosa; o bien darle prioridad sobre otro pedido anterior, lo que significa que ese pedido se entregará tarde, o bien, aumentar la capacidad disponible del recurso.

Como ya hemos dicho, todas esas manipulaciones pueden hacerse, pero no las debe hacer el sistema de información, las debe hacer el propio usuario. Pero, un momento, antes de pasar la pelota al usuario, ¿hemos terminado todo lo que teníamos que hacer? No del todo. Por un lado, puede que sólo hayamos presentado al usuario una parte de la historia; la situación real puede ser aún peor de lo que se ha presentado. Por otro lado, puede que todavía podamos ayudar a resolver los conflictos.

Primero vamos a comprobar si el cuadro que hemos presentado de verdad ha tenido en cuenta todo lo que sabíamos. Lo que nos preocupa es el hecho de que, para explotar al recurso limitado, le hemos dicho que trabaje desde el presente. ¿Hemos comprobado si se pueden cumplir esas instrucciones, si los materiales necesarios para realizar estos trabajos están, de hecho, allí, esperando a la limitación? Porque si no es este el caso, ¿cuál es la probabilidad de que estén disponibles dentro de unos minutos? ¿Estamos dándole a la limitación unas instrucciones de trabajo que son

imposibles de cumplir en este momento? Lanzar un pograma donde, ya en el primer paso, la limitación tiene que improvisar, no es lanzar un programa, parece más bien que estamos lanzando el caos.

¿Podemos tener en cuenta esta consideración? Por supuesto. Disponemos de todos los datos necesarios. Sabemos qué material, y cuánto, está esperando en este momento delante de la limitación. Así que lo que tenemos que hacer es manipular un poco más en el cuadro que aparece en la pantalla. Nos tenemos que asegurar de que los primeros bloques ya tienen su material correspondiente esperando. Lo que significa que, si no es así, el sistema tendrá que buscar los primeros bloques que ya tengan material disponible para recolocarlos al principio del todo.

¿A qué debemos llamar el principio? ¿Qué período de tiempo debemos proteger al principio programando en él bloques que ya tengan el material disponible? Lo suficiente como para que nos permita traer el resto de lo que se necesita. Bonita respuesta, pero, ¿tenemos alguna pista de cuál es este intervalo de tiempo? Sí, y más que una pista. El usuario nos ha dado el cálculo del *buffer* del recurso limitado. Este es el cálculo del tiempo necesario para que un trabajo llegue desde su lanzamiento hasta el recurso limitado, para que llegue con una probabilidad alta y sin problemas. ¿Debemos aplicar a nuestra situación ese conocimiento, tal y como está? No olvidemos que estamos bajo presión, que estamos posponiendo cosas urgentes. Bien, digamos entonces que, como es urgente y estamos dispuestos a acelerar trabajos, deberíamos proteger los bloques sólo hasta el punto de que tengan que hacerse no más tarde que dentro de medio *buffer*, contando desde el momento presente. ¿Por qué medio? ¿Por qué no dos tercios del *buffer*? Si empezamos improvisando en la limitación, seguro que vamos a causar que llegue tarde más de un pedido. Bien, que sean dos tercios. Pero, después de este período de tiempo, podemos suponer con seguridad que los materiales necesarios ya habrán llegado a la limitación. Por lo tanto, podemos programar el recurso limitado de ahí en adelante, sin preocuparnos de la disponibilidad actual de los materiales.

Ahora estamos bastante protegidos, pero ¿no agravarán nuestros problemas todas estas manipulaciones? ¿No se crearán más bloques rojos? Probablemente, pero esto es la vida real. No es realista planificar trabajo si no está disponible el material, sólo para que el cuadro tenga mejor aspecto. El usuario tendrá que resolver todos los bloques rojos, incluyendo el que acabamos de crear. ¿Podemos echarle una mano? Sí, ciertamente, pero recuerden que en este momento, más que en ningún otro, nuestro sistema de información debe pasar a segundo plano, es el usuario el que debe tomar el mando.

Cuando nuestro recurso requiera tiempo de cambios de modelo, hay una posibilidad remota de mejorar su rendimiento sin chocar con los límites de las condiciones necesarias. Mejorar su rendimiento, ahorrando una cantidad restringida de tiempo de cambio, para que la única penalización sea un aumento del inventario. Dijimos que siempre que haya que elegir entre valor generado (ingresos netos) e inventario, nuestro sistema lo haría sin pedir permiso. Lo dijimos y, sin embargo, aquí nos vamos a saltar la regla.

Aquí, en todo caso, el usuario va a tener que manipular la secuencia manualmente. Esta oportunidad de mejora del rendimiento que vamos a comentar ahora no será suficiente, en la mayoría de los casos, para resolver todos los conflictos, y, por lo tanto, tendremos que detener la ejecución y molestar al usuario. Es más, al mismo tiempo, sólo podemos conseguir la mejora cambiando la secuencia natural de los bloques. Ya no seguirán el orden de las fechas de entrega y, por lo tanto, al usuario le resultará más confuso conseguir una buena gestión de los cambios adicionales que se requieren. Así pues, creo que es mejor presentar al usuario el cuadro «desnudo» y pedirle una instrucción específica para realizar los ahorros triviales de tiempo de cambio.

¿Qué es esta oportunidad que hemos mencionado? ¿A qué llamamos ahorros triviales de tiempo de cambio? Siempre que haya dos bloques idénticos existe la posibilidad de ahorrar tiempo de cambio. Bloques idénticos quiere decir que dos pedidos distintos necesitan la ejecución del mismo código de pieza/operación. Cada bloque tiene una previsión de tiempo de cambio, a menos que ambos se hagan consecutivamente, uno inmediatamente después del otro. En esos cambios no hay necesidad de hacer otro cambio en el recurso, y, por lo tanto, no hay que prever tiempo de cambio para el segundo bloque. Por lo tanto, mover uno de los dos bloques idénticos para que se fabriquen consecutivamente permitirá «pegar» los bloques, lo que liberará capacidad, permitiendo que la limitación satisfaga más pedidos en el mismo intervalo de tiempo.

¿Cuáles son los aspectos negativos? Uno es que para poder «pegar» bloques tenemos que adelantar el bloque más lejano en el tiempo, haciéndolo antes de lo que sería necesario (si atrasamos el bloque que está más cercano en el tiempo atrasaríamos su pedido correspondiente). Esto supone que lanzamos el inventario antes. El inventario aumentará debido a nuestro deseo de ahorrar tiempos de cambio. Si esto ayuda a enviar a tiempo otros pedidos, estamos dispuestos a pagar el precio del aumento de inventario. Por supuesto, esto significa que si intentamos «pegar» dos bloques idénticos cuando no hay ningún bloque rojo después del segundo, es una indicación clara de que hemos perdido de vista la meta. ¿A quién va a ayudar este «ahorro» del tiempo de cambio? ¿Vamos a hacer más envíos, a incrementar los ingresos netos?

Si el aumento de inventario fuera el único aspecto negativo de «pegar»

bloques, lo haríamos rutinariamente. Todos conocemos muchos casos en los que el tiempo de cambio es bastante largo y hay muchos pedidos «enanos» que requieren tiempos de proceso ridículamente cortos. Si no pegamos los bloques en estos casos, nos encontramos con una situación en la que la mayoría del tiempo del recurso limitado no estará dedicado a producir, sino a hacer cambios. Entonces, ¿por qué decimos que estos ahorros de tiempo de cambio son una mejora pequeña? La respuesta reside en el hecho de que no estamos en contra de pegar bloques, estamos en contra de permitir que el sistema de información lo haga sin un control cuidadoso por parte del usuario. En la mayoría de los casos el ahorro de tiempo de cambio lleva una penalización mayor que el simple aumento del inventario. Vamos a investigarlo con más profundidad.

Cuando pegamos dos bloques, todos los bloques que se iban a hacer después del segundo bloque saldrán beneficiados; se va a trabajar en ellos antes de lo que se haría si no hubiéramos pegado los bloques. Se van a adelantar un tiempo igual a la cantidad de tiempo ahorrada en el cambio. Pero, ¿qué sucede con los bloques que estaban entre los dos que hemos pegado? Todos ellos se harán más tarde. Su ejecución se pospondrá un tiempo igual al necesario para producir el segundo bloque, el que se saltó la fila. Si estamos tratando con un caso en el que hay, al menos, un bloque rojo entre los dos bloques idénticos, pegar los dos bloques idénticos supone que sabemos qué pedido es el más importante para entregarlo a tiempo. Este tipo de decisión no debe tomarla el sistema de información.

A pesar de ello, hay bastantes situaciones en las que el usuario tendrá que tomar decisiones de este tipo. ¿Va a permitir el sistema que el pegado sólo se pueda hacer manualmente? En algunas situaciones bastante comunes, la manipulación manual —tratando cada bloque por separado— volverá loco al usuario. Debemos proporcionar una ayuda automatizada. Es más, recordemos que cada vez que ahorramos un cambio, no sólo estamos cambiando la situación de un bloque, estamos afectando a muchos bloques. Y como ya hemos dicho, algunos se pondrán rojos y es de esperar que otros dejen de estarlo. Es evidente que debemos proporcionar algún método que permita al usuario no sólo dar instrucciones bloque a bloque sino, también, dar una instrucción mucho más amplia, permitiéndole ver el efecto de su instrucción en todos los puntos conflictivos.

¿Qué tipo de instrucción general sería apropiada? El pegado de bloques es más importante cuando los tiempos de cambio son relativamente grandes. No olvidemos que el tiempo de cambio no es sólo función del recurso, también es función del trabajo específico que vaya a realizar. Por lo tanto, aunque aquí estamos tratando con un solo tipo de recurso, los distintos bloques representarán oportunidades de ahorro del tiempo de cambio que pueden ser mayores o menores. Por otra parte, para ahorrar tiempo de

cambio, tenemos que ir hacia el futuro y traemos de allí un bloque. Cuanto más lejos vayamos, habrá más bloques que sufrirán debido a esta maniobra (todos los que están entre el bloque original y el que estamos adelantando).

Así pues, es razonable pedir al usuario una instrucción general expresada como un número que represente una relación, la relación entre el tiempo que se va a ahorrar y el tiempo que se permite que el sistema vaya hacia el futuro. Por ejemplo, un número 100 se interpretará de la siguiente forma: por cada hora de cambio ahorrada, se permite que el sistema traiga un bloque que esté separado de su bloque idéntico por un máximo de 100 horas hacia el futuro.

Como la mayoría de los usuarios nunca han tenido la oportunidad de ver el impacto de esto en sus operaciones, este número no debe ser un parámetro de entrada: el usuario debe tener la capacidad de probar, sobre la marcha, varios números distintos hasta que quede satisfecho con los colores de los bloques que le muestra el cuadro. Como la ejecución de una instrucción así (intentando ver la influencia de un número sobre los colores de los bloques) no debe tardar más de unos segundos, aquí estamos hablando de un auténtico sistema de soporte de decisiones —un sistema que se ha diseñado de forma que no ignore el hecho de que la intuición humana no siempre se puede expresar en números y que, al mismo tiempo, no «ayuda» presentando datos conocidos, sino presentando los resultados de la decisión alternativa del usuario.

¿Hemos resuelto ya todos los conflictos? ¿Ya hemos eliminado el rojo de la pantalla? Probablemente hemos reducido el número de bloques rojos, pero, todavía, nos quedan algunos: recuerden, *sabemos que estamos tratando con un cuello de botella.* En realidad, en las situaciones en las que el recurso limitado no tiene un tiempo de cambio significativo, todavía no hemos hecho nada para eliminar los conflictos.

Así que ha llegado el momento de usar un cañón más grande. Las horas extras.

33

Resolución de los restantes conflictos

Como ya hemos mencionado antes, el sistema debe facilitar que entre los datos de entrada se puedan dar unas directrices amplias sobre las horas extras permitidas en cada recurso. Estas directrices se dan con el entendimiento claro de que las horas extras se usarán sólo en el caso de que los ingresos netos se puedan perjudicar si no se hacen, y no se usarán para ningún otro propósito artificial.

Las instrucciones para las horas extras se dan en forma de límite: cuál es el máximo de horas que se permite por día, por fin de semana, y cuál es el total máximo que se permite por semana. La necesidad de especificar las horas semanales se debe a que, algunas veces, el total de horas que se permite por semana es inferior a la suma de las horas diarias más las del fin de semana. El cansancio de la gente debe ser una consideración prioritaria.

Como podremos ver más tarde, las instrucciones sobre horas extras, en lo que respecta a recursos no limitados, se van a obedecer sin que el usuario tenga que dar ningún permiso adicional. Este no es el caso cuando tratamos con un recurso que se ha identificado como limitación; aquí es donde estuvimos ocupados intentando ahorrar tiempos de cambio. Sólo después de que el usuario haya agotado esa vía tendrá sentido la inyección de horas extras y, por lo tanto, se requiere la iniciativa del usuario.

Lo que tenemos que tener presente es que, siempre que autoricemos horas extras para una fecha determinada, el beneficio no va a ser sólo para los bloques que había que terminar al final de ese día; también se van a beneficiar todos los demás bloques posteriores a esa fecha, ya que se podrán hacer antes de lo que se había planificado. Por lo tanto, nos podemos encontrar con una situación en la que cuando permitimos algunas horas extras para ayudar a un determinado bloque rojo, puede que éste siga siendo rojo, pero muchos otros bloques de los que se van a hacer

más tarde pueden cambiar de rojo a normal. Es más, esta última observación nos indica que para «ayudar» a un bloque determinado no sólo podemos hacer horas extras en el día de fabricación de ese bloque, también las podemos hacer en los días anteriores. Por supuesto que si autorizamos que se hagan las horas extras con antelación, respecto a la fecha del bloque, aumentaremos el inventario de la empresa.

