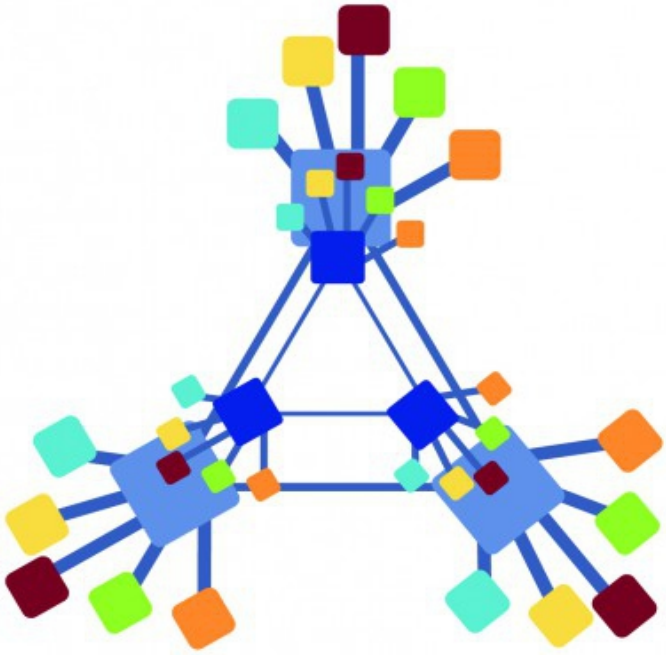
La computación competitiva

Andrés Gómez Tato



LA COMPUTACIÓN COMPETITIVA

Andrés Gómez Tato

Ames, septiembre de 2013

La computación competitiva

Primera Edición: mayo de 2014

LaComputacionCompetitiva@gmail.com

ISBN: 978-84-616-9976-6

Depósito legal: C-728-2014

ÍNDICE

PRÓLOGO

Las tecnologías de la información y las comunicaciones (en forma breve, TIC) se han convertido en omnipresentes en nuestras vidas. Ordenadores, Internet, teléfonos fijos y móviles son casi imprescindibles para nuestros trabajos, nuestra vida cotidiana. De hecho, estamos inmersos en una nueva cresta de la ola innovadora en el macrosector de las TIC, con la irrupción casi simultánea de tres nuevas facetas: la nube, el tratamiento de enormes volúmenes de datos (o *Big Data*) y las *apps* para móviles y *tablets*. Por su menor visibilidad, es menos conocido que también la computación de altas prestaciones es un área en auge que ha salido de los nichos tradicionales de la investigación, las grandes organizaciones y las entidades gubernamentales para esparcirse en más sectores y en más áreas de conocimiento.

Este pequeño ensayo tiene el modesto propósito de inspirar a los emprendedores para que aprovechen el mundo de oportunidades presentes y futuras que tiene la computación de altas prestaciones, también conocida como HPC. A pesar de que la computación ha estado en el germen de los ordenadores, se encuentra ahora en una encrucijada, necesitando soluciones ingeniosas al bloqueo en el crecimiento de la frecuencia de reloj en las CPUs para seguir mejorando sus capacidades y para adaptarse

a nuevas formas de uso. Existen multitud de problemas y necesidades que se pueden resolver con la computación de altas prestaciones, lo que crea también multitud de oportunidades. Es una tecnología que será necesario dominar para poder competir eficazmente sin perder el bienestar que hemos alcanzado en los últimos decenios. A fin de cuentas, la HPC no hace más que explotar el conocimiento adquirido previamente y el conocimiento es la fuente de la riqueza de los países cuando están escasos de recursos naturales. Y Europa es una fuente inagotable del mismo. Es nuestra tarea conseguir explotarlo para mejorar la calidad de vida de nuestra sociedad.

Es posible que mis colegas del área de la computación no estén totalmente de acuerdo con el concepto empleado para definir HPC. Algunos lo podrán considerar demasiado cerrado. Sin embargo, creo que es el más correcto, ya que los otros usos de los computadores que se han asociado a HPC no tienen la característica fundamental de necesitar que los resultados obtenidos con los computadores se ajusten a la realidad. De los nuevos usos más habituales, quizá el tratamiento de grandes volúmenes de datos comerciales o de las ciudades inteligentes sea el más cercano y merezca su inclusión en la definición utilizada para perfilar el contenido del libro. En cualquier caso, con o sin estos nuevos usos, la HPC es una tecnología que es necesario dominar y aprovechar.

Para no seguir favoreciendo a las grandes compañías de software y hardware que ya tienen suficiente publicidad gratuita en los medios de comunicación (pensemos todos en unas gafas mágicas que tampoco son la panacea y que están estos días hasta en la sopa), en caso de dar ejemplos, he preferido utilizar pequeñas compañías frente a las grandes, a ser posible europeas, y proyectos públicos innovadores en vez de iniciativas empresariales, al estar al alcance de todos.

El libro se divide en seis capítulos. El primero busca contextualizar la computación de altas prestaciones y su contribución a la competitividad y a la historia de la informática. El segundo describe los diferentes componentes técnicos que están involucrados, desde el hardware hasta las herramientas de software. Este capítulo quizá le resulte complicado al lector ajeno a las tecnologías de los computadores, y en una primera lectura podría saltarse, aunque recomiendo que no se haga, ya que ayuda a comprender el hilo argumental del libro. Le sigue otro en donde se muestra como se utiliza esta tecnología en diferentes sectores, en que se aplica eficazmente y en donde se ha convertido en imprescindible. La innovación está presente en los dos siguientes: el primero analizando la relación existente entre la

nube (el Cloud) y la HPC; el segundo dando un breve bosquejo del panorama de la investigación, el desarrollo y la innovación en el área de la computación y sus aplicaciones. Bosquejo, ya que abordar detalladamente las necesidades de I+D+I necesitaría todo un libro solo para describirlas. Finalmente, el último capítulo cierra este ensayo con unas pequeñas conclusiones sobre las oportunidades que no se han de dejar pasar. Es un capítulo de reflexión, remarcando aquellos puntos ya descritos anteriormente que pueden aprovecharse para crear nuevas actividades, nuevos negocios prósperos desde cualquier lugar, incluso en la periferia de Europa en donde tengo el placer de vivir. Al final del libro también hay una pequeña bibliografía seleccionada para cada capítulo, por si el lector quiere profundizar sobre los temas tratados.

Si al final de la lectura de este libro el lector ha aprendido algo nuevo, me doy por satisfecho. Si además, como consecuencia de su lectura decide emprender una nueva actividad investigadora o innovadora relacionada con la computación de altas prestaciones, no deje de comunicármelo, ya que me hará extremadamente feliz y compensará el esfuerzo invertido.

No quiero terminar este breve prólogo sin agradecer a aquellos que me han ayudado a escribir este libro. Primero a mis compañeros del CESGA, especialmente a Fernando, Nacho y Javier, que han aportado ideas y comentarios útiles. A José, mi vecino y amigo, por sus comentarios durante las cervezas sabatinas y abrirme los ojos sobre la necesidad de destacar el papel de la HPC en el mundo, aunque probablemente él no sea consciente de lo que disparó en mí esa idea. Finalmente, a mi familia, que han soportado un verano de arduo trabajo vespertino y, especialmente a mi mujer, que ha sido mi correctora, mi diseñadora gráfica y mi crítica más fiel. Va por ellos.

Ames, verano de 2013.

«The purpose of computing is insight, not numbers» [1]

Richard Hamming

CAPÍTULO 1. COMPETIR COMPUTANDO

¿Por qué el 97% de las empresas industriales que utilizan la computación de altas prestaciones la consideran indispensable para competir? Pocas veces se produce una coincidencia tan abrumadora sobre una tecnología. Probablemente hoy obtendríamos la misma respuesta si se preguntara sobre la mensajería electrónica o la utilización de Internet en general. Estas tienen un nivel de implantación muy alto tanto en las empresas como en los ciudadanos y han significado un mundo de oportunidades de negocio, todavía sin conocer el final. Sin embargo, la computación de altas prestaciones no está tan ampliamente instalada entre aquellas compañías que podrían aprovecharse de sus capacidades. En muchos casos incluso desconocen que existe o como la pueden utilizar para competir en un mercado más globalizado.

La respuesta a la pregunta probablemente tenga mucho que ver con los ahorros de costes y la mejora de la calidad que ha supuesto la irrupción de esta tecnología en diversos sectores. Así, en el sector del automóvil ha permitido reducir los tiempos medios de desarrollo de un nuevo vehículo de 5 a solo 2 años mientras se ha mejorado el confort de los pasajeros, el respeto al medio ambiente o la respuesta en caso de colisión. Otros sectores de la manufactura también se benefician enormemente, como puede ser el aeronáutico o el eólico. Pero la computación de altas prestaciones se ha

1. El propósito de la computación es comprender, no obtener números. Traducción del autor.

extendido como una mancha de aceite a otras áreas como la sanitaria, ayudando en el descubrimiento de nuevos fármacos o en el diagnóstico de enfermedades; en la banca, mejorando los análisis de riesgos y en la detección de fraudes; o en el energético, sirviendo como herramienta para la detección de nuevos yacimientos petrolíferos. Sus aplicaciones son innumerables, sujetas solo a las limitaciones del propio conocimiento sobre los procesos en donde se quiere utilizar.

El concepto de computación de altas prestaciones

Precisamente es el conocimiento, o mejor dicho, el tipo de conocimiento necesario para la creación y utilización de aplicaciones de computación de altas prestaciones (a lo largo del libro se utilizará también las siglas en inglés HPC para referirse a ella) lo que puede ayudar a distinguir entre esta y la informática más convencional de gestión y ocio a la que estamos más acostumbrados. Así, la compañía Intersect360 proporciona la siguiente definición cuando realiza sus encuestas sobre su utilización entre las empresas:

«*La computación de altas prestaciones (HPC) es el uso de servidores, clústeres y supercomputadores —más el software, herramientas, componentes, almacenamiento y servicios— para tareas científicas, de ingeniería o analíticas que son particularmente intensivas en computación, uso de memoria o manejo de datos.*»[2]

La definición incluye cuatro elementos que en conjunto la hacen única: hardware, software, forma de uso y tipo de utilización. Pero lo más destacable es este último: para tareas científicas, de ingeniería o analíticas. Aunque la innovación constante hace que se creen nuevos modelos de utilización de este tipo de computación, su núcleo está asociado a estos tres tipos de tareas, que requieren un conocimiento importante, sino exhaustivo, del software y de los métodos que está utilizando. Las otras características reflejadas en la definición, aunque importantes, no son suficientes para distinguirla de otras. El uso intensivo de la computación también se produce en los juegos de computador o de las consolas. Las bases de datos utilizan intensivamente la memoria o manejan cantidades importantes de datos. Incluso los servidores, ordenadores o computadores, como los queramos nombrar, son ahora casi iguales, sino idénticos, a los existentes en los entornos de gestión. De hecho, actualmente los supercomputadores se fabrican fundamentalmente con componentes casi convencionales (*off-the-shell*) unidos por redes de comunicaciones de muy altas prestaciones.

2. Traducción del autor de: «High Performance Computing (HPC) is the use of servers, clusters, and supercomputers —plus associated software, tools, components, storage, and services— for scientific, engineering, or analytical tasks that are particularly intensive in computation, memory usage, or data management.»

Como se comentó anteriormente, la definición se queda incluso corta en la forma de uso, debido a su introducción en otros muchos campos de la actividad diaria. Aunque su implantación está muy arraigada en las áreas de las ciencias físicas, matemáticas, químicas o ingenierías, se ha extendido rápidamente su utilización a otros ámbitos muy pujantes como la biología, la medicina o las finanzas en donde se ha convertido en una potente herramienta de trabajo, en muchos casos imprescindible.

Para comprender mejor lo que es la computación de altas prestaciones consideremos un caso modelo sencillo de uso relacionado con la ingeniería o la ciencia básica. Supongamos que queremos conocer mejor un proceso que afecta al procedimiento de fabricación de un producto para mejorarlo y, por lo tanto, hacerlo más efectivo o eficaz. Lo primero que se necesita es tener un modelo conceptual de lo que ocurre en el proceso, es decir, a partir de nuestro conocimiento del mismo (y en muchos casos de la física, la química o la biología que está involucrada), imaginamos como se desarrolla el mismo. Partiendo de esta hipótesis, se construye un modelo matemático del proceso que refleje esa realidad (Ilustración 1). En algunos casos, este modelo se podrá resolver de forma analítica, es decir, solucionando las ecuaciones matemáticas hasta tener un resultado dado por una fórmula o una función sencilla que depende de una serie de parámetros o magnitudes (como puede ser la presión o la temperatura) que deseablemente son los que se pueden controlar en el proceso. Sin embargo, existen otros muchos casos en donde no es posible obtener este tipo de soluciones, por lo que es necesario recurrir a buscar soluciones numéricas aproximadas.

Para ello se desarrolla un programa informático que resuelve numéricamente el modelo matemático utilizando la capacidad computacional de los ordenadores. Pueden existir multitud de métodos de resolución de las ecuaciones que modelan el proceso, cada uno de ellos con sus limitaciones y aproximaciones. Pero todos ellos tienen en común por la necesidad de hacer una cantidad ingente de operaciones matemáticas. De ahí el uso intensivo de computación, generalmente acompañado también de una utilización también intensiva de la memoria.

Cuando el programa de ordenador esté disponible, se podrá estudiar el comportamiento del proceso variando uno o varios de los parámetros que lo controlan. Por ejemplo, la temperatura. Pero antes de que realmente sea posible utilizarlo de forma rutinaria es necesario, como en cualquier programa informático, hacer la verificación y la validación del mismo.

Aunque los procesos de verificación y validación se han de hacer en todos los programas de ordenador, existe para este caso una sutil diferencia. La verificación es el proceso para comprobar que el resultado numérico proporcionado coincide con suficiente exactitud con la solución del modelo matemático propuesto. Por ejemplo, uno de los test que se realiza frecuentemente es la comparación de la solución numérica con una solución que se pueda obtener de forma analítica en algún caso sencillo. Básicamente, estas pruebas lo que intentan es responder a la pregunta de si estamos resolviendo las ecuaciones que modelan el proceso de forma correcta.