Parece que ya estamos preparados para dar instrucciones sobre horas extras, incluso a un estúpido ordenador. Lo que necesitamos es convertir en acciones lógicas lo que acabamos de decir. Dijimos que las autorizaciones de horas extras sólo se darán para mejorar los ingresos netos; por lo tanto, lo que debe disparar la consideración de horas extras es el hecho de que haya bloques rojos en nuestra pantalla. Si ninguno de los bloques está en rojo, no hay ningún pedido en peligro de llegar tarde. ¿Por qué íbamos, siquiera, a considerar las horas extras?

¿Qué bloques rojos debemos considerar primero? Dijimos que una hora extra en una fecha específica ayuda, exactamente en la misma medida, no sólo a un bloque, sino a todos los que se deben hacer después de esa fecha. No queremos despilfarrar horas extras (recordemos que, al contrario que las horas normales, cada hora extra aumenta los gastos operativos) y, por lo tanto, lo mejor es que las situemos donde más pueden ayudar, es decir, antes del primer bloque rojo. También hemos dicho que cuanto más temprana sea la fecha de las horas extras, la penalización que pagamos en incremento de inventario es mayor. Así pues, no sólo debemos situar las horas extras antes (a la izquierda de) del primer bloque, también tenemos que situarlas tan cerca de ese bloque como sea posible.

Así que vamos a empezar concentrándonos en el primer bloque rojo, ignorando los demás. Nos movemos hacia atrás en el tiempo, asignando horas extras donde esté permitido. Continuamos haciéndolo hasta que pase una de estas dos cosas: o el bloque rojo cambia de color o, desgraciadamente, topamos con el presente. Por supuesto que si pasa esto último quiere decir que hemos agotado la vía de las horas extras, como posible ayuda al pedido correspondiente, y tendremos que pedir al usuario algún medio más drástico.

¿Hemos terminado? Todavía no, recordemos que parece que nuestra pantalla tiene viruela. El bloque rojo que hemos atacado no era el único, sólo era el primero de muchos. Es cierto que las horas extras que ya hemos autorizado también habrán ayudado a los demás bloques rojos, puede que alguno, incluso, haya cambiado de color. Pero, a pesar de ello, normalmente no será suficiente, todavía tendremos muchos bloques rojos coloreando nuestra pantalla. Lo que el sistema tiene que hacer es, simplemente, repetir el mismo proceso, esta vez con el siguiente bloque rojo, que, por definición, debe estar a la derecha de nuestro bloque original (que esperamos

que ya no esté en rojo). El sistema debe continuar bailando esta danza tan interesante, un paso adelante —al próximo bloque rojo— varios pasos atrás —asignando horas extras donde se permita— hasta que haya terminado con los bloques rojos.

Si la situación es realmente grave, o si las posibilidades de horas extras son escasas, terminaremos teniendo, todavía, algunas manchas rojas perturbando nuestra bella pantalla. Ahora debemos pasar a un tratamiento «individual». El usuario todavía puede manipular la pantalla. Puede decidir desviar un bloque determinado para que lo haga otro recurso, dividir un bloque y quitarle una parte para hacerla en otra fecha, o decidirse por una mayor inyección puntual de horas extras. Todo vale, el usuario es el que manda.

Este tratamiento «individual» debe entenderse correctamente. Las actuaciones se centran en resolver el problema de un bloque rojo determinado, pero esto no quiere decir que sólo tengan influencia en ese bloque. Normalmente pasa lo contrario. Por ejemplo, si desviamos un bloque determinado a otro centro de trabajo (no limitado), ciertamente ayudaremos a este pedido en particular, pero, también, mejoraremos la situación de todos los bloques cuya fecha de ejecución es posterior a la del bloque desviado. Esto significa que la pantalla que presenta la situación actualizada de todos los bloques de la limitación es muy importante para ver el impacto de la decisión del usuario, de hecho, es vital. Esta presentación de los bloques en el tiempo resulta ser más efectiva de lo que no imaginábamos al principio.

Pero esta felicidad no debe cegarnos, la alternativa más importante sigue en manos del usuario; éste siempre se puede dar por vencido. No, no es una broma. Aunque todavía haya bloques rojos en la pantalla, el usuario puede decidir lo siguiente: «He hecho todo lo humanamente posible, tendré que aceptar que algunos pedidos van a llegar tarde». Una afirmación así significa que no hay ninguna forma conocida de eliminar los conflictos gestionando la capacidad limitada del recurso limitado. Por lo tanto, la única forma que nos queda de resolver el conflicto es retrasar la fecha prometida de entrega de los pedidos correspondientes.

Ahora, el sistema debe hacer exactamente eso. Debe empujar hacia el futuro la fecha de entrega de cada pedido que tenga un bloque rojo. ¿Cuánto? La nueva fecha de entrega debe ajustarse para que sea igual a la fecha de finalización del último de sus bloques rojos, más el *buffer* de envíos. Se deben mostrar los pedidos afectados. Muchas veces, cuando nos damos cuenta de la magnitud de los atrasos, encontramos, de repente, «nuevas» formas de proporcionar más capacidad. Se debe presentar al usuario el resultado final y, después, se le debe dar la oportunidad de que continúe luchando con los bloques rojos.

No nos gusta fallar en las fechas de entrega, pero hay algo a lo que debemos temer aún más: fallar en una fecha de entrega sin avisar al cliente. ¿Cuál es la situación aquí? No tenemos suficiente capacidad, ya hemos intentado todo lo que se nos ha podido ocurrir, y nos sigue faltando. A no ser que suceda un milagro, seguro que vamos a fallar en algunas fechas de entrega. Es mucho mejor llamar ahora, para dar al cliente la mala noticia con algunas semanas de antelación. Es muchísimo mejor que disculparse después del hecho.

Una vez terminada esta fase, ya hemos terminado nuestro primer intento de los que se suele llamar «programa maestro». Debemos destacar que este programa maestro se diferencia en un punto importante del concepto generalmente aceptado. No sólo se crea con los datos de los pedidos, sino, también, con los datos que se refieren al programa detallado del recurso limitado.

¿Por qué tenemos la precaución de referirnos a él como a un «primer intento»? Porque todavía no hemos terminado. Puede haber más recursos limitados. Así que, ahora, tenemos que realizar el siguiente paso: *subordinar todo lo demás a la decisión anterior*. Tenemos que derivar las acciones de todos los demás recursos, para que apoyen, con toda seguridad, lo que ya hemos decidido.

En cierto modo, lo que hemos hecho es fijar firmemente algunas actividades en el eje de tiempo, asegurándonos de que no había conflictos internos. Ahora, las demás actividades tienen que marchar de acuerdo con esto. Por eso llamamos «autorizar el *tambor*» a la fase que acabamos de terminar. Hemos determinado el ritmo de tambor al que deberá marchar toda la empresa.

Antes de profundizar y empezar a idear el procedimiento adecuado para subordinar todos los demás recursos, puede que debamos introducir en este punto un nuevo concepto, el concepto de *barras*. De hecho, en la mayoría de los sistemas no tiene aplicación en esta fase, pero hay algunos en que sí la tiene, y en unos pocos es dominante. Es más, en todos los entornos el concepto de *barra* es vital si se ha identificado un segundo recurso limitado, así que vamos a tratarlo ahora.

Podemos encontrarnos con situaciones en las que el recurso limitado tiene que realizar trabajo en más de una fase del mismo pedido. En este caso, el mismo pedido generará más de un bloque en nuestro programa. Esto no es especialmente problemático, siempre que un bloque no alimente al otro pasando por operaciones hechas por otros recursos. El dilema es, por supuesto, ¿cuanto deberíamos proteger la segunda (más tardía) operación de la limitación? No podemos dejarla sin protección —Murphy podría atacar en las operaciones intermedias— pero, por otra parte, tampoco es deseable retrasar el trabajo que ya ha hecho la limitación. Es evidente que

tenemos que proteger la segunda operación de la limitación, pero, también, es evidente que no podemos ser excesivamente precavidos, así que no debemos protegerla con un *buffer* de la limitación entero. Parecería razonable protegerla con la mitad de la longitud del *buffer* de la limitación.

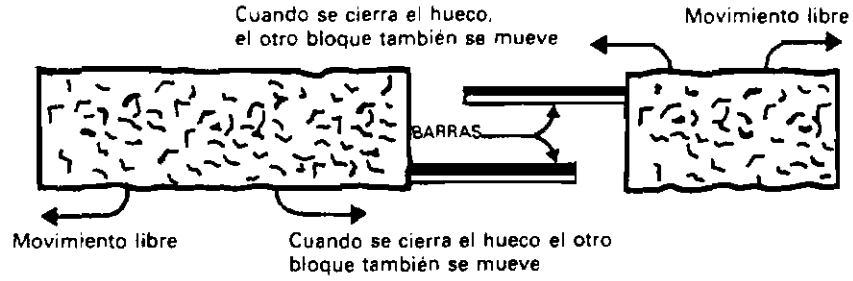


Fig. 33.1. *El concepto de barra.*

¿Eso es todo? Sólo en parte. En esta situación debemos tener cuidado cuando movemos bloques en el eje del tiempo. Si movemos el primer bloque, hacia atrás en el tiempo, no hay ningún problema, pero si lo movemos hacia adelante tendremos que mover a su «hermano menor». El hueco mínimo de tiempo que se permite entre los dos es, como ya dijimos, la mitad del *buffer* del recurso limitado. De la misma manera, si movemos hacia atrás el segundo bloque necesitaremos mover a su «hermano mayor». Parece como si hubiera unas barras «barras de acero» conectadas a estos bloques. El primer bloque tiene una barra que apunta hacia adelante en el tiempo, y el segundo bloque tiene una barra que apunta hacia atrás en el tiempo. La longitud de cada barra es igual a medio *buffer* del recurso limitado. Estos bloques se pueden mover libremente en el eje del tiempo, pero sus barras pueden provocar que su bloque correspondiente también se mueva. La figura 33.1 ilustra la situación. Por supuesto, un recurso limitado se puede alimentar a sí mismo muchas veces a lo largo del mismo trabajo, como en la industria electrónica, y, por lo tanto, puede tener más de una barra conectada. En este caso el movimiento de un bloque puede provocar el movimiento de otros muchos.

Todo esto es bastante fácil, pero antes de dejar el tema vamos a fijarnos un poco más en la longitud de la barra. No, no pretendo cuestionar la decisión, algo arbitraria, de coger medio *buffer*; a lo que me refiero es al hecho de que estamos tratando con los bloques como si fueran un punto, en vez de un intervalo de tiempo. Veamos qué debemos tener en cuenta si consideramos la longitud de los bloques, y el hecho de que dos bloques pueden diferir considerablemente en su longitud.

Tenemos que asegurarnos de que cada unidad que se hace en el primer bloque tendrá tiempo suficiente como para llegar a su proceso en el segundo bloque. Si el primer bloque es mucho más largo que el segundo, tenemos garantizado que la última unidad se encontrará con problemas. Tampoco podemos decidirnos por mirar solamente la última unidad del

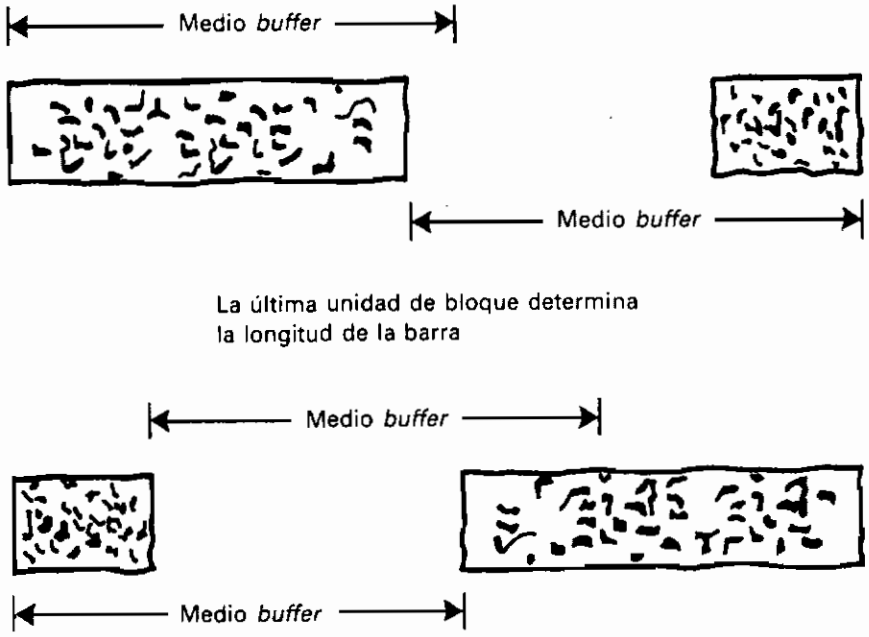


Fig. 33.2. *La primera unidad del bloque determina la longitud de la barra.*

bloque, porque si el segundo bloque es mucho más largo que el primero, la situación es exactamente la opuesta, como se muestra en la figura 33.2. La única solución es tener en cuenta los dos extremos cuando se calcula la distancia mínima que debe separar el comienzo de los dos bloques. Esta será la longitud efectiva de las barras correspondientes.

Una vez superado este tema, me resulta difícil encontrar otra buena razón para seguir entreteniéndome. Parece que no nos queda más remedio que continuar hacia la siguiente etapa de nuestro viaje, el tema de la subordinación.