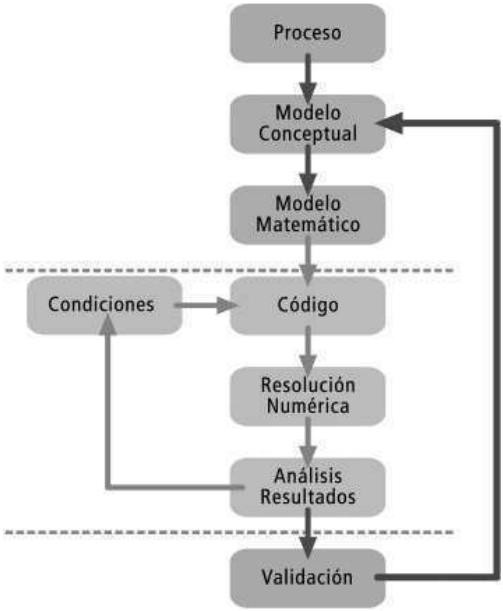


Ilustración 1: Modelo simplificado del uso de la computación de altas prestaciones.

La validación es más amplia, ya que abarca tanto el programa de ordenador como el modelo matemático utilizado. Es decir, necesita comprobar si hemos resuelto las ecuaciones correctas, si el modelo conceptual o matemático es el adecuado. Para ello, lo más acertado es comparar los resultados numéricos con los datos experimentales que se puedan obtener del proceso. Si la solución numérica no coincide con lo esperado,

es necesario reformular de nuevo el modelo conceptual y, por tanto, las matemáticas involucradas.

La Asociación Americana de Ingenieros Mecánicos (ASME) define el proceso de validación como «*la determinación del grado de que un modelo es una representación exacta del mundo real desde la perspectiva de los usos previstos del modelo*». Por tanto, no es necesario que el modelo y el programa de computador que lo resuelve de forma aproximada, se puedan utilizar para resolver todas las posibles variantes del proceso, sino solo aquellas en donde existe interés en conocer la solución. Es decir, el programa solamente proporciona soluciones correctas dentro de un dominio de aplicación. De ahí que la documentación que tiene que acompañar a estos programas de computación ha de incluir información exhaustiva sobre el modelo conceptual o matemático subyacente, las aproximaciones que se han aplicado y los métodos numéricos utilizados, de tal forma que el usuario final de la aplicación informática pueda evaluar si es posible o no emplearla en la simulación de su propio proceso.

De este modelo simplificado se puede extraer también la conclusión de que los programas informáticos que se utilizan necesitan de un equipo multidisciplinar para su creación que incluya, como mínimo, a expertos en el proceso que se quiere evaluar, en las técnicas matemáticas a utilizar y en las tecnologías de computación a usar[3] . Además, el ciclo de vida del software creado es mucho más largo que el del software convencional. Frente a ciclos de semanas o meses de nuevos programas comerciales tradicionales, la creación, verificación y validación de una nueva aplicación informática para los sectores de la ingeniería o de las ciencias básicas puede llevar años. En compensación, también tienen una vida comercial habitualmente muy larga, de decenas de años.

Motivos para usar la HPC

En cualquier caso, la computación de altas prestaciones se ha de entender como una herramienta de trabajo no un fin en sí mismo. Así, existen muchas motivaciones para su utilización, pero destacaremos cuatro principales. La primera es **comprender**. Frecuentemente en las ciencias básicas y aplicadas existen resultados experimentales que no tienen una explicación sencilla debido a la complejidad del sistema estudiado. Sin embargo, se da por supuesto que, para esos sistemas, las leyes básicas del área científica en la que se desarrolla ese proceso se cumplen y que, por tanto, una simulación desde esos primeros principios puede explicar que pasa. La simulación permite por tanto entender el sistema estudiado y, como consecuencia, poder mejorarlo o utilizarlo en el futuro.

3. La intersección de las tres área se suele denominar Ciencia Computacional (Computational Science)

La segunda de las motivaciones es **predecir**, esto es, conocer de antemano el comportamiento del sistema que se está estudiando o analizando. Quizá la aplicación más popular que ha surgido como resultado de la utilización de la computación de altas prestaciones se encuentra en este ámbito: el pronóstico del tiempo. Desde hace años se utilizan modelos meteorológicos para analizar la evolución de la atmósfera y concluir, con cierto riesgo de equivocación, como será el tiempo en los próximos días. Con los años, este pronóstico ha ido mejorando tanto en calidad como en su extensión temporal, permitiéndonos saber con antelación suficiente la temperatura que tendremos, el viento que soplará y su intensidad, la probabilidad de que llueva o haya tormentas. Pero también existen otros ámbitos en donde esa capacidad predictiva es esencial. Por ejemplo, en el área de ingeniería, esta capacidad permite conocer la respuesta del producto o proceso que se está diseñando antes de fabricarlo físicamente. La construcción de un modelo virtual permite analizarlo en detalle y predecir su comportamiento en las condiciones de uso finales y, por tanto, comprobar virtualmente que cumple con los requisitos o solicitaciones demandadas antes de fabricarlo, lo cual genera unos ahorros sustanciales en tiempo y dinero. Asimismo, en un futuro no muy lejano la predicción ayudará a conocer el resultado de una intervención médica antes de realizarla, como puede ser el cambio en la circulación sanguínea al realizar un *by-pass*.

Diagnosticar es otro de los usos que tiene cada vez más impacto, sobre todo social. Se refiere a la determinación de las causas que han producido los hechos que se están analizando. En el campo de la medicina, por ejemplo, los análisis genéticos de última generación permiten determinar de forma fehaciente algunas enfermedades, ayudados por la computación que es capaz de tratar de forma rápida y eficiente la ingente cantidad de información que genera la secuenciación del genoma humano. Un segundo ejemplo es su utilización para el análisis de hechos ya ocurridos con el objetivo de detectar las causas que los han generado o de confirmar las hipótesis que se hayan obtenido previamente.

La última motivación que destacaremos es el **descubrimiento**. Gracias a la capacidad de analizar enormes cantidades de datos o de convertirse en un laboratorio virtual, la computación de altas prestaciones puede utilizarse para generar nuevo conocimiento o para buscar y encontrar nuevas soluciones. El sector farmacéutico hace uso frecuentemente de esta capacidad para descubrir nuevos fármacos. Para ello, analiza virtualmente cómo interactúan pequeños compuestos químicos con las proteínas responsables de la enfermedad que se quiere combatir para anular sus efectos. El objetivo es reducir el número de compuestos a probar en el laboratorio, acortando

los tiempos de desarrollo de nuevos fármacos y, de nuevo, reducir los costes de diseño del mismo. En la industria energética, gracias a la combinación de gran cantidad de datos sísmicos y a los modelos del terreno, ayuda a la detección de nuevos yacimientos petrolíferos, mejorando el nivel de éxito de las prospecciones.