34

Subordinación manual: el método DBR

¿Dónde estamos? Ya se han fijado, en un paso previo, las fechas de entrega de los pedidos y las fechas exactas para las operaciones (los bloques) del recurso limitado. Tampoco nos tenemos que preocupar de que haya desajustes entre las fechas; tenemos garantizado que cada una de las fechas de entrega está separada de su último bloque por, al menos, un *buffer* de envío (o, en casos de emergencia, por, al menos, medio *buffer* de envío). Ahora tenemos que fijar las fechas de todas las demás actividades.

Primero vamos a determinar las fechas de lanzamiento de los materiales para asegurarnos de que llegarán a tiempo al recurso limitado. Ya hemos establecido la regla que vamos a seguir: el material se debe lanzar un *buffer* de recurso antes de la fecha en la que lo tiene que consumir el recurso limitado. Por lo tanto, para determinar esas fechas de lanzamiento de materiales, sólo tenemos que restar un tiempo igual al *buffer* de recurso de las fechas que se han establecido en el *drum*.

¿Qué pasa con las fechas de todas las operaciones intermedias? No olvidemos lo que habíamos acordado; ciertamente no queremos que retengan el inventario, ya hemos decidido lanzar. Queremos que el inventario se acumule delante de las limitaciones, ya que sólo allí sirve de protección contra posibles perturbaciones. Así pues, las fechas que damos a los recursos no limitados que deben realizar operaciones intermedias son las fechas del lanzamiento del material; queremos que hagan todas las operaciones en el inventario tan pronto como lo tengan disponible, idealmente, el mismo día del lanzamiento.

¿Es cierto que todos esos recursos van a hacer su trabajo en esa fecha? No somos tan ingenuos: Murphy existe, seguro que nos vamos a encontrar con colas de material. Precisamente por esto es por lo que hemos lanzado tan pronto. Esto implica que, cuando demos a un recurso no limitado una orden con fecha específica, la interpretación no debe ser de ninguna mane-

ra «hazlo en esa fecha». La interpretación correcta es: «hágalo tan pronto como pueda, preferiblemente en el momento en que le llegue el material, pero, si el material llega antes de la fecha especificada, espere, no trabaje en él. Alguien ha cometido un error. Desde el punto de vista global del sistema, y teniendo en cuenta las *perturbaciones*, no hay necesidad de trabajar ahora mismo en esas piezas. Por favor, reténgalo hasta la fecha especificada».

Lo que nos lleva a la conclusión de que, a menos que haya mucho exceso de inventario en planta, no tiene mucho sentido dar un programa a los diversos centros de trabajo. Es suficiente con controlar firmemente el lanzamiento de materiales, y decir a todos los demás que trabajen, en la secuencia que sea, sobre el material que les llegue.

A todos, menos, por supuesto, a los recursos limitados. Los recursos limitados deben seguir rigurosamente la secuencia que con tanto trabajo hemos pulido. Deben intentar seguirla de la mejor forma que puedan y, por lo tanto, se les debe proporcionar una lista del *drum*.

Un momento, hay otra excepción, las operaciones que usan piezas comunes, piezas que se usan en más de una referencia/operación.

En estos centros de trabajo, una orden del tipo «si tienes material, trabaja en él» puede producir una divergencia del material hacia un canal incorrecto, causando, por un lado, falta de piezas y, por otro, exceso de inventario. Tendremos que dar el programa a esos recursos (no limitados) para evitar que usen demasiado pronto el material disponible. El programa para los recursos no limitados tiene ahora un significado completamente nuevo. No dice a los recursos cuándo deben hacer las cosas, sino lo contrario. El programa dice a los recursos cuándo no deben hacer algo; es una lista de «no hacerlo antes de...». No ha sido demasiado difícil determinar las fechas de las actividades que alimentan directamente a los recursos limitados. ¿Qué hay de las actividades que alimentan pedidos libres, pedidos que no requieren ninguna actividad de los recursos limitados? Lo que hemos dicho antes parece totalmente aplicable; lo único distinto que tenemos que hacer es usar el *buffer* de envío donde antes usábamos el *buffer* del recurso. La lógica, aun en este caso, parece impecable.

¿Hemos terminado? Todavía no. ¿Qué pasa con las actividades de las rutas rojas, las actividades que hay que hacer una vez que el recurso limitado ha hecho su parte? ¿Usamos como *drum* la fecha de cumplimiento del último bloque, intentando sacar esos trabajos tan rápido como podamos, o usamos, como en otros casos, la fecha de la limitación a la que alimentan: la fecha de entrega del pedido?

A primera vista puede parecer una pregunta con truco. Si ya nos hemos ocupado de resolver los conflictos entre las fechas del recurso limitado y las de entrega de los pedidos, entonces ¿qué importancia tiene que sigamos una u otra? Si lo examinamos un poco más, resulta aparente que la eliminación de los conflictos no quiere decir, necesariamente, que la pregunta no tenga significado.

Hemos construido el *drum* desplazando los bloques en el tiempo hacia atrás y hacia delante. Esto implica que algunos bloques se pueden haber programado antes de lo que sería estrictamente necesario para cumplir con su pedido. Se han programado antes porque, después, la limitación debe dedicar su tiempo a otros pedidos. Recordemos que los pedidos de los clientes no están llegando maravillosamente ordenados de acuerdo con la disponibilidad de nuestro recurso limitado, por no mencionar tendencias estacionales y otras cosas de ese tipo. Es bastante probable que encontremos bloques programados para hacerse antes de la fecha de entrega de su pedido menos el *buffer* de envío.

Por lo tanto, una vez más, en lo que respecta a todas las operaciones intermedias, la pregunta es: ¿debemos ordenar a los recursos no limitados que agilicen el paso del material tanto como puedan, lo que quiere decir que estaremos siguiendo la fecha de cumplimiento del último bloque? ¿O debemos restringirlos para que empiecen a trabajar en las unidades disponibles después de la limitación guiándose sólo por las instrucciones que se derivan de la fecha de entrega del pedido? No es una pregunta con truco, es una cuestión muy real.

¿De verdad? Si hacemos caso a los consejos tan peculiares que hemos desarrollado, no programar los recursos no limitados, entonces, no importa que nos guiemos por una u otra fecha de cumplimiento, ¿no? Cualquier material activará el trabajo en el recurso adecuado en el momento en que esté disponible después de la limitación, ¿por qué tenemos que preocuparnos?

En efecto, sucederá así, siempre que nos ocupemos de dos tipos de situación. La primera es bastante obvia: dijimos que debíamos dar un programa a los recursos no limitados que usen piezas comunes. ¿Qué debemos hacer aquí? En esta situación está clara la fecha que tenemos que seguir. ¿Para qué tenemos que dar un programa a esos recursos? Porque tememos que se roben piezas, tememos que usen material común para un pedido cuando la intención era usarlo para otro. Por lo tanto, en el caso de los recursos que consumen directamente piezas comunes, tendremos que generar un programa basado en las fechas de entrega de los pedidos. No hay otra alternativa.

Esta era una de las situaciones. ¿Cuál era la otra? Esperemos que nos lleve a la misma conclusión porque, si no lo hace, estaremos atascados. El otro tipo de situación que tenemos que tratar es cuando el pedido es para un producto que supone el ensamblaje de muchas piezas, de las que sólo una requiere la participación de un recurso limitado. Supongamos que el recurso limitado se ha programado para que haga esa pieza antes de lo

necesario, debido a sus limitaciones de capacidad. En ese caso, el hecho de que tengamos que hacer una pieza con antelación no es razón válida para que se fabriquen y ensamblen todas las demás. El cliente no quiere aceptar el pedido antes de la fecha de entrega. ¿Qué hacemos? Es evidente que, también en este caso, debemos derivar las fechas de las demás operaciones de acuerdo con la fecha de entrega del pedido que estén sirviendo, y no de acuerdo con la fecha del bloque que las alimenta.

¿Ya lo hemos cubierto todo? Vamos a comprobarlo. Hemos cubierto la operación que alimenta a un recurso limitado; hemos cubierto las operaciones que alimentan pedidos libres; hemos cubierto las rutas rojas, las operaciones que hay entre los bloques y los pedidos. ¿Se nos ha escapado algo? Ah sí, debemos tratar de las operaciones que ni alimentan a un recurso limitado ni a un pedido libre. ¿Hay operaciones de este tipo? Sí, y tienen que ver con la situación que acabamos de analizar, un ensamblaje que sólo requiere algunas de su piezas de un recurso limitado. ¿Qué pasa con las operaciones que se necesitan para el resto de las piezas? Estas operaciones no están en una ruta roja, pero tampoco están alimentando un pedido libre.

¿Cuál es el problema? Acabamos de decir que en esas operaciones debemos seguir la fecha de entrega del pedido correspondiente. Correcto, pero ¿cómo la vamos a seguir? ¿No deberíamos usar el *buffer* de envío para estos casos? Sí, por supuesto. Primero debemos derivar la fecha de ensamblaje restando el *buffer* de envío de la fecha de entrega del pedido. Pero esto no es suficiente, tenemos que garantizar que las piezas no limitadas llegan antes que las piezas que proceden de los recursos limitados.

Por lo tanto, tenemos que hacer otra resta, reduciendo el *buffer* de ensamblaje de la fecha de ensamblaje que habíamos calculado. La fecha resultante es la que usaremos para todas estas actividades y, en particular, para el lanzamiento del material de las piezas no limitadas.

La figura 34.1 nos muestra el caso del pedido de un producto que requiere más de un nivel de ensamblaje y más de un operación realizada por un recurso limitado. La fecha de entrega del pedido y las fechas de los bloques ya han sido establecidas en el paso del *drum*, todas las demás se han derivado de acuerdo con lo que se ha dicho hasta ahora. Verifíquenlo, por favor.

En todos los casos que hemos examinado hasta ahora, la respuesta era la misma: el programa de una actuación se debe derivar de acuerdo con la fecha de la limitación a la que alimenta esa operación. Una simetría tan buena tiene que ser válida. Lo que hemos descrito hasta ahora es el procedimiento DBR manual, un procedimiento que ha demostrado su validez en muchas implantaciones prácticas. Desgraciadamente, está lejos de ser suficiente para nuestro propósito. No sólo queremos un programa, nuestro propósito principal es la identificación de todas las limitaciones de la empresa.

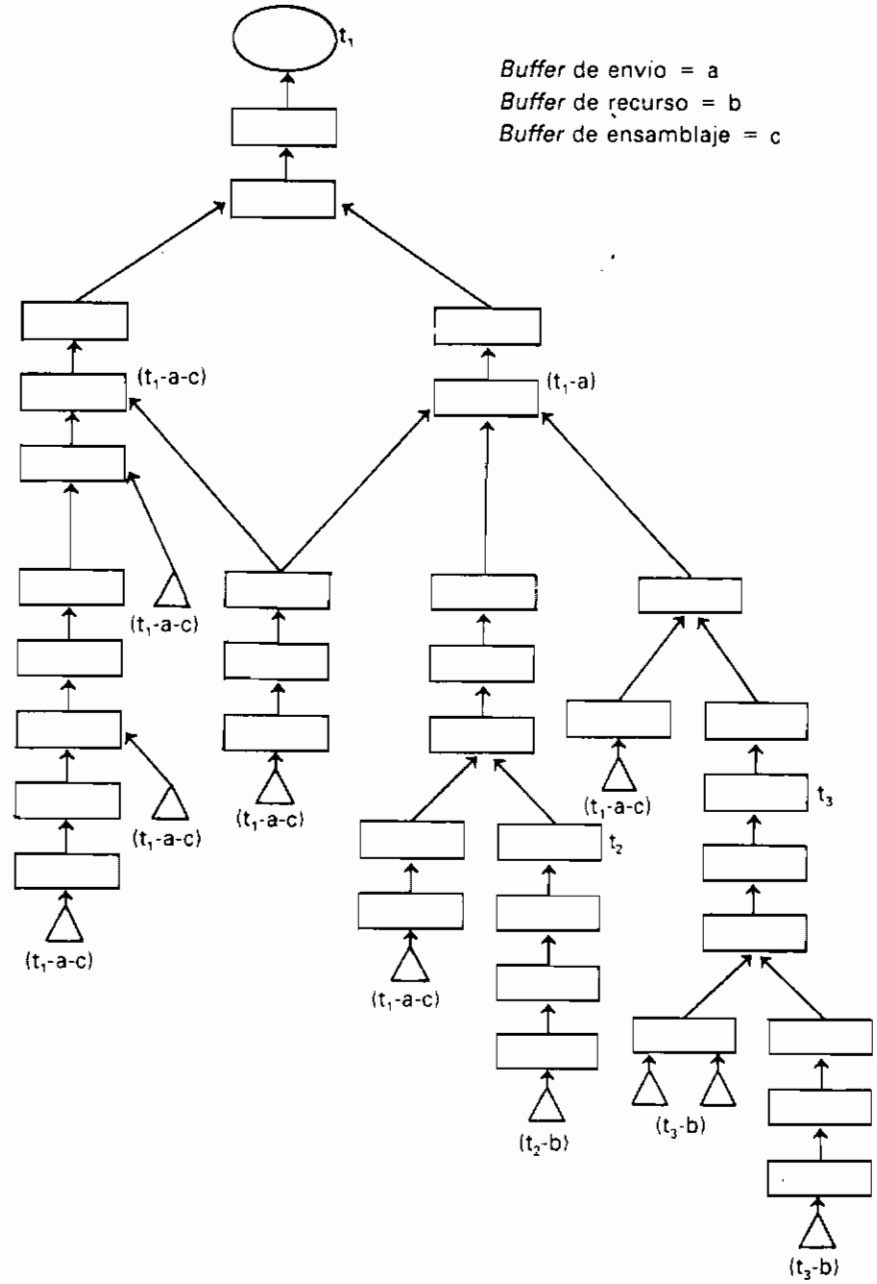


Fig. 34.1. *Las fechas de lanzamiento de los materiales se derivan de las fechas del «drum» y de los tamaños de los buffers de tiempo.*

Es más, queríamos generar un programa que no tuviera ningún conflicto conocido entre las limitaciones identificadas. Un sistema de información no puede permitirse dejar la eliminación de conflictos en manos del personal de planta, es demasiado tarde.