Las herramientas

Volviendo a la definición de la primera página de este capítulo, hasta ahora nos hemos centrado fundamentalmente en el tipo de uso. Sin embargo, esta hace hincapié en el «*uso de servidores, clústeres y supercomputadores*». El concepto de servidor es un computador único construido con una o varias CPUs que se dedica en exclusiva o compartido al tipo de uso que se ha especificado. Aunque muchas veces los servidores están dedicados a estas tareas y no se utilizan frecuentemente como puesto de trabajo, la realidad es que en muchas empresas e incluso entornos de investigación, las propias estaciones de trabajo (o los PC) se utilizan para ejecutar las aplicaciones científico-técnicas. Por tanto, se puede asumir en la definición que estos puestos de trabajo también pueden incluirse en el entorno de HPC. Un clúster es una agrupación de servidores (también denominados nodos) que comparten algún tipo de recurso computacional, como el almacenamiento o el sistema de gestión. Pueden tener una configuración homogénea o heterogénea. En el primer caso, todos los servidores que lo forman tienen la misma configuración hardware, sistema operativo y software (quizá excluyendo uno de ellos que actúa como nodo de cabecera que puede tener una configuración más específica para permitir el acceso directo de los usuarios). Tienen como ventaja la facilidad de gestión y uso, al ser todos los elementos iguales. Como desventaja, la lentitud en la adaptación a nuevas tecnologías y en su crecimiento en caso de necesidad. En el caso heterogéneo, la configuración de cada nodo puede ser diferente, tanto en tipo de CPU como en otros recursos necesarios en computación (almacenamiento, memoria RAM, red de comunicaciones, incluso en sistema operativo). El inconveniente es que son más complejos de gestionar y, en algunos casos, utilizar para el propósito de la computación técnica, pero se adaptan más rápidamente a las novedades técnicas y son más fáciles de ampliar en caso necesario. Incluso un clúster heterogéneo puede combinar partes especializadas en alguna forma de uso.

Finalmente están los supercomputadores. Frecuentemente se solapan los conceptos de computación de altas prestaciones y supercomputación, creando una confusión que se convierte en una barrera psicológica para el acercamiento a la HPC. En realidad, no es fácil distinguir tecnológica-

mente un clúster de un supercomputador, ya que los elementos que los componen habitualmente son los mismos, simplemente se distinguen en el número de elementos que tienen, siendo en el segundo mucho más grande. Por ejemplo, el supercomputador más potente del mundo en junio de 2013 (llamado Tianhe-2, que en castellano significa Vía Láctea-2) tenía más de tres millones de núcleos de cálculo repartidos en 16.000 nodos, que dista mucho de un clúster típico de una empresa o de un centro de investigación, que pueden tener unos miles de núcleos repartidos en decenas o quizá centenas de servidores.

Solo en casos muy específicos, los supercomputadores tienen algún elemento diferente o una arquitectura específica, como la serie SX de la compañía japonesa NEC basada sobre procesadores vectoriales. De hecho, en muchos de ellos es posible que la CPU que se está utilizando no sea la más rápida (considerando la frecuencia o el número de operaciones que podría realizar por cada segundo), sino que su capacidad se consigue agregando varias CPU para la solución de un único problema computacional. La asunción de que el prefijo «super» significa que es mucho más «potente» que mi propio computador, lleva muchas veces a desilusiones de los potenciales usuarios cuando se percatan de que su programa no se ejecuta más rápidamente al no poder utilizar más de un núcleo del procesador.

De hecho, definir lo que se entiende por un supercomputador no es sencillo. Lo más común es distinguir con ese apelativo a aquellos computadores más grandes o rápidos existentes en un momento dado. Por tanto, es más una etiqueta o distinción temporal que una característica en sí del propio computador. ¿Y cómo se sabe que son los más rápidos? La clasificación que se utiliza actualmente es la lista TOP500 que se actualiza cada seis meses (en junio y noviembre de cada año) y que como su nombre indica, incluye los 500 computadores más potentes del mundo (que han solicitado su inclusión enviando la información que se demanda). Dicha lista se confecciona ejecutando un programa informático específico que permite medir el número de operaciones matemáticas simples (sumas y multiplicaciones) que puede ejecutar en cada segundo. En el caso del Tianhe-2 citado anteriormente, era capaz de calcular 34 000 billones de operaciones matemáticas por segundo (en concreto, 33 862 700 000 000 000). Un número sorprendentemente grande para el cual los humanos, si pudiéramos hacer una operación matemática (por ejemplo, una suma) por segundo, necesitaríamos algo más de 1000 millones de años.

Esta integración de servidores en una única infraestructura abre la posibilidad de utilizarla conjuntamente en la resolución de un gran modelo que de otra forma no podría solucionarse en un tiempo razonable. Técnica-

mente es lo que se denomina *capability computing*. En contraste, existe la necesidad de ejecutar modelos más modestos, que no necesitan tantos recursos computacionales pero que frecuentemente sí demandan que se ejecuten las aplicaciones varias veces, incluso miles de veces. Típicamente este modelo de uso de los supercomputadores se denomina *capacity computing*. Existe una tercera opción, similar a la anterior, en donde lo importante es cuantas ejecuciones son posibles por cada fracción de tiempo, ya que lo que se busca es ejecutar miles de casos diferentes, aunque frecuentemente solo necesitan una CPU cada una de ellas. Es la computación de alto caudal (o *High Throughput Computing - HTC*).

Un breve de historia

Es esta capacidad de resolver problemas técnicos, que a los humanos nos sería complicado o imposible de solucionar directamente, lo que ha impulsado históricamente el desarrollo de la computación. En España estamos más acostumbrados a utilizar la palabra ordenador (derivada del latín *ordinator*) que nos acerca más al concepto de colocar o tratar información, curiosamente una de las nuevas demandas para la computación de altas prestaciones relacionadas con el análisis de datos. Sin embargo, el término computador (del latín *computare*, es decir, calcular) está más arraigado en otros países, por ejemplo en los anglosajones. Aunque los computadores son uno de las máquinas que más ha evolucionado en el siglo XX, su historia es muy larga. Una de las primeras calculadoras se la debemos al matemático, físico y filósofo Blaise Pascal llamada Pascalina, inventada en 1642. Funcionaba con ruedas y engranajes para realizar sumas y, con el tiempo, Pascal también fue capaz de incluir la resta. Aunque de origen noble, fue lo que hoy consideramos un emprendedor además de inventor, creando una compañía para su comercialización. Llegó a vender cincuenta, de las cuales aún se conservan nueve. Antes de la Pascalina, existen solo evidencias escritas de otras máquinas de calcular. De entre los genios siempre presentes en la historia de la técnica está Leonardo da Vinci. En sus manuscritos (en concreto en el llamado Códice de Madrid) existe un esquema de una máquina de sumar, descubierto en 1967. Este ingenio es más de cien años anterior al invento de Pascal, aunque probablemente no lo conociera ni se basara en él para su invención. Tampoco se cree que fuera conocedor de otra calculadora de la que solo se conservan esquemas manuscritos, concebida por Wilhelm Schickard, ministro luterano y profesor de astronomía y matemáticas. Según la correspondencia entre él y Kepler, había construido una calculadora que permitía realizar rápidamente sumas, restas, multiplicaciones y divisiones. Era 1623, es decir, casi 20 años anterior a la Pascalina y con capacidades superiores a esta.

Sin embargo, aunque la correspondencia con Kepler incluye información sobre el encargo de la fabricación de una para él, no hay evidencias de que se llegara a construir. Lo más importante es que la utilización de la máquina de Schickard estaba inspirada por la necesidad de hacer cálculos complejos de astronomía, frente a la Pascalina, que fue desarrollada para hacer cálculos monetarios en la hacienda francesa.

Doscientos años después de la calculadora de Pascal, de nuevo la astronomía y la necesidad de contar con tablas matemáticas fiables, llevó a Charles Babbage a proponer en 1822 la construcción de la primera máquina considerada como programable: la máquina diferencial. Esta no llegó nunca a funcionar, a pesar de contar con una importante financiación del gobierno británico. Tampoco se llegó a construir un segundo diseño llamado máquina analítica, más sencilla mecánicamente. Ambos diseños utilizaban el sistema decimal para los cálculos, tarjetas perforadas para introducir los programas (que recordarán los lectores más veteranos) y una impresora para la salida.

No es hasta 1941 que se introduce la aritmética binaria con la construcción en Alemania del primer computador electromecánico, fabricado por Konrad Zuse (el Z3). En este caso, también la computación técnica estuvo detrás de su desarrollo, ya que se fabricó y utilizó para realizar análisis estadísticos de las vibraciones de las alas de los aviones. Más conocida popularmente, y durante mucho tiempo reconocida como la madre de la computadora moderna, es la americana ENIAC, utilizada para el desarrollo de la bomba atómica (aunque pensada inicialmente para el cálculo balístico). Este computador, presentado en sociedad en 1946, puede decirse que fue el detonante para que a partir de la finalización de la Segunda Guerra Mundial, la explosión de la computación haya sido imparable hasta nuestros días. Existen precedentes menos conocidos, y con menos impacto posterior, como la computadora Colossus británica dedicada a romper los mensajes cifrados o también la americana de Atanasoff-Berry.

La historia de la computación se acostumbra a dividir en generaciones que se distinguen por la tecnología utilizada. Así, la primera utilizaba como componente electrónico fundamental la válvula de vacío, lo que implicaba un gran consumo de electricidad y generación de calor, características comunes a los supercomputadores modernos como veremos, aunque por diferentes motivos. La segunda generación utiliza ya el transistor como substituto, reduciendo drásticamente el consumo eléctrico y el tamaño de los computadores. A partir de 1964 se empezaron a utilizar los circuitos integrados, que de nuevo redujeron el tamaño de los computadores y los hicieron más eficientes. Esta tercera generación cambió radicalmente

con la introducción del microprocesador por Intel en 1975, en donde se integran los diferentes elementos de procesado en un solo dispositivo y en donde nos encontramos actualmente, esperando a la próxima revolución tecnológica de la computación, quizá con la introducción comercial de la computación cuántica de propósito general.

Durante este tiempo de evolución tecnológica merece una especial mención Seymour Cray. Nacido en 1925 en Wisconsin (Estados Unidos), este ingeniero eléctrico trabajó casi toda su vida profesional (de más de 45 años) creando las máquinas más rápidas para computación. Esta era su pasión, alejándose de los ordenadores de propósito general. Su cita más conocida es «*Cualquiera puede diseñar una CPU rápida. El truco es construir un sistema rápido*»[4] . Sus sistemas o computadoras estaban siempre orientados al cálculo científico y técnico, diseñados de forma integral. Fue el primero en introducir un computador para cálculo científico utilizando transistores, el 1604 de la recientemente creada compañía Control Data Corporation. En su diseño y construcción utilizaron inicialmente transistores desechados, ya que no tenían dinero para comprarlos nuevos en fábrica, por lo que el diseño era muy robusto para sobreponerse del fallo de uno de ellos. En 1963 lanza el CDC 6600 que incluía la posibilidad de realizar varias operaciones matemáticas simultáneamente (o en paralelo) y que podía hacer nueve millones de operaciones por segundo. Siete años más tarde termina el considerado el primer supercomputador de la historia como tal, el CDC 7600, cuatro veces más rápido que el anterior.

El giro hacia computadores más comerciales le hace abandonar CDC para fundar su propia compañía, Cray Research, dedicada exclusivamente a diseñar y fabricar computadores para cálculo científico. Como resultado crea el primer computador vectorial[5] en 1976, el CRAY-1, que fue un gran éxito comercial (pensaban vender una docena y vendieron realmente más de 80). Incluía todavía la tecnología de circuitos integrados cuando ya se había introducido el microprocesador, debido a que los primeros habían alcanzado la velocidad suficiente para la computación rápida a cambio de necesitar de refrigeración con Freón. De hecho, multiplicaba por cuatro la capacidad de cálculo de su predecesor, llegando a los 170 millones de operaciones. No fue hasta 1985 cuando batió la cifra de los 1000 millones con la introducción del Cray-2, último de la serie que logró comercializar pero con menor éxito. En este caso, debido al gran calor generado, sus circuitos estaban sumergidos en un líquido refrigerante.

Su vida profesional está dedicada casi exclusivamente al diseño de computadores para cálculo científico y técnico. Se enfrentó a muchos de los problemas que todavía necesitan soluciones para seguir creciendo en ca-

4. Traducción del autor de «Anyone can build a fast CPU. The trick is to build a fast system»

5. En el siguiente capítulo se hablará de este concepto de tipo de CPU.

pacidad: unas CPU más eficientes, accesos más rápidos para escribir y leer información, refrigeración de todo el sistema, software de control adaptado, etc. Sus innovaciones durante este tiempo le hacen merecedor del reconocimiento que tiene como padre de la supercomputación.

La contribución española

A pesar de los orígenes de la computación en Europa, los norteamericanos han dominado la expansión de la computación durante el siglo XX y principios del XXI, lo que ha influido notablemente en su historia más conocida. Pero ¿qué hemos hecho los españoles? ¿Hemos contribuido realmente a esta nueva tecnología? En este caso no tenemos que sonrojarnos, ya que tenemos contribuciones importantes de españoles, pero desgraciadamente poco divulgadas. La primera es la calculadora inventada y patentada por el estradense Ramón Verea nacido en 1833. Estudiante de Filosofía en la Universidad de Santiago de Compostela por su interés de seguir una carrera literaria y periodista de profesión, concebía un cambio en la colonización asegurando que «*la espada fue sustituida con la patente*». Su calculadora patentada en 1878 resolvía el problema de la multiplicación en un solo paso, frente a las anteriores que lo hacían a través de sumas sucesivas. El afán que puso en su invención solo estaba motivado por «*contribuir con algo al avance de la ciencia y un poco de amor propio*». Esta máquina de calcular no se llegó a comercializar por la falta de interés de su inventor, pero fue la precursora de otras exitosas como la Millionaire (patentada en 1892), que se puede ver en la exposición que existe en el Centro de Supercomputación de Galicia.