Si seguimos el procedimiento que hemos descrito, ¿cómo se manifestarán los choques con la realidad? Sí, esta es la pregunta que debemos comprobar si hay choques con la realidad. Así pues, una vez más, ¿cómo se van a manifestar esos choques? ¡En forma de pedidos de lanzamiento de material en el pasado! ¿Cómo vamos a resolver este tipo de conflicto? La única manera parece ser posponiendo las órdenes correspondientes en el *drum*. Esto, por definición, supone posponer un pedido. Parece que en esta fase la única forma de eliminar los conflictos es reduciendo la generación de valor. ¿Estamos seguros de que hemos hecho todo lo posible para eliminar o, al menos, reducir este perjuicio? ¿Cómo puede ser que tengamos atadas las manos para eliminar los conflictos, aun estando dispuestos a pagar en inventario y en gasto operativo? En el recurso limitado pudimos hacer mucho para eliminar los conflictos sin tener que pagar en valor generado, pero, cuando llegamos a la subordinación, cuando estamos tratando con los recursos no limitados, de repente, resulta que no podemos hacer nada. Suena raro. Tiene que haber una forma de hacerlo mucho mejor. Probablemente hemos desechado oportunidades de mejora por culpa de parámetros o suposiciones que tendremos que revisar y deshacer.

35

Los recursos no limitados. Subordinación y capacidad. El enfoque conceptual

¿Dónde podemos empezar a buscar la llave de esta situación tan intolerable? Vamos a verlo cuidadosamente. Intentemos reunir todos los puntos que están abiertos, todos los que nos parezca que hemos dejado en una situación insatisfactoria.

Se suponía que íbamos a identificar todas las limitaciones del sistema. Hemos dicho una y otra vez que, probablemente, hay más de un recurso limitado. Puede que sea otro cuello de botella o, al menos, otro recurso que no tiene suficiente capacidad de protección. Dijimos que los íbamos a cazar uno a uno durante el ciclo de identificar, explotar y subordinar. Y henos aquí, al final de la subordinación, con un montón de conflictos. Pero son conflictos con el tiempo, con la necesidad impracticable de hacer cosas en el pasado. No podemos usar estos conflictos para identificar la siguiente limitación de capacidad; no son conflictos entre carga y capacidad disponible.

¿De qué nos sorprendemos? Durante la fase de subordinación no se ha considerado en absoluto la capacidad. ¿Y por qué no? ¿No se suponía que en esa fase debíamos ignorar la capacidad? ¿No debíamos tener en cuenta la capacidad sólo cuando supiéramos que faltaba? Probablemente, no.

Parece que todo el proceso de subordinación gira alrededor de restar los diversos *buffers* de las fechas que ha establecido el *drum*. Los únicos datos adicionales que usamos son la longitud de los *buffers*. ¿Tiene la capacidad algo que ver con la longitud de los *buffers*? ¡Eso es! Esa parece ser la clave. Nos hemos tropezado con otro punto que habíamos dejado abierto. ¿Recuerdan que, cuando discutimos los factores que determinan la longitud de un *buffer*, dijimos que la capacidad era uno de los dominantes? ¿Qué nombre le pusimos? Disponibilidad no instantánea de los recursos, o algo así. ¿De qué recursos estábamos hablando? De recursos no limitados. Sí, desde luego que esta es la clave de nuestro problema. Todo encaja.

Primero vamos a refrescarnos la memoria. No sería justo pretender que nos acordáramos con exactitud de algo que se llama «disponibilidad no instantánea de los recursos». ¿Cuál era el problema? Ah sí, comentábamos el hecho de que el tiempo real de proceso contribuía muy poco al *lead-time* global. Ahora, empezamos a recordarlo. Dijimos que, aunque tengamos suficiente capacidad media, puede que el trabajo tenga que esperar en la cola cuando llegue el recurso, porque éste puede estar ocupado con algún otro trabajo necesario.

¿Qué podemos hacer al respecto? Probablemente mucho. Tenemos todos los datos necesarios. Quizás podamos conseguir una buena estimación de cuándo van a ocurrir estas sobrecargas temporales. Si podemos conseguir esto y tenerlo en cuenta, quizás podamos tener en cuenta la capacidad durante el proceso de subordinación. Vamos a intentar ver cómo lo podemos hacer en la práctica. No tenemos nada que perder, de todas las maneras, estamos metidos en un buen lío.

No hay nada que nos impida calcular las cargas diarias de cada recurso no limitado. Es un cálculo bastante directo, tal y como lo ha estado haciendo MRP durante años. Tenemos la disponibilidad diaria del recurso, ya que tenemos las fechas. Si dividimos una lista por la otra obtendremos una indicación clara de cuándo y dónde van a ocurrir las sobrecargas. Es verdad que será sólo una aproximación; sabemos que el trabajo tenderá a llegar más tarde, debido, principalmente, a estas mismas sobrecargas, pero, ciertamente, será una aproximación muy útil y muy válida.

¿Útil? ¿Cómo? ¿Qué vamos a hacer con el cuadro resultante, que, probablemente, se parecerá al horizonte de Manhattan? Y, por favor, no me digan que nos va a proporcionar una mejor introspección. Cuando estamos trabajando con un ordenador, mejor introspección equivale a «no sabemos qué hacer con esto». ¿Qué va a hacer un ordenador con una mejor introspección?

Vaya, ahora queremos devolverle la pelota al usuario. ¿No pretenderemos que sea el usuario el que traslade los picos a los valles? ¿Cómo iba a hacer el usuario un milagro así? MRP intentó hacerlo bajo el «sofisticado» nombre del «bucle cerrado». Me gustaría ver, siquiera, una implantación que haya tenido éxito saliendo de ese bucle. ¿No nos damos cuenta de que si retrasamos un pico de carga en un recurso, necesitaremos hacer los cambios correspondientes en los demás? Se alimentan uno a otro. Puede que solucionemos un pico, pero, probablemente, crearemos otros, quizás mayores, en los demás recursos. Todos sabemos que este «bucle cerrado» es un «bucle sin fin»; este proceso de iteración, simplemente, no converge.

Bien, nos hemos desahogado un poco, pero está claro que hay una buena solución escondida en alguna parte de esta área, así que, vamos a buscarla... pero despacio, sin duda está profundamente escondida en un campo de minas.

Vamos a considerar uno de estos picos de carga en un recurso. Es evidente que no lo podemos dejar ahí; no podemos hacer que un recurso trabaje por encima del 100 por 100 de su capacidad. No estamos tratando con un retorcido sistema de primas, donde todos trabajan regularmente al 130 por 100; estamos trabajando con nuestra mejor estimación. Entonces, ¿qué vamos a hacer con este pico de carga? Por favor, no se atrevan ni a pensar en las horas extras. No olvidemos que estamos tratando con recursos no limitados. ¿Debemos utilizar horas extras para aumentar la capacidad de unos recursos que ya tienen más que suficiente? ¡Qué idea tan ridícula!

Es obvio que tenemos que intentar trasladar el pico a algún valle cercano. Si estamos tratando con un recurso no limitado, tiene que haber valles suficientes. No, no debemos buscar valles hacia adelante en el tiempo; ¿qué íbamos a hacer con un valle así? Si trasladamos el pico en esa dirección estaremos posponiendo pedidos, estaremos perjudicando el valor generado. Debemos mover el pico hacia el pasado, hacia atrás en el tiempo.

Un momento, si tocamos el pico, en cualquier dirección que lo movamos, ¿no necesitaremos mover también las cargas de los otros recursos que lo alimentan?

Sí, pero, ¿cuál es el problema? De lo que hemos dicho ahora se puede concluir, lógicamente, lo siguiente: debemos trasladar los picos hacia el pasado, y debemos hacerlo de forma que garanticemos que las cargas resultantes se podrán hacer en todos los recursos. Por lo tanto, está clarísimo que no debemos mirar los perfiles de carga después de haber terminado todo el programa. Es demasiado tarde.

En vez de eso, al construir el programa debemos tener cuidado de movernos hacia atrás en el tiempo de forma constante. No se debe programar ninguna operación en una fecha específica hasta que se haya terminado con todas las demás operaciones que tienen que programarse en fechas posteriores. Esta es la única forma de garantizar que si movemos un pico hacia una fecha más temprana, las operaciones que lo alimentan caerán en un valle en vez de caer en un pico aún más alto. También implica que tenemos que recoger los datos de carga de cada recurso, y hacer los traslados necesarios, no después de, sino durante la construcción del programa para los recursos no limitados.

Según lo que se ha dicho, parece muy sencillo. No es que entienda del todo lo que me están diciendo, pero, ¿no ha programado MRP siempre hacia atrás en el tiempo? Si estamos en lo cierto, ¿cuál es el problema para que MRP calcule las cargas mientras programa, en vez de terminar primero con el programa y hacer después los cálculos de carga? ¿Por qué no han hecho ellos lo mismo? Probablemente no es tan fácil. Probablemente es

mucho más complicado de lo que se está diciendo... y, por lo tanto, sospecho que estamos equivocados.

Bien, vamos a examinarlo detalladamente. De todas formas, todavía lo tenemos que convertir en un procedimiento práctico, hasta ahora sólo hemos hecho una descripción somera. Pero, antes de hacerlo, contésteme: ¿por qué se ha dicho que MRP programe hacia atrás? Sí, ya sé que todo el mundo piensa así, pero eso no quiere decir que sea verdad. Tal y como yo entiendo lo que hace MRP, y déjenme que elija uno de los procedimientos más simples (de todas formas, todos llevan exactamente a la misma respuesta), la forma en que MRP se mueve por el eje del tiempo es... Será mejor que ponga un ejemplo.

Imaginemos que, en la estructura, sólo tenemos un producto que consiste en el ensamblaje de dos piezas. Para este producto, tenemos muchos pedidos con distintas fechas de entrega. ¿Cómo se moverá MRP sobre el eje de tiempo cuando programe esta empresa? Debe empezar con el pedido más temprano, si no MRP tendrá problemas para asignar los inventarios existentes. ¿No está claro? Bien. Supongamos que MRP ha empezado con el pedido más tardío, recordemos que MRP asigna los inventarios en proceso mientras hace la programación. Intenten imaginar la reacción del usuario cuando se dé cuenta de que los inventarios que ha fabricado para un pedido urgente se han asignado, de repente, a un pedido que tiene que entregarse dentro de un mes. Además, el estúpido ordenador le está pidiendo que se dé prisa con la producción de ese mismo inventario.

Así que MRP empieza con el pedido más temprano. Por favor, pongan su dedo sobre el punto apropiado del eje de tiempo, sobre la fecha de entrega de este pedido. Ahora MRP empieza a hacer la explosión de necesidades por la estructura del producto, al hacerlo se mueve hacia atrás en el tiempo. Por favor, muevan el dedo de acuerdo con esto. Debe elegir una de las dos rutas, una de las dos piezas que hay que fabricar. Elige una y continúa moviéndose por ella, y su dedo continuá moviéndose hacia la izquierda. Hasta ahora nos estamos moviendo hacia atrás en el tiempo, no hay problema. MRP llega a la materia prima y se ocupa de ella: ¿qué más? Todavía tenemos que programar otra pieza, así que va al ensamblaje y empieza a meterse en la segunda ruta. ¡Un momento, no tan deprisa! ¿Qué debe hacer ahora su dedo? Volver al ensamblaje significa moverse hacia adelante en el tiempo, y meterse después en la segunda ruta significa moverse hacia atrás. El dedo ha hecho un zig-zag maravilloso sobre el eje de tiempo. Ya hemos terminado con este pedido. ¿Ahora qué?

¿Por qué no bailamos un tango? Si vamos al siguiente pedido, el dedo se mueve más hacia la derecha. Nos metemos en una pieza, ahora hacia la izquierda, por favor. Ahora hacia ensamblaje, otra vez a la derecha; a la segunda pieza, a la izquierda; al siguiente pedido, más hacia la derecha... y

así sucesivamente. ¿A esto le llamamos hacia atrás en el tiempo? A mí me parece un zigzag en el tiempo, con una fuerte tendencia a moverse continuamente hacia delante.

¿MRP programa hacia atrás? Seamos serios. ¿Cómo podemos diseñar un procedimiento que se mueva hacia atrás en el tiempo constantemente, al mismo tiempo que nivela las sobrecargas?

36

Buffers de tiempo dinámicos y capacidad de protección

Ir constantemente hacia atrás en el tiempo nos obliga a empezar con lo último que ibamos a hacer, lo que ibamos a hacer más tarde. Si empezamos en cualquier otro punto, llegará un momento en que tendremos que ir hacia adelante. Lo último pedido que tengamos antes de la fecha del horizonte, más un *buffer* de envío. ¿De acuerdo?

No, ése no es el sitio en el que hay que empezar. Si lo hacemos así, los inventarios se asignarán a los últimos pedidos. Tenemos que empezar asignando los inventarios.

¿No hicimos esto ya cuando generamos los bloques del programa? Sí, pero en ese momento sólo asignamos los inventarios que estaban en las líneas rojas. En esa fase sólo hacía falta asignar esa parte, y no tenía sentido asignar los inventarios que hubiera en la otra parte. Recordemos que los estamos asignando de acuerdo con las fechas de entrega de los pedidos, al que llega primero se le sirve primero. Cuando autorizamos el programa, seguramente cambiamos algunas fechas de entrega y, por tanto, ahora es el momento de terminar con la asignación.