La segunda gran contribución se la debemos ya a un genio no suficientemente conocido en España, y aún menos reconocido, a pesar de sus grandes contribuciones tecnológicas: Don Leonardo Torres Quevedo. Nacido en un pequeño pueblo de Cantabria en 1852, este ingeniero y sobre todo inventor, ha hecho importantes contribuciones a la técnica que empiezan a ser ahora reconocidas mundialmente. Inventó el primer mando a distancia (el telekino) que enviaba instrucciones a un autómata a través de ondas electromagnéticas, al igual que hacemos ahora con nuestros aparatos electrónicos, pero hace 110 años, en 1903. Su pasión por la automática también le llevó a construir un ajedrecista mecánico que jugaba el final de una partida de rey y torre contra rey, donde el contrincante movía libremente este último, aunque siempre perdiera. Quizá exageradamente se considera el precursor de los videojuegos. Pero su contribución en la computación está en la concepción, diseño y fabricación de las máquinas de calcular, como demostración de los conceptos revolucionarios y

avanzados sobre automática expuestos en su obra cumbre «Ensayos sobre Automática». En este trabajo describe las máquinas de calcular como un caso particular del autómata que «*tengan discernimiento*» e «*imiten a los seres vivos, ejecutando sus actos con arreglo a las impresiones que reciben y adaptando su conducta a las circunstancias*». Para conseguir las nuevas máquinas llega a la conclusión de que es necesario cambiar de métodos mecánicos y fundamentalmente analógicos, a las nuevas tecnologías electromecánicas y digitales. Introduce también una representación de los números similar a la actual de coma flotante. La demostración de las ideas de nuestro particular Leonardo da Vinci, se plasma en 1920 con su *aritmómetro* que él denominaba máquina analítica. Este nuevo «computador» electromecánico utilizaba la tecnología de los relés y constaba de elementos similares a los presentes en los ordenadores actuales: unidad aritmética, unidad de control, memoria reducida y dispositivos de entrada y salida. Era capaz, por primera vez, de comparar dos cantidades de varias cifras. Es decir, una revolución en toda regla.

En la actualidad también contamos con investigadores e ingenieros que han contribuido o están contribuyendo significativamente al avance de la computación de altas prestaciones y sus utilizaciones prácticas. En 2007, el profesor Mateo Valero fue reconocido con el premio Eckert-Mauchly otorgado conjuntamente por la ACM (Association for Computing Machinery) y la IEEE Computer Society «*por sus contribuciones seminales en el área de computación vectorial y multihilo, y por sus pioneros nuevos enfoques básicos para el paralelismo a nivel de instrucción*». También en el área del software existen importantes aportaciones, tanto a nivel de herramientas como de aplicaciones específicas. Entre ellas tenemos un record mundial en electromagnetismo computacional, en donde un equipo multidisciplinar formado por investigadores y técnicos de las universidades de Vigo y Extremadura y del Centro de Supercomputación de Galicia resolvió un problema práctico con 1000 millones de incógnitas utilizando el supercomputador Finis Terrae.

Por tanto, la computación de altas prestaciones está desde el principio de la informática, llevando al límite sus capacidades, dando a cambio una herramienta que permite a las empresas competir en un mundo más globalizado y a los investigadores a estar en la vanguardia de la ciencia y la técnica. Conocer sus posibilidades es apostar por un mundo nuevo de oportunidades.

«For over a decade prophets have voiced the contention that the organization of a single computer has reached its limits and that truly significant advances can be made only by interconnection of a multiplicity of computers in such a manner as to permit cooperative solution»[6]

Gene M. Amdahl (1967)

CAPÍTULO 2. LA TECNOLOGÍA DE HPC

Gene M. Amdahl incluyó la frase que abre este capítulo en 1967 cuando trabajaba para IBM en una conferencia titulada «*Validity of the single processor approach to achieving large scale computing capabilities*»[7] en donde se posicionaba a favor de los computadores de una sola CPU. Su argumentación estaba basada en la imposibilidad de utilizar varias CPUs para el 40% de las instrucciones de un programa, asociadas al mantenimiento de los datos y, más relacionado directamente con la computación de altas prestaciones, que los problemas físicos a resolver tenían complicaciones que llevaban a irregularidades difíciles de abordar. Esta disertación dada en la conferencia de primavera de la American Federation of Information Processing Societies es la referencia habitual de la denominada Ley de Amdahl (ver Ilustración 2), aunque realmente no llega a establecerla literalmente. Esta Ley de la computación indica cual es la máxima aceleración (*speedup*) que se puede obtener cuando se paraleliza un código, dependiendo de la fracción de instrucciones que se han de ejecutar en serie. Como consecuencia, para alcanzar un nivel de aceleración muy grande, es necesario que la fracción de instrucciones a ejecutar en serie sea tremen-

6. «Durante más de una década, profetas han opinado que la organización de un único computador ha llegado a su límite y que los avances verdaderamente significativos esta se pueden hacer mediante la interconexión de múltiples computadores que permita una solución cooperativa». Traducción del autor.

7. «Validez del enfoque de un único procesador para lograr capacidades de computación a gran escala». Traducción del autor.

damente pequeña o que se puedan simultanear estas operaciones con otras que se ejecuten en paralelo.

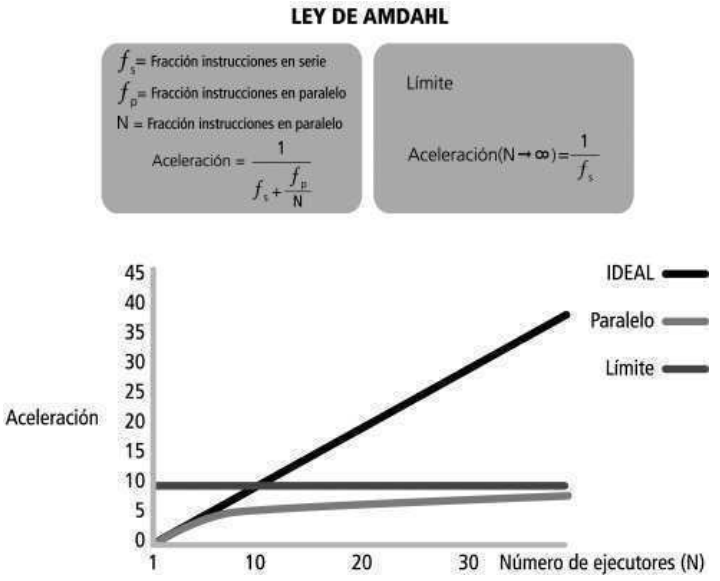


Ilustración 2: Ley de Amdahl. La figura representa el caso cuando la fracción de instrucciones en serie tiene un valor de 0.1 respecto al total de operaciones a realizar.

El hardware

Aunque Amdahl tenía en parte razón y se necesitan cada vez unidades de proceso más rápidas, los computadores actuales que se utilizan en la computación de altas prestaciones son un conjunto de servidores o nodos interconectados que tienen unas necesidades específicas de comunicaciones, de herramientas de programación y de gestión y de entorno de operación. Las unidades de proceso siguen habitualmente el modelo propuesto por el genial matemático húngaro John von Neumann en el diseño del sucesor del computador ENIAC, denominado EDVAC. Este modelo de organización del computador incluye cuatro bloques: una unidad que realiza los cálculos y operaciones lógicas (Aritmétic Logic Unit o ALU); otra de control que carga, interpreta y ejecuta las instrucciones del programa; la memoria (en diferentes niveles, desde registros internos en la unidad de proceso, caché y memoria principal), en donde se almacenan los programas y los datos; y, finalmente, un sistema de entrada/salida que proporciona el interface de comunicación con el usuario del sistema (y otros

sistemas de almacenamiento o comunicaciones, como los discos duros, el teclado, la pantalla, la tarjeta de red, etc.). Esta arquitectura se diferencia fundamentalmente en que los programas son almacenados, frente a otras en donde están «cableados», es decir, definidos en el propio circuito sin posibilidad de cambio (por ejemplo, una calculadora no programable, o el propio ENIAC que necesitaba recableado para resolver cada problema). El almacenamiento de los programas y los datos se realiza en la memoria principal, siendo necesario cargar las instrucciones del programa, interpretarlas y traducirlas a las que están realmente disponibles en la unidad de proceso. Una segunda novedad propuesta por von Neumann y sus colegas en el proyecto EDVAC fue la utilización del sistema binario frente al decimal. Realmente, ambos conceptos (programas almacenados y lógica binaria) ya habían sido aplicados anteriormente en Europa en el desarrollo del computador Zuse Z3 creado por Konrad Zuse en 1941 y destruido dos años más tarde por un bombardeo sobre Berlín. La historia hace de nuevo balancear el péndulo hacia los ganadores.

La introducción del programa almacenado hace que existan dos flujos de información hacia la unidad de proceso: el flujo de instrucciones, es decir, el programa a ejecutar que se divide en miles o millones de instrucciones diferentes, y el flujo de datos que incluye la información que ha de procesar el programa. En computación, una instrucción actúa generalmente sobre uno (por ejemplo, la carga de un dato desde la memoria principal a un registro interno), dos (el producto de dos números, a*b) o tres datos (como la recientemente añadida operación FMA o Fused Multiply-Add que calcula en una sola operación el producto de dos números más la suma de un tercero, a*b+c). En función de cómo se relacionan los flujos de datos y de instrucciones, Michael J. Flynn propuso una división de los computadores en cuatro modelos:

- *Single Instruction Stream —Single Data Stream* (un flujo de instrucciones— un flujo de datos, o SISD). Esta organización no aprovecha el paralelismo de instrucciones ni de datos. Esta es la forma básica de un computador de sobremesa que consta de una sola unidad de proceso que no tiene características avanzadas, como un PC de hace algunos años.
- *Multiple Instruction Streams —Single Data Stream* (múltiples flujos de instrucciones— un flujo de datos, o MISD). Es una organización poco común que se utiliza en casos muy específicos. Sobre el mismo conjunto de datos se aplican varios flujos de instrucciones simultáneamente. No se conoce un microprocesador comercial con esta arquitectura.

- *Single Instruction Stream —Multiple Data Streams* (un flujo de instrucciones— múltiples flujo de datos, o SIMD). En este caso, el mismo conjunto de instrucciones es ejecutado simultáneamente por varias unidades de proceso sobre diferentes conjuntos de datos, explotando la capacidad del paralelismo de los mismos, es decir, que no hay dependencias entre ellos, ni en la entrada o carga desde la memoria ni en la salida o escritura. Entre los ejemplos de este tipo de arquitecturas están los procesadores vectoriales, utilizados mayoritariamente en la computación de altas prestaciones hasta hace unos años y que fueron substituidos por los clústeres, aunque todavía existen importantes centros de supercomputación o de servicio que los siguen empleando. Más actual es la utilización de esta arquitectura en los co-procesadores basados en tarjetas gráficas o GPU (*Graphics Processing Unit*) o la inclusión de unidades vectoriales en los microprocesadores, como las extensiones avanzadas vectoriales (AVX de Intel) o las anteriores SSE (*Streaming SIMD Extensions*).
- *Multiple Instruction Streams —Multiple Data Streams* (múltiples flujos de instrucciones— múltiples flujos de datos, o MIMD). En este caso, cada unidad de proceso tiene su propio flujo de instrucciones que actúa sobre su propio conjunto de datos. Es decir, existen diferentes ejecutores que aplican las instrucciones de forma independiente sobre un conjunto de datos también independiente. Explotan el paralelismo a nivel de hebra o hilo cuando hablamos de organización de un microprocesador. Por ejemplo, el *hyperthreading* de los procesadores de Intel o el *Hardware Threading* de los núcleos del co-procesador Intel Xeon Phi, en donde cada uno de ellos puede ejecutar hasta cuatro hilos[8] independientes simultáneamente, podrían incluirse en esta categoría.

En la actualidad, clasificar un computador completo en una de las categorías de Flynn es complicado, ya que la mayoría de los existentes son una combinación heterogénea, tanto a nivel de microprocesador como en conjunto. De hecho, los servidores o nodos que se utilizan actualmente en HPC no difieren significativamente de los que se utilizan para otro tipo de trabajos informáticos.

Una representación genérica de un computador de altas prestaciones se muestra en la Ilustración 3. Consta habitualmente de un conjunto de nodos individuales dedicados en exclusiva a calcular (zona de cómputo) conectados entre sí a través de una red local de alta velocidad (del orden de decenas a centenares de gigabits por segundo) y baja latencia[9] (del orden

8. Subprocesos ligeros que comparten los recursos y datos creados o utilizados por el proceso principal.

9. La latencia se define como el tiempo empleado en transmitir un mensaje de longitud cero entre dos puntos.

de microsegundos -millonésima parte- o nanosegundos -o diezmillonésima parte de un segundo). Por ejemplo, las redes más habituales que se están instalando en el momento de escribir este libro son de tipo Infiniband, que en su versión FDR tiene un ancho de banda máximo de 163.64 Gigabit por segundo con una latencia típica de 1 microsegundo o inferior. Es decir, un DVD de 4.7 Gbytes - 37.6 Gbits - se podría transferir entre dos nodos conectados por esta red en 0.23 segundos. Un parpadeo.