No se tardará mucho en hacer esto. Como todos los datos relevantes están en la memoria *on-line*, se pueden dar esos pasos a una velocidad de vértigo. Si queremos seguir entreteniéndonos, tendremos que pensar en alguna idea mejor, más «larga».

Pero, ¿qué pasa con la longitud del *buffer?*

Dijimos que una de las razones dominantes del tiempo total de proceso de los trabajos era la disponibilidad no instantánea de los recursos no limitados. Ahora vamos a tener esto en cuenta. ¿No cambiará esto la elección inicial del usuario en cuanto a la longitud adecuada de los diversos *buffers?* ¿No deberíamos discutir esto antes de empezar a subordinar? Recordemos que el proceso de subordinación está fuertemente basado en la longitud de los *buffers*.

Ciertamente hay algo de eso. ¿Cuál es, de verdad, el significado de pasar la carga de un pico a un valle anterior? En palabras menos coloristas, significa simplemente que, por consideraciones de capacidad, tendremos que programar antes un trabajo que nos habría gustado programar en una fecha más tardía. ¿Es esta la única actividad cuya fecha se ve afectada? No, todas las actividades que la alimentan tienen que moverse de acuerdo con ésta, incluyendo la fecha de lanzamiento del material correspondiente. El resultado final del tratamiento de los picos de carga es que lanzamos los materiales antes de la fecha definida por la longitud del *buffer*. Esto equivale a incrementar la longitud original del *buffer*, pero sólo para estas actividades.

Pues sí, este mecanismo va a considerar, con bastante exactitud, el impacto que tiene la disponibilidad no instantánea en el tiempo total de proceso (*lead-time*) del trabajo. Intentemos digerir esta sorpresa tan agradable. Uno de los problemas más difíciles con los que hemos estado luchando era la estimación de los tiempos de cola individuales. Todo el que haya intentado implantar MRP sabe que la estimación de los tiempos de cola es una lucha constante entre el personal de producción y el de gestión de materiales.

Las emergencias constantes y los cambios de prioridades hacen sospechar que se han subestimado los tiempos de cola. La presión que existe para reducir el tiempo total de proceso (*lead-time* global) nos está obligando a reconocer que, cuando se agregan, se han sobreestimado. En algunas industrias de armamento nos encontramos con que los tiempos de cola se han fijado en una semana por operación, y no tienen relación alguna con la duración de la actividad. A pesar de ello, las faltas de material en ensamblaje se toman casi como un *fait accompli*. A pesar de las continuas discusiones, todos los afectados saben muy bien que las estimaciones de tiempos de cola en un determinado centro de trabajo sólo son conjeturas sin base. En la ingeniería de diseño la situación es aún peor. Aquí los tiempos de cola se mezclan con los tiempos de ejecución, lo que lleva a una desconfianza total sobre cualquier cálculo de tiempo.

No es de extrañar. Los tiempos de cola no están en función de los trabajos individuales que se van a realizar, están principalmente, en función de la carga que se asigna al recurso. El flujo de los pedidos no es uniforme, y la mezcla de productos puede cambiar constantemente. En consecuencia, la carga que afecta a cada tipo de recurso puede fluctuar considerablemente: durante dos días un recurso determinado trabaja desesperadamente y, al día siguiente, casi no tiene nada que hacer. El intento de asignar un número al tiempo de cola es un intento de representar esta entidad dinámica con un «tiempo promedio de cola» imaginario. El intento de representar una entidad que fluctúa acusadamente con un número

que representa una media, sólo nos puede conducir a unos resultados muy insatisfactorios.

No hay duda de que mejoramos considerablemente cuando usamos *buffers* de tiempo en vez de tiempos de cola individuales. El concepto habitual de tiempo de cola intenta realizar el promedio al nivel del recurso: cuánto tiempo, de promedio, tiene que esperar un trabajo delante de ese recurso. Esto no nos sirve. La capacidad excedente de mañana no me sirve para hoy, ni tampoco el exceso de capacidad de ayer, cuando el trabajo todavía no había llegado. Pero, sin duda, sí que nos sirve hacer el promedio al nivel del trabajo, usando el concepto de *buffers* de tiempo y de la aceleración de trabajos.

Lo que estamos proponiendo aquí parece ser la próxima fase de mejora. Podemos predecir, con bastante fiabilidad, las fluctuaciones de cargas que esperamos tener. Están definidas por los cambios dados en las limitaciones del mercado, modificados por las limitaciones internas. Lo que sugerimos es programar el lanzamiento de materiales de acuerdo con las fluctuaciones de cargas que esperamos tener. Esto reducirá, sin duda, el tiempo que tiene que esperar el trabajo delante del recurso y, por tanto, reducirá significativamente el tiempo total de proceso.

Lo único que queríamos era un mecanismo para identificar las limitaciones de la empresa. Hemos conseguido, como beneficio añadido, otra disminución significativa del inventario. Sin duda, esto es señal de que vamos por buen camino.

Sí, tendremos que reducir considerablemente los cálculos originales de los *buffers* de tiempo. Lo que tenemos que dar a nuestro sistema de información no es el cálculo del tiempo total de proceso, sino algo más pequeño. Sólo tenemos que calcular el impacto que tiene en el tiempo total de proceso el «Murphy puro»; podemos ignorar el impacto de la disponibilidad no instantánea. Esta última parte la va a hacer el propio sistema, caso a caso. Lo que en realidad tenemos que dar al sistema es la parte «fija» del *buffer* de tiempo. El sistema tendrá que calcular la parte variable. Lo que vamos a hacer es usar un *buffer dinámico*.

Una nota al margen. En realidad, tendríamos que retroceder y volver a escribir toda la explicación del proceso de subordinación, usando los términos «parte fija» y «parte variable». Por suerte, resulta que podemos continuar tranquilamente sin crear confusión alguna. Donde hacíamos referencia al *buffer* de tiempo, se debe entender que nos referíamos solamente a la parte fija del mismo.

Es cierto que la «gestión del *buffer*» va a modificar continuamente los cálculos originales de la longitud del *buffer* de tiempo, pero, ¿cómo podemos llegar a una estimación inicial? Tenemos que empezar en algún sitio, y todavía no hay nadie que tenga la experiencia necesaria para evaluar

sólo una parte del tiempo total de proceso: la parte que proviene del Murphy puro. En este punto mi consejo depende del nivel de experiencia real que tenga el usuario en la implantación de los *buffers* de tiempo. Si ya tiene una implantación manual de DBR, puede dejar la longitud del *buffer* en la mitad. Si no es así, puede calcular el promedio del tiempo total de proceso de los trabajos, y dividirlo por cinco. Esto nos proporcionará un punto de arranque satisfactorio.

Verán, esta nivelación de los picos de trabajo va a provocar que un recurso está a plena carga durante unos períodos de tiempo bastante largos. Básicamente, puede que se programe a un recurso para que trabaje al 100 por 100 de su capacidad disponible durante muchos días. Es obvio que durante este intervalo de tiempo este recurso no tendrá capacidad de protección; toda su capacidad disponible se dedica a la producción. ¿Durante cuánto tiempo, durante cuántos días vamos a permitir que un recurso no limitado se quede sin capacidad de protección, antes de admitir que tenemos un problema?

Vaya pregunta... Pero, es cierto, tenemos que abordar este tema. Veamos... Un recurso necesita tener capacidad de protección para restaurar el daño causado por las perturbaciones, no sólo en ese recurso, sino, también, en todas las actividades que lo alimentan. ¿De qué tipo de daño estamos hablando en este caso? Básicamente, del que resulta de dejar sin protección a las limitaciones.

En cualquier momento dado la limitación sólo está protegida por el contenido del «material» que se encuentra en el origen del *buffer*. Recordemos que este inventario no tiene que ser físico; en el caso de una limitación de mercado puede tomar la forma de envíos adelantados; en el caso de ingeniería puede ser un plano o unos datos necesarios. Hemos subrayado que la protección no la proporciona cualquier inventario; tiene que ser el inventario adecuado, los «materiales» que están programados para que los consuma la limitación.

Ahora intentemos crearnos el siguiente cuadro mental. Olvidemos el tema de las sobrecargas temporales, de ésas ya nos encargamos por otro lado, aquí estamos tratando de las perturbaciones normales. Consideremos un trabajo que se ha programado para que lo haga la limitación en algún punto entre el momento actual y el momento más tardío de un *buffer* de tiempo. Si este trabajo no está ahora en el origen del *buffer*, lo vamos a llamar un «agujero en el origen del *buffer*».

Intentemos seguir el viaje típico de un agujero de este tipo. Supongamos que ahora hay un *buffer* de tiempo antes de la fecha programada para el trabajo, empieza a penetrar en el origen del *buffer* un agujero. A medida que pasa el tiempo, si no llega nuestro trabajo, el agujero continúa penetrando más y más profundamente en el origen del *buffer*. Pronto cruza la

zona de seguimiento y entra en la zona de aceleración. Si sigue faltando, la limitación tendrá que desviarse de su plan. Su rendimiento se verá dañado.

Volvamos al tema de la capacidad de protección teniendo en cuenta este cuadro. Examinemos un recurso no limitado trabajando al 100 por 100 en un día determinado. Ese día este recurso no tiene capacidad de protección, lo que supone que los trabajos se atrasarán (como promedio) en su llegada al origen del *buffer* debido a las perturbaciones. Los agujeros penetran en el origen del *buffer*. ¿Cuánto? Bueno, depende. ¿De qué? Debe depender del nivel de capacidad de protección que necesitaba el recurso y de la longitud del *buffer*.

¿Cómo podemos entender mejor este tema? Puede que debamos intentarlo con algunos números. Supongamos que un recurso necesita un 5 por 100 de capacidad de protección. Esto significa que debe tener libre el 5 por 100 de su tiempo para ayudar a que la empresa se recupere del impacto de las perturbaciones. En otras palabras, quiere decir que se necesita el 5 por 100 de capacidad de protección para que el recurso no «contribuya» a la progresión de los agujeros en el origen del *buffer* al que alimenta. Si el recurso no tiene capacidad de protección durante un día, la penetración adicional del agujero en el origen del *buffer* será del 5 por 100 de un día. Esto es un promedio, por supuesto. No estamos tratando con sucesos determinísticos, sino estadísticos, las perturbaciones. Ahora, la respuesta es obvia. Cada día consecutivo que esté a plena carga el recurso, el agujero progresará una fracción de día equivalente al porcentaje de la capacidad de protección necesaria. ¿Cuántos días consecutivos podemos permitir que trabaje a plena carga un recurso no limitado? Tenemos que decidir cuánto vamos a permitir que penetren los agujeros sin que lo consideremos como un problema. Vamos a decidir que sea medio *buffer;* más de eso sería como jugar a la ruleta rusa con cinco balas.

Esto supone que, cuando subordinemos, tendremos que tener la precaución de no dejar que un recurso trabaje demasiado tiempo antes de darle algún tiempo no programado. Básicamente tenemos que hacer un seguimiento constante de lo que va a hacer la carga de cada recurso con nuestros agujeros imaginarios. Pero, como entendemos la relación que hay entre la capacidad de protección y la longitud del *buffer*, esto se convierte en un trabajo de programación relativamente simple.

¿Queda algo más?

37

Algunos temas residuales

¿Podemos empezar a construir ya el procedimiento de subordinación? Todavía no, todavía tenemos algunos puntos que hay que refinar.

Cuando hablamos de nivelar los picos hacia atrás en el tiempo, ¿cómo vamos a tratar un pico creado por una actividad de una línea roja? Si pasamos esa actividad a una fecha anterior, también tendremos que mover el bloque correspondiente, es decir, la actividad que hace la limitación. Yo pensaba que subordinar no significaba modificar, sino seguir.

Bien, es evidente que los picos creados por una actividad en línea roja, «picos de línea roja» para abreviar, necesitan una atención especial.

Primero tenemos que comprobar si esta situación se da de verdad en nuestro pico de línea roja, en otras palabras, si la nivelación del pico crea la necesidad de mover el bloque correspondiente. Esto sólo ocurrirá cuando no haya «holgura», cuando la fecha del bloque no sea anterior a la fecha de envío menos al *buffer* de envío. Como ya dijimos, o al menos indicamos, este no es siempre el caso. A veces, el bloque está situado mucho antes debido a un pico de carga en la limitación. Otras veces, un pedido genera más de un bloque, y si la fecha de entrega del pedido se cambia debido a la existencia de un bloque rojo, todos los demás bloques tendrán «holgura».

Si hay suficiente holgura, el pico de línea roja no requiere un tratamiento especial. Si no la hay, tendremos que ocuparnos de él.

Primero, recordemos que en estos casos extremos nos conformamos con la mitad del *buffer*. Si dejamos el pico como está, este trabajo generará un agujero en el origen de *buffer;* por cada hora sobre la capacidad disponible (en la fecha del pico) el agujero penetrará una hora en el origen del *buffer*. Esto equivale, de hecho, a nivelar hacia adelante, sacrificando la protección.

Supongamos que hemos usado este margen al máximo y todavía tenemos el agujero. Ahora usamos el permiso para hacer horas extras, si está disponible. Si esto tampoco ayuda, avisamos al usuario. El usuario tendrá

que elegir entre autorizar horas extras adicionales (el sistema tendrá que decir cuántas más hacen falta), desviar el trabajo a otro recurso (el sistema tendrá que decir cuál es la cantidad mínima que hay que desviar), o retrasar el pedido (el sistema tendrá que decir hasta qué fecha). ¿Hay más posibilidades? Puede que el usuario sepa que el problema ha ocurrido porque los datos son erróneos, y puede que decida indicar al sistema que ignore el pico y que continúe.

Pero, ¿qué sucede si es un problema real? ¿Qué puede hacer el usuario si ninguna de las posibilidades que se han mencionado son factibles? Entonces, parece que tenemos una limitación adicional, ¿no es sí? El recurso que tiene el pico se debe considerar como limitación, y el sistema debe abandonar todo intento de continuar en esta fase. Hemos conseguido un objetivo intermedio: hemos identificado otra limitación. El sistema debe pasar a la siguiente fase, resolver los conflictos entre las limitaciones identificadas.

Parece que esto suena bien. Pasemos a otro punto pendiente. Me gustaría preguntar cómo tenemos en cuenta el tiempo que hace falta para realizar la operación en sí. Entiendo que estamos teniendo en cuenta el impacto que tienen los tiempos de proceso sobre el tiempo total de proceso (*lead-time*), a través de las cargas que suponen para los recursos. También entiendo, y estoy de acuerdo, que la contribución directa del tiempo de proceso es bastante pequeña, pero aun así, los tiempos de proceso contribuyen en alguna medida. Si tenemos los datos necesarios, ¿por qué no tenemos en cuenta esta contribución directa?

Veamos. Lo que se sugiere es calcular el tiempo que se tarda en hacer un lote en cada operación, multiplicando el tiempo de proceso unitario por el número de unidades requeridas, añadir el tiempo de cambio y, después, sumar los resultados a lo largo de la secuencia de operaciones necesarias para hacer el trabajo. Observen lo que sucederá si seguimos esta sugerencia.

Supongamos que estamos tratando con una línea industrial. Supongamos que la línea tiene diez centros de trabajo distintos, y que el tiempo de proceso de una unidad varía de menos de un minuto a más de dos minutos entre los distintos centros de trabajo. Ahora tomemos un caso en el que el pedido es de varios cientos de unidades. ¿Qué conseguiremos si seguimos esa sugerencia? Como el tiempo de proceso de todo el pedido en cada centro de trabajo es, aproximadamente, de un día, obtendremos la impresión de que, aunque no haya perturbaciones, tardaremos en terminar el pedido, aproximadamente, una semana. Es ridículo, sabemos que sólo se tarda un día, más o menos.

Si tenemos en cuenta el tiempo necesario para hacer todo un lote en cada operación, estamos ignorando la posibilidad de solapar los lotes entre

las distintas operaciones. Como en el caso de una línea de montaje, aunque se tarde un día en terminar el pedido en un centro de trabajo, eso no impide que el siguiente centro de trabajo empiece a trabajar en cuanto esté terminada la primera unidad. Como pueden ver, si nos preguntamos cuál es la contribución directa de los tiempos de proceso, resulta que es aún más pequeña de lo que pensábamos. Si, como en el caso de una línea de montaje, no hay problemas de disponibilidad de recursos, y Murphy no existe, entonces el tiempo que se tarda en terminar el pedido es casi igual al tiempo requerido para procesar el pedido en un solo centro de trabajo: el que tenga el tiempo de proceso más largo. Si queremos ser aún más precisos, el tiempo que se tarda en hacer el pedido es igual al tiempo que se tarda en hacerlo entero en la operación más larga, más la suma del tiempo que se tarda en hacer una unidad en cada una de las demás operaciones. Esto siempre resulta ser ridículamente pequeño comparado con la disponibilidad no instantánea y con Murphy. ¿Por qué nos tenemos que molestar en calcularlo sólo porque los datos estén disponibles?

Lo que se ha dicho no suena mal, pero se está presumiendo la capacidad de solapar actividades a lo largo de todas las operaciones necesarias para realizar el trabajo. Esto no siempre es posible. No me estoy refiriendo a entornos en los que, sin duda, es posible, pero en los que está prohibido por alguna política ridícula que se escuda en la falsa pretensión de un mejor control. No, en esos entornos deberían elevarse las limitaciones políticas. Me refiero a situaciones en las que el solape no es práctico por limitaciones técnicas. Por ejemplo, un horno que funcione por lotes enteros, en el que una vez que entra el lote y se cierra la puerta, todas las unidades del lote tienen que esperar hasta que se termine el proceso y se abra la puerta. El transporte es otro caso en el que de ninguna manera podemos dedicar una camioneta a cada unidad. En los casos en los que se procesa junto todo el lote, técnicamente no existe la posibilidad de solapar.

Así pues, la contribución del tiempo real de proceso al tiempo total de proceso (*lead-time*) es el tiempo necesario para procesar el lote en la operación más larga, más el tiempo necesario para procesar el lote en las operaciones que no se pueden solapar, más el tiempo de proceso de una unidad en el resto de las operaciones. Tenemos todos los datos necesarios, ¿por qué no los usamos?

Bien, si insisten. ¿Podemos empezar ya con el mecanismo de subordinación, o hay algo más?

Una última pregunta, si aún le queda paciencia. ¿Debemos intentar ahorrar tiempos de cambio durante la fase de subordinación?

¿Por qué íbamos a tener que hacerlo? ¿Cuál es el propósito de ahorrar tiempos de cambio? ¿Qué es lo que en realidad estamos ahorrando? No es

dinero, sino tiempo. ¿Podemos hacer algo con este tiempo? ¿Nos ayudará a incrementar los ingresos netos? Los ingresos netos vienen determinados por las limitaciones de la organización, y aquí sólo estamos tratando con los recursos no limitados. Si un recurso no tiene suficiente capacidad por culpa de los tiempos de cambio, lo consideramos como limitación y tratamos los tiempos de cambio consecuentemente. Así que, ¿por qué nos tenemos que preocupar de los tiempos de cambio durante el proceso de subordinación?

Los ingresos netos no son lo único que afecta a nuestro resultado. También tenemos que considerar, aunque en menor medida, el inventario y los gastos operativos. Veamos si esto nos obliga a considerar el ahorro de tiempos de cambio, incluso en la fase de subordinación.

Siempre que intentamos ahorrar tiempo de cambios tenemos que «saltar» un lote que, de no ser así, se haría en un futuro más remoto. Al ahorrar cambios aumentamos el inventario, y las consideraciones sobre inventario nos aconsejan alejarnos de los intentos de ahorrar tiempo de cambios.

A menos que la contribución del tiempo de cambio sea tan grande que cree una limitación, por supuesto. Si este es el caso, tenemos que pagar con inventario para proteger de las perturbaciones a la limitación. Decidamos hacer dos cosas. La primera es intentar ahorrar cambios antes de considerar una segunda limitación en los recursos. Recordemos que al identificar el primer recurso limitado utilizamos el mínimo tiempo de cambios (un cambio por cada código de pieza/operación, no por cada pedido), que es equivalente al máximo ahorro posible de tiempos de cambio.

¿En qué momento intentamos identificar una limitación adicional en los recursos? Cuando hemos terminado la fase de subordinación y tenemos un pico de carga que no podemos nivelar hacia el pasado. En este punto examinaremos todos los lotes del pico e intentaremos conseguir todos los ahorros posibles en los tiempos de cambio. Dudo que esto vaya a suponer mucho, pero, probablemente, nos ayudará a reducir la cantidad de horas extras y de desvíos que harían falta para resolver el pico sin tener que considerarlo como recurso limitado.

Además, podemos hacer otra cosa. Si los cambios son importantes en nuestro entorno, entonces, vamos a identificar muy pronto un recurso limitado que requerirá que «peguemos» muchos de sus bloques. Si en la subordinación, en vez de considerar cada bloque original por separado, tratamos una serie de bloques pegados como si fuera un solo bloque, ahorraremos muchos cambios en los recursos que alimentan a la limitación. En este caso ya habremos pagado la mayoría de la penalización por inventario, ya que el programa de la limitación tuvo que desplazar los bloques. La penalización adicional que supone tratar los bloques pegados como si fueran un solo bloque es relativamente pequeña, y la posibilidad de ayudar es relativamente grande.

¿Podemos ahorrar tiempos de cambio para reducir su impacto en los gastos operativos? Esto sería ir demasiado lejos. La única forma efectiva de influir en los gastos operativos es reduciendo la necesidad de horas extras. ¿Dónde permitimos las horas extras? En el tambor (programa de la limitación), donde, de todas formas, ya estamos intentando ahorrar cambios; en los picos de carga del primer día, donde, de nuevo, ya hemos decidido ahorrar cambios; y en los picos de líneas rojas. En este último caso, de todas formas, no podemos desplazar el lote hacia el pasado. Así que, ¿de qué estamos hablando en realidad?

Bien, ya es suficiente. Vamos a construir un procedimiento que permita que la subordinación tenga en cuenta, adecuadamente, la capacidad de los recursos no limitados.

38

Los detalles del procedimiento de subordinación

Básicamente, ya están en nuestro poder todos los ingredientes necesarios para construir el procedimiento de subordinación; incluso sospecho que alguno está hasta demasiado pulido. Ahora lo único que tenemos que hacer es mezclarlos. Desgraciadamente, una vez solucionadas todas las posibles dificultades, la descripción del procedimiento resulta muy técnica. Todo el que no sea un fanático de los sistemas puede dejarnos ahora y volver después de este capítulo, siempre que no prefiera quedarse dormido.

La directriz más importante es ser constante, moviéndonos sólo hacia atrás en el tiempo. Esto supone que antes de que el sistema pueda asignar una operación a una fecha específica, debe asegurarse de que ya ha solucionado todas las operaciones que se deben asignar a fechas más tardías. No nos gustaría dejar cosas pendientes, ya que cualquier intento de retroceder para recoger algo «pendiente» sería una violación de nuestra decisión de ir constantemente hacia atrás en el tiempo.

Este párrafo tan inocente, que parece casi innecesario, de hecho nos va a dictar todo el procedimiento.

Ahora es obvio que, al retroceder en el tiempo, el sistema tiene que ser muy consciente de la fecha con la que está tratando actualmente. De ahora en adelante vamos a llamar a esa fecha la «fecha actual».

La advertencia casi histérica de que no dejemos nada pendiente nos indica, simplemente, que no debemos movernos de la fecha actual sin un buen motivo. Esto, inmediatamente, nos hace preguntarnos por las cosas que nos harán movernos por el eje de tiempo. Si consideramos la influencia tan insignificante que tiene el tiempo real de proceso de cada operación sobre el tiempo total de proceso del pedido, cada operación nos hará movernos en el tiempo. Esto parece ser un caso evidente de sofisticación excesiva: tener en cuenta algo que complica apreciablemente las cosas, pero que no influye en absoluto en la vida real.

Por tanto, tenemos que decidirnos por un intervalo de tiempo que represente la máxima sensibilidad del sistema. Todo lo que sea menor que este intervalo no debe aceptarse como razón para cambiar la fecha actual. Como las fechas de entrega de los pedidos se suelen dar sin especificar una hora determinada, parece razonable que, en lo que se refiere a movimientos hacia atrás en el tiempo, escojamos el día como unidad de sensibilidad máxima.

Esto nos deja con tres categorías distintas que nos obligarán a movernos en el tiempo. La primera es el tambor (programa de la limitación). Siempre que lleguemos a una actividad del tambor tendremos que tener en cuenta su fecha, que podría ser distinta de la fecha actual.

La segunda categoría la componen los *buffers*. Siempre que profundizamos desde un pedido, o desde una operación hecha por un recurso limitado, o desde una operación de línea roja hasta una operación normal, debemos proporcionar el *buffer* de tiempo correspondiente.

La tercera categoría son los picos de sobrecarga. La nivelación de un pico puede suponer que se asignen a una fecha anterior las operaciones correspondientes. Para tratar esta categoría tendremos que elegir una unidad mínima de tiempo, ya que la carga no se define en un punto, sino en su intervalo. Una vez más, necesitamos una unidad de tiempo que consideremos significativa. ¿Por qué no continuamos con la elección que habíamos hecho: un día?

A medida que bajemos por la estructura del producto, cada vez que nos encontremos con una de estas tres categorías tendremos que marcar dónde estamos, para poder volver a ese punto, pero no debemos continuar bajando. A medida que continua el proceso de subordinación, probablemente tendremos que ir guardando más recordatorios de este tipo, así que será conveniente que los dispongamos en una lista ordenada de recordatorios. El primero en la lista será el más cercano a la fecha actual y, el último, el más cercano al momento presente. Recordemos que nos estamos moviendo hacia atrás en el tiempo, todo está invertido.

Cuando empezamos nuestra lista de recordatorios no está vacía. Debe contener todo el tambor: las fechas de entrega de los pedidos y las fechas de terminación de los bloques del recurso limitado. También sabemos que, a lo largo del proceso de subordinación, debido a la segunda y a la tercera categorías, vamos a añadir muchos más «recordatorios» a nuestra lista.

Ahora vamos a empezar el proceso de subordinación. Vamos a coger el primer elemento de la lista para empezar a bajar desde él. Lo más probable es que el primer elemento sea un pedido, y debemos bajar desde él hacia las operaciones que lo alimentan (podemos encontrarnos con más de una, ya que el pedido puede ser para una diversidad de productos). Pero, primero, tenemos que sustraer el *buffer* de envío. Esto nos obligará a

retroceder. No podemos hacerlo inmediatamente porque puede que haya otros pedidos con la misma fecha actual. Por tanto, sólo vamos a identificar las operaciones que alimentan directamente al pedido y vamos a poner las notas correspondientes en nuestra lista de recordatorios.

Siempre que añadimos un nuevo miembro a la lista de recordatorios, la posición que toma se basa en la fecha que se le haya asignado. Recuerden que cualquier fecha anterior al presente debe igualarse a la fecha del presente, no tiene sentido dar instrucciones para el pasado. Así pues, en este caso, la fecha que se asigna es la fecha del pedido menos el *buffer* de envío, o la fecha presente, la que sea más grande de las dos. Ahora que ya hemos terminado con este pedido, podemos borrarlo de la lista de recordatorios y pasar al siguiente candidato.

Si continuamos siguienso esos pasos acabaremos cogiendo de la lista no un pedido, sino una operación. Entonces, tendremos que ocuparnos de la operación en sí. Tendremos que calcular la carga que representa, ajustando, también, de acuerdo con esta carga, la capacidad actual disponible del recurso que tiene que hacer esa operación.

Si la carga adicional es mayor que la capacidad disponible, tendremos que poner el exceso en el campo «carga restante» de ese recurso. Como hemos agotado la capacidad actual disponible, no se podrán programar para la fecha actual otras operaciones que necesiten este recurso. La única excepción se da cuando la fecha actual llega a la fecha del momento presente, en este caso todas las cargas se añaden al primer día, porque no hay posibilidad de dejar restos de cargas.

Nada de eso afecta a la fecha actual y, por tanto, podemos continuar bajando, registrando cada ensamblaje por el que pasemos, hasta que nos encontremos con una de las tres situaciones siguientes.

La primera es, probablemente, la más común: hemos llegado hasta el material. En este caso tendremos que volver al ensamblaje más cercano —recordemos que aún no ha ocurrido ningún movimiento en el tiempo— para bajar por las ramas adicionales que haya. Si no hay ningún ensamblaje marcado, podemos volver a nuestra lista de recordatorios.

La segunda situación se da cuando, al bajar, llegamos a una operación del tambor. No nos ocupamos de esta operación, porque ya ha sido considerada. Solamente tenemos que volver al punto de ensamblaje más alto (si lo hay) o a la lista de recordatorios.

La tercera situación ocurre cuando intentamos ajustar la disponibilidad del recurso correspondiente, encontrándonos con que su disponibilidad actual ha llegado a cero. En este caso tenemos que volver a la lista de recordatorios, ya que abordar esta operacion supone ir hacia atrás en el tiempo. El sitio adecuado de la lista lo determinará la cantidad de carga pendiente del recurso correspondiente.

Esta técnica nos obligará a saltar, como un saltamontes, de un punto a otro de las estructuras de producto, pero esto no supone ningún problema técnico, porque todos los datos están en la memoria. Todo el proceso se hace usando sólo la memoria *on-line*, porque no hay necesidad de archivar el programa resultante, ni para lanzar materiales ni para ninguna otra operación.

Recordemos que en esta fase no estamos seguros de si nos vamos a encontrar otra limitación, así que, ¿para qué vamos a perder el tiempo registrando en el disco los resultados? Una limitación adicional necesitará, por definición, un cambio en la subordinación, lo que hace muy probable que el programa actual no sea el definitivo. Cuando no se ha encontrado ningún conflicto al final de la subordinación, repetiremos toda la última vuelta de subordinación y escribiremos el programa en el disco. Es mucho más económico «desperdiciar» un ciclo de subordinación que «desperdiciar» el tiempo de escribir en el disco.

Continuaremos usando este proceso básico, aplicando las directrices sobre casos especiales que se han desarrollado en los dos últimos capítulos. Los detalles técnicos son enormemente aburridos y se pueden encontrar en el manual correspondiente, así que, no tenemos por qué sufrirlos; seguro que no nos van a enseñar nada nuevo.

El sistema continuará trabajando hasta que agotemos la lista de recordatorios. Con esto concluye esta vuelta de la subordinación. Ahora tenemos que descubrir cómo deberíamos comprobar la existencia de los conflictos que puedan haber surgido.

39

Identificación de la siguiente limitación: cerrando el bucle

Lo más probable es que al final de la fase de subordinación nos hayan quedado algunos recursos que tienen sobrecargas el primer día. En realidad, esta descripción no es lo suficientemente exacta. Cuando todavía hay un recurso limitado sin identificar, la nivelación de los picos hacia atrás en el tiempo no sólo crea sobrecargas en el primer día, crea montañas de sobrecarga, y no sólo en un recurso. No todos los recursos que muestran sobrecargas espectaculares son limitaciones. Al tener en cuenta las limitaciones de capacidad de un recurso reduciremos la carga de todos los demás. ¿Cómo debemos escoger la siguiente limitación?

Cuando calculamos la falta de capacidad, la cantidad de horas que nos faltan no nos muestran totalmente la gravedad de las limitaciones que sufre la empresa. Consideremos, por ejemplo, un recurso al que le faltan 100 horas (el número de horas de carga acumuladas en el primer día menos la disponibilidad del recurso) y otro, al que sólo le faltan 50 horas. No deberíamos apresurarnos a concluir que debemos elegir el primero de ellos. Puede que el *buffer* más pequeño al que alimenta el primer recurso sea de 200 horas, mientras que el segundo alimenta a un *buffer* de sólo 20 horas. En este caso es obvio que el segundo recurso resulta ser una limitación mucho mayor que el primero.

Esto supone que lo primero que debe intentar el sistema es minimizar las sobrecargas ahorrando tiempos de cambio, asignando horas extras y nivelando hacia adelante, hasta el límite de medio *buffer*. Pero, a partir de ahí, deberíamos considerar la sobrecarga restante de acuerdo con la limitación que imponga en la operación global, lo que quiere decir, de acuerdo con el daño que resulta para la empresa, no por el número de horas que todavía faltan. Como mejor se ilustra el daño potencial es mostrando las sobrecargas en unidades de penetración de los agujeros resultantes, expresadas en cuanto a la longitud de *buffer*. Por ejemplo, una sobrecarga de tres

significaría que la sobrecarga va a provocar que los agujeros entren en el origen del *buffer* más allá del límite de peligro (recordemos que ya hemos usado la mitad de la protección) hasta una distancia igual a tres veces la longitud del *buffer*.

Como el usuario todavía puede hacer cosas para resolver una sobrecarga antes de decidir que tiene una limitación adicional, debe mostrarse la lista de todos los recursos que tienen un pico el primer día. El sistema deberá ordenar la lista en función de la profundidad de penetración, mostrando, también, las horas de sobrecarga.

Este último dato sigue siendo importante porque ayuda al usuario a decidir si hay necesidad de considerar otra limitación. Recordemos que no tenemos prisa en hacer esto, ya que cada limitación que se añade necesita protección adicional y, por tanto, un incremento del inventario. Por otra parte, si, de todas formas, vamos a considerar que un recurso está limitado, no tiene sentido aliviar la sobrecarga del primer día de ese recurso, además de la de los demás recursos no limitados.

Puede que este último punto necesite un poco más de aclaración. Supongamos que un recurso necesita un 50 por 100 de capacidad de protección. Esta capacidad no protege al recurso en sí, sino a la viabilidad de explotar las limitaciones del sistema. Por tanto, en el momento que se reconoce que el recurso es una limitación, toda esta capacidad de protección queda liberada para explotación. Pagamos aumentando la protección en otras partes del sistema, pero aquí conseguimos más capacidad. En consecuencia, todos los intentos de aumentar la capacidad del recurso a base de horas extras y desvíos de lotes se considerarán nulos en el momento en que el recurso sea declarado como limitación. La conclusión es que, si resulta obvio que no podemos eliminar el pico del primer día, no tiene sentido intentar minimizarlo.

La mayor arma que tiene el usuario en esta fase no son las horas extras, sino el desvío de lotes hacia otros recursos. Por tanto, tiene que haber una lista de posibles lotes que puedan desviarse en cada recurso para que el usuario la pueda consultar cuando lo desee.

Supongamos que, a pesar de todos los esfuerzos, se identifica otro recurso limitado, ¿cómo debemos tratarlo? Ahora el método se entiende mucho mejor. Nos deberíamos preguntar, concentrándonos en este recurso, qué debería hacer si no hubiera limitaciones internas. En otras palabras, construimos el cuadro de las «ruinas» para este recurso. El tambor lo cogemos tal y cómo está, y también las longitudes de los *buffers* de envío, pero del *buffer* del recurso sólo cogemos la mitad (ésta era la regla) cuando una operación de un recurso limitado alimenta a otra. Calculamos qué (el bloque) y cuándo (la fecha de terminación del bloque) tiene que hacer esta nueva limitación, suponiendo que disponemos de capacidad infinita.

Ahora empezamos a tener en cuenta la disponibilidad de este recurso, y comprobamos si hay conflictos. Debemos tener un poco más de cuidado en esta fase, comprobando si hay conflictos con las instrucciones del tambor actual. Anteriormente no tuvimos que hacer esto porque no había posibilidad de que surgieran conflictos, ya que el tambor sólo estaba compuesto por las fechas de entrega de los pedidos. Pero, aquí, la situación es distinta, porque el tambor actual incluye instrucciones para otro recurso limitado.

Para representar las interrelaciones que hay entre los bloques del primer recurso limitado (bloques viejos) y los del segundo recurso limitado (bloques nuevos), vamos a usar, otra vez, el concepto de barras, pero, en este caso, tendremos que definir «barras de tiempo».

Si un bloque nuevo alimenta, directa o indirectamente, a un bloque viejo, el bloque nuevo deberá estar teminado, al menos, medio *buffer* antes de la fecha del bloque viejo. Así pues, vamos a acoplar al bloque nuevo una «barra de tiempo» imaginaria. Esta barra de tiempo, en particular, es igual a la longitud de medio *buffer* de recurso, y es sensible a la fecha del bloque viejo. Para nuestra «barra de tiempo», esta fecha es como una pared de acero. Para describir con exactitud la libertad de movimiento en el tiempo que se permite al bloque nuevo, tendremos que acoplar la barra de tiempo a la derecha de éste. Vamos a llamarlos bloques A (para simbolizar que son bloques con la barra apuntando hacia *adelante* en el tiempo).

Si a un bloque nuevo lo alimenta uno viejo, le acoplamos la barra de tiempo a la izquierda. Lo demás no varía; la longitud sigue siendo medio *buffer* de recurso, y la fecha que hace de pared de acero es la fecha del bloque viejo. A éstos les llamamos bloques R (la barra apunta hacia atrás, *retrocede* en el tiempo).

Como ya dijimos, al considerar la disponibilidad de la nueva limitación, pueden revelarse conflictos con la fecha establecida de la primera limitación. Las barras de tiempo ayudarán a aclarar estos posibles conflictos.

El primero de ellos es infrecuente. Puede ocurrir cuando el «viejo» recurso limitado tiene bloques con barras. Para refrescar la memoria, se acoplan barras a los bloques cuando un bloque alimenta, pasando por otras operaciones intermedias, a otro bloque hecho por el mismo recurso limitado. Intenten imaginar que una de las operaciones intermedias la hace el segundo recurso limitado. Si no hay holgura entre los bloques «viejos», no habrá sitio para el *buffer* adicional que se requiere ahora para el bloque «nuevo». En otras palabras, siempre que tengamos un bloque nuevo que tenga barras de tiempo hacia atrás y hacia adelante (un bloque AR) podemos encontramos con un conflicto con el tambor viejo.

Aunque hubiera algo de holgura podríamos seguir encontrándonos con problemas. Es bastante obvio que tenemos muy poca libertad para colocar

los bloques AR, y puede que haya más de uno compitiendo en el mismo intervalo de tiempo. Por tanto, teniendo en cuenta el hecho de que sólo disponemos de un número finito de unidades del nuevo recurso limitado, vamos a usar una máquina niveladora refinada.

Cuando se dispone a trabajar con las «ruinas» de la nueva limitación, nuestra niveladora refinada levanta todos los bloques y trabaja, inicialmente, sólo con los bloques AR. A medida que nivela los bloques AR, puede encontrarse con la necesidad de mover un bloque violando una barra de tiempo. Aquí tenemos un conflicto que sólo se puede resolver de dos maneras. O el bloque nuevo se desvía del nuevo recurso limitado, o se modifica el tambor inicial. Esta elección no es del sistema, sino del usuario.

Una vez que ha terminado con los bloques AR, la máquina tiene que nivelar los bloques A, tratando a los AR como bloques inamovibles. Aquí nos podemos encontrar con que la niveladora empuja bloques A más allá del presente, en una clara violación de la realidad. Una vez más, la única forma de resolver los conflictos es desviando lotes o modificando el tambor previo. Si tenemos que modificar el tambor original, lo hacemos en esta fase. Todos los bloques viejos que participan en las violaciones se convierten ahora en bloques R, adquiriendo barras de tiempo que asegurarán el mínimo empuje hacia el futuro. El proceso ha vuelto a empezar. Nos olvidamos de la existencia del segundo recurso limitado; ya ha dejado su marca con el acoplamiento de las barras de tiempo a los bloques de la limitación «vieja». Repetimos la explotación, la subordinación y la elección del nuevo recurso limitado.

¿No deberíamos temer que vamos a tener que repetir este ciclo eternamente? En absoluto. El nuevo recurso limitado siempre sitúa limitaciones sobre el antiguo en una dirección. Globalmente, este proceso fuerza a un movimiento de la carga hacia el futuro y, por tanto, alivia la posibilidad de conflictos dentro del horizonte especificado. La naturaleza de este proceso iterativo es tal que debe converger y, además, rápidamente.

Si no se ha dado ninguna violación de los recursos limitados anteriores, todavía tenemos que ocuparnos de las violaciones de capacidad disponible de los recursos limitados actuales, y de los conflictos con la demanda del mercado. Por tanto, el sistema debería repetir el proceso con los bloques R y con los bloques libres. Aquí no hay necesidad de comprobar; no hay posibilidad de que se dé un conflicto con la anterior limitación. A los bloques rojos se les da el mismo tratamiento. Después, la subordinación se lleva a cabo cuando el punto de arranque de la lisa de recordatorios contiene los bloques de todas las limitaciones.

El sistema repite el proceso hasta que se resuelven todas las sobrecargas del primer día, se identifican todas las limitaciones, se explotan, se hace la

subordinación, y no quedan conflictos conocidos para que los resuelva el personal de fábrica. Parece que, por fin, lo hemos conseguido.

Pero un momento, ¿cuántas limitaciones vamos a encontrar así? Si estamos en lo cierto, hemos demostrado que si una cadena tiene más de un eslabón débil, la realidad la romperá rápidamente. O, de forma menos metafórica, no deberíamos permitir recursos limitados interactivos. El simple hecho de considerar casos en los que un recurso limitado tiene una «barra» apuntando a otro recurso limitado nos indica, claramente, que estamos permitiendo limitaciones interactivas. ¿Debemos permitirlo?

Casi todos los intentos de usar el sistema de información para analizar empresas reales han revelado que este tipo de situación sí se da en la realidad. Sí, en todos los casos el cumplimiento de programa era bastante horrible y parecía que las empresas se gestionaban a base de acelerar pedidos, pero, de hecho, estas empresas sobreviven. Por otra parte, si revisamos la lógica que nos ha llevado a intentar evitar las limitaciones interactivas, no parece que haya ningún fallo. La única conclusión posible es que se nos ha escapado algo. Puede que, en la realidad, los casos que muestran a un recurso limitado alimentando a otro sean conceptualmente distintos del caso que hemos analizado lógicamente.

¿Cómo puede ser? ¿Podemos representar un caso en el que un recurso limitado está alimentando a otro, y, a pesar de todo, en realidad sólo tenemos un recurso limitado? A primera vista parece que la pregunta es absurda, pero, un momento, puede que merezca la pena. Puede que haya una pista en la forma secuencial en que se nos revelan las limitaciones. La primera limitación que encontramos fue la demanda del mercado. Puede que debamos rehacer la pregunta mirando la situación desde el aspecto del mercado. Ahora resultaría algo así: ¿podemos representar un caso en el que un recurso limitado está alimentando a otro, y, a pesar de todo, cada limitación de mercado sólo se alimenta de un recurso limitado?

No parece que ayude mucho, pero, pensándolo bien... Supongamos que la demanda del mercado para un producto supone una carga de un 100 por 100 para un recurso, mientras que para otro, que alimenta al primero, sólo supone un 70 por 100. Ahora supongamos que la demanda del mercado para otro producto supone una carga cero para el primero y un 30 por 100 para el segundo. Si cada vez que el segundo recurso se retrasa, da prioridad absoluta al primer producto, al producto que creaba la limitación en el otro recurso, ¿nos encontramos, de hecho, en una situación de limitaciones interactivas? Si miramos nuestra empresa a través del mercado que pide el primer producto, sólo vemos la involucración de una limitación. El otro recurso, aunque tenga una carga del 100 por 100, tiene, en realidad, mucha capacidad de protección para nuestro producto. De hecho, si seguimos la regla de prioridad que hemos dicho antes, el 30 por

100 de su tiempo disponible es capacidad de protección para esa demanda del mercado. Y, ciertamente, la demanda del mercado para el segundo producto sólo ve un recurso limitado.

Sí, podemos tener un recurso limitado alimentando a otro recurso limitado, y, aún así, estar en un sistema que no tiene recursos limitados interactivos. Nuestro análisis original ignoró el caso en el que hay una salida adicional al mercado entre dos recursos limitados. Este tipo de situación no sólo existe, sino que, además, es muy recomendable, porque hace posible una utilización mucho mejor de las inversiones que se han hecho en los recursos. Por supuesto que, si no se obedece rigurosamente la regla de prioridad que hemos dicho antes, puede ser una situación peligrosa, especialmente para los clientes de esas empresas.

Como estamos suponiendo la existencia de gestión del *buffer*, tenemos el mecanismo para avisar a los centros de trabajo cuando haya retrasos significativos: agujeros en las zonas de seguimiento y de aceleración de los orígenes. Lo que tendremos que demandar de la fase de programación es que aparezca un número cerca de cada bloque que tenga una barra apuntando a una limitación previa. Este número, simplemente, indicará la importancia relativa de la limitación alimentada. En otras palabras, en qué interación se encontró a este recurso limitado. ¿Hemos terminado ya con la fase de programación? Esta pregunta, por definición, es ridícula. Nunca se acaba. El número de opciones, como combinaciones de recursos, bucles de retrabajo, etc., es probablemente infinito. Hemos terminado con el caso genérico que puede ser aplicable, tal y como está, a la gran mayoría de las empresas, pero, siempre hay oportunidades más que suficientes para hacerlo todavía mejor.

¿Deberíamos continuar ahora desarrollando los detalles de la fase de control? Personalmente creo que no. Mientras la fase de programación no esté implantada, y la gestión de *buffer* se haga manualmente, lo más probable es que cualquier intento de automatizar las medidas del rendimiento local lleve a la empresa al caos. Es mejor esperar y disfrutar primero de los grandes beneficios que podemos conseguir implantando la primera fase de nuestro sistema de información, la programación. Ahora, procede hacer un resumen conciso, no sólo de los beneficios cuantitativos, sino, también, de los cualitativos. Pero, mejor lo dejamos para el próximo capítulo, que nos servirá de sumario de nuestra discusión.

40

Sumario parcial de los beneficios

¿Cuáles son los beneficios de tener funcionando la fase de programación, aparte del beneficio obvio de ser capaz, por fin, de generar un programa fiable e inmunizado? Vamos a verlo sistemáticamente desde el punto de vista de cada función directiva.

El primer beneficio significativo de tener todos los datos en la memoria, y, por ello, de ser capaces de realizar los cálculos a velocidades impresionantes, es el hecho de que el cálculo del requerimiento neto —calcular cuántas unidades tienen que producirse en cada punto— ya no es una cuestión de horas, es cuestión de segundos. Esto acabará con la práctica actual de generar, muy infrecuentemente, el requerimiento neto, y lo pondrá al alcance de cualquier directivo cuando lo necesite. La generación neta, como llamamos al intento de calcular sólo los cambios, se convertirá en algo del pasado. Este método tan engorroso, que se había desarrollado para reducir el muchísimo tiempo de ordenador que era necesario, ha llevado, en muchos entornos, a una lucha continua contra las faltas de material. No es de extrañar; si el cálculo de requerimientos se basa en seguir los cambios a través de los informes de transacciones, las discrepancias son casi inevitables. La rapidez no sólo supone que podamos tener antes las cosas, muchas veces nos hace capaces de librarnos de procedimientos ineficientes y embarazosos.

Pero, un momento. Si seguimos así, vamos a convertir este breve resumen en un examen de las prácticas actuales. Esto, sin duda, nos llevaría tanto como lo que ya llevamos de nuestra discusión. Sí, desde luego que tenemos derecho a estar orgullosos de vencer tantos obstáculos, pero, mejor nos limitamos a hacer un resumen breve y conciso de los beneficios.

Los beneficios para la gestión de materiales son muy claros. Este sistema es nada menos que la herramienta que hemos esperado durante tanto tiempo, la herramienta que MRP prometía ser, pero no fue. Pero parece

que los beneficios para los directores de producción están casi al mismo nivel. Todo el que haya pasado algún tiempo en una planta es perfectamente consciente de la lucha constante para determinar el tamaño de las tiradas de cualquier máquina que esté bastante cargada y tenga tiempos de cambio largos. Por fin, tenemos un instrumento que puede ayudar a encontrar una respuesta global dinámica para todas las tiradas. Esta nueva capacidad, combinada con lo que casi es una bola de cristal para predecir la necesidad de horas extras, parece demasiado buena para ser cierta. Por no mencionar el hecho de que al poner en manos de los directores de materiales un instrumento mucho más fiable, seguro que la vida del personal de producción va a resultar más fácil.

Pero no son éstos los únicos que van a disfrutar de los beneficios. Si no me equivoco, es la primera vez que ventas va a tener unos preavisos fiables. El sistema les va a dar la capacidad de comunicarse con los directores de producción, no a base de echarse las culpas, sino a base de una terminología común: de hechos y consideraciones de la vida real.

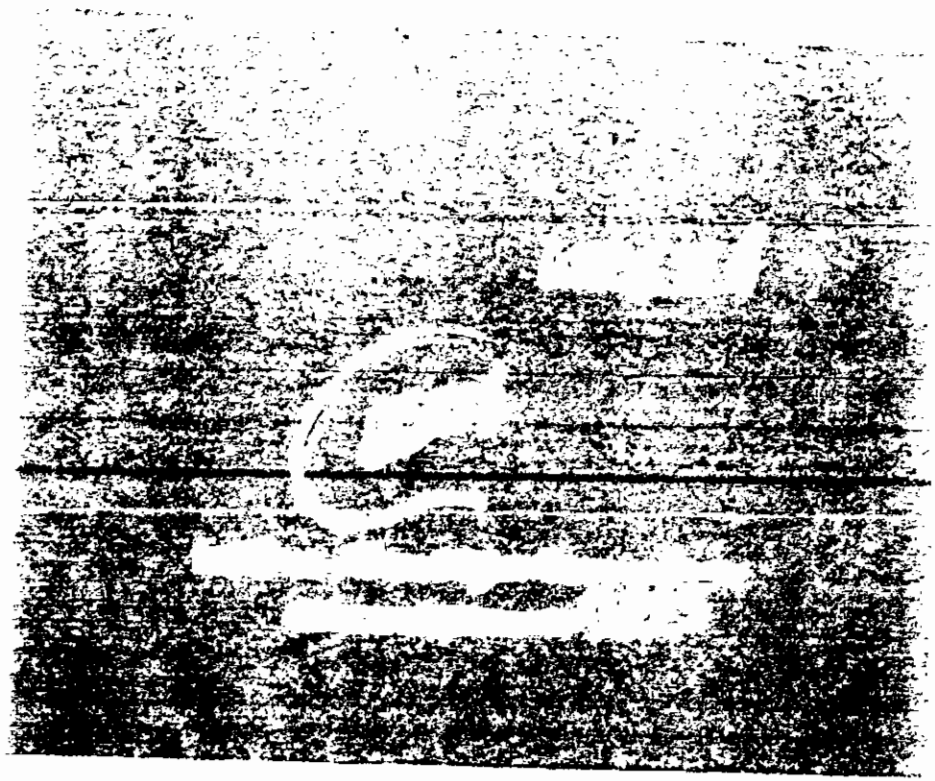
Los que probablemente van a disfrutar más de los beneficios son los ingenieros de proceso y los directores de calidad. Aunque la técnica de *buffers* dinámicos reduce drásticamente el tiempo total de fabricación y el inventario, en realidad, su principal beneficio reside en otra área. Por primera vez, el seguimiento de los agujeros en los orígenes de *buffer* apuntará hacia los procesos que se deben mejorar, en vez de hacia centros de trabajo que no tienen suficiente capacidad de protección. Se puede proporcionar a los círculos de calidad la información vital de qué problemas son los que deben concentrarse en resolver, con la seguridad de que cada vez que resuelvan uno toda la empresa va a cosechar las ganancias. Esto evitará que los círculos de calidad se deterioren convirtiéndose en reuniones sociales sin sentido.

Pero, probablemente, para quien es más importante el sistema, aunque sólo tenga la primera fase de programación, es para la alta dirección. Esto se debe, por una parte, a que hemos insistido en construir el sistema de forma que no considere ninguna limitación política, y, por otra parte, a que nos hemos asegurado de que se identificarán todas las limitaciones físicas y de que se resolverán todos los conflictos que haya entre ellas. El programa resultante es inmune y duradero y, por tanto, cada vez que digamos que «no se puede seguir el programa», sabremos que la única razón para decirlo es que existe una limitación política. Lo que ha convertido a nuestro sistema en un instrumento eficaz para identificar limitaciones políticas internas es precisamente la terquedad en no reconocer esas limitaciones políticas.

Es bastante asombroso darse cuenta de que todos estos beneficios palidecen cuando se los compara con los beneficios que se pueden esperar de

la fase de control, que, a su vez, son casi triviales si se los compara con el propósito principal de todo el sistema: la fase de simulación. Pero esto, ciertamente, es tema de otra discusión.

Puede que la mejor forma de acabar esta discusión sea recordándonos que hemos desarrollado este sistema basándonos en el reconocimiento del mundo del valor. Hoy en día la cultura de nuestras empresas todavía está demasiado inmersa en el mundo del coste. No nos engañemos pensando en que es posible cambiar la cultura de nuestras empresas usando un ordenador.



A. Goldratt Institute Ibérica

En este libro se contienen algunos de los conceptos básicos de un enfoque de gestión llamado Dirección de las Limitaciones (Theory Of Constraints - T. O. C.). Si desea profundizar en T. O. C. y cómo aplicarlo en su empresa,

Cumplimente los datos siguientes:

Nombre _____

Empresa _____

Puesto _____

Dirección _____

Ciudad _____ D.P _____

Teléfono ( ) _____ Fax ( ) _____

Actividad de la empresa _____

Nº de empleados _____

Estoy interesado en obtener información sobre:

_____ Otras publicaciones

_____ Actividades del Instituto Goldratt

Envíe el impreso superior, por correo o fax, a:

A. Goldratt Institute Ibérica

Puerto de los leones, 1 - Of 113
28220 Majadahonda - Madrid - ESPAÑA
Tel. (91) 639 49 55
Fax (91) 639 41 23