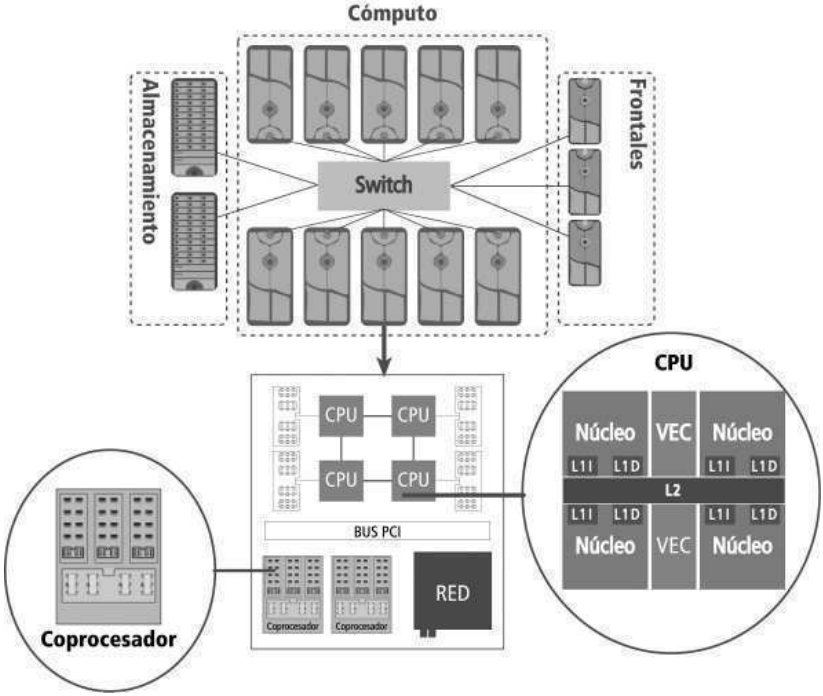


Ilustración 3: Esquema básico de un computador para HPC

Cada uno de los nodos de cálculo se compone a su vez de varios microprocesadores (CPU). La gran integración de los microprocesadores actuales y las limitaciones constructivas, que no permiten por el momento seguir subiendo la frecuencia del reloj debido al calor generado, hacen que cada uno de ellos incluya realmente varias unidades de proceso idénticas que pueden trabajar de forma autónoma (llamados núcleos. Y al microprocesador se le denomina multinúcleo). En función del fabricante y el modelo, su configuración final puede variar de dos a diez núcleos o más. Un punto importante, además de la frecuencia del reloj máxima a la cual funcionan,

es el ancho de banda de acceso a la memoria principal. Este es el cuello de botella actual que impide en muchos casos obtener la máxima eficiencia en la computación, ya que el procesador ha de esperar a que los datos se transfieran de la memoria hasta los registros internos.

Para intentar paliar este problema se ha diseñado una jerarquía completa de memorias, a la cual más rápida, así como técnicas especiales de predicción o programación. De ese modo, para utilizar un dato que está en la memoria, se intenta conjeturar con suficiente antelación su necesidad, ordenando que se copie al nivel más cercano posible. Los niveles más habituales son el registro —ya dentro del núcleo, que siempre existe y que es realmente el que se utiliza para calcular—; la caché de nivel 1 —que puede estar dividida en dos: una para los datos en sí (L1D) y otra para las instrucciones del programa (L1I)—; la caché de nivel 2 (L2 y que en algunos sistemas está compartida entre dos o más núcleos) y finalmente la memoria principal o RAM. Otros sistemas pueden incluir niveles intermedios adicionales, siendo frecuente la existencia de cachés de nivel 3. Según se sube de nivel, el tamaño crece pero disminuyen sus prestaciones de acceso. Así, las cachés de nivel 1 suelen tener un tamaño del orden de kilobytes y necesitar unos pocos ciclos de reloj para mover los datos hasta los registros internos. Las de nivel 2 son frecuentemente de Megabytes y las de nivel 3 de decenas de Megabytes. Las memorias principales varían mucho de uno a otro computador, pero el ratio entre la capacidad de la memoria y el número de núcleos oscila entre 1 y 8 Gigabytes por núcleo. También aquí la tecnología avanza rápidamente, aumentando la capacidad de transmisión de datos a costa de incrementar al mismo tiempo el consumo eléctrico y, por tanto, el calor generado.

Otros dos conceptos importantes sobre la configuración de los computadores están relacionados con la forma en que se accede ella. Lo más habitual es que la memoria principal esté conectada directamente a uno de los microprocesadores (como en la Ilustración 3). Aunque todos los núcleos dentro del nodo pueden acceder a cualquier dato que esté en la memoria física— siempre que le esté permitido, claro—, el tiempo para acceder a él dependerá de si está en la memoria conectada directamente o si lo tiene que cargar de un banco de memoria que está asociado a otro microprocesador. En este caso, se dice que el modelo es NUMA, del inglés *Non Uniform Memory Access* (o Acceso a Memoria No Uniforme). Por el contrario, si los bancos de memoria no están conectados directamente a los microprocesadores sino que se conectan a un elemento intermedio, todos ellos podrán acceder a cualquier dato directamente necesitando el mismo tiempo para ello. Se denominada entonces UMA (*Uniform Memory Ac-*

cess). Lo más común en la actualidad es tener nodos con configuraciones NUMA. Como consecuencia de este modelo de acceso a los datos, con jerarquías de memoria para leer o escribir, es necesario mantener también la coherencia entre las diferentes copias del mismo dato que están en las cachés de memoria, existiendo técnicas para ello que pueden de nuevo limitar la eficiencia de la ejecución de los programas de computador. El caso más habitual es el ccNUMA o *cache coherence* NUMA.

Complementariamente, un microprocesador puede incluir una unidad vectorial (SIMD según la clasificación de Flynn), compartida o no entre los diferentes núcleos. Cada nuevo modelo incrementa el número de datos que se pueden procesar simultáneamente. La más reciente tiene un tamaño de entrada de 512 bits, es decir, 8 números de 64 bits o 16 de 32 bits, que le permitirían calcular en cada caso 4 u 8 operaciones de forma simultánea.

Adicionalmente, un nodo de computación puede tener co-procesadores ayudando en alguno de los cálculos. A pesar de que esta inclusión es reciente —aunque los más veteranos recordarán los co-procesadores matemáticos de los antiguos PC—, se está imponiendo rápidamente. Frente a los núcleos, que podríamos considerar como SISD, los co-procesadores más habituales están basados en las tarjetas gráficas, teniendo miles de núcleos en cada una de ellas. La diferencia es que estos están agrupados en conjuntos que ejecutan sobre datos diferentes las mismas instrucciones de forma sincronizada, es decir, siguen de nuevo el modelo de SIMD, como las unidades vectoriales. Por ejemplo, en las tarjetas o GPU de la compañía NVIDIA, las ejecuciones se dividen en bloques de hilos. Estos se ejecutan en grupos simultáneos de 32 que repiten las mismas instrucciones, pero sobre datos diferentes. En cada momento, miles de estos hilos se están ejecutando simultáneamente. A pesar de que las frecuencias utilizadas son bajas comparadas con los núcleos de propósito general, la gran cantidad de ellos permite calcular rápidamente operaciones complejas, siempre que no haya dependencias entre ellas. Por ejemplo, para procesar un cambio de color en una fotografía, cada pixel se puede cambiar utilizando solo información de los píxeles de alrededor, aprovechando perfectamente el paralelismo que ofrecen estos sistemas. La penalización en este caso es la complejidad de la programación, sobre la que se está trabajando y que comentaremos más tarde.

Otros modelos de co-procesadores incluyen también multitud de núcleos, como los Intel Xeon Phi, con 61 núcleos a día de hoy, pudiendo ejecutar cada uno de ellos hasta 4 hilos, pero en este caso, independientes. Es de-

cir, se pueden ejecutar en este co-procesador programas que hagan 244 tareas simultáneamente.

Un segundo componente importante es el almacenamiento permanente de datos. Los programas de computación de altas prestaciones pueden acceder a gran cantidad de datos o escribir como salida enormes ficheros. Estos conceptos dichos así son bastante relativos, pero es frecuente que tengan un tamaño entre Gigabytes y Terabytes. Para manejarlos de forma eficiente y no penalizar la ejecución, estos sistemas de almacenamiento están conectados a la red de alta capacidad descrita anteriormente y además escriben o leen en paralelo de varios discos, aumentando la velocidad de acceso a la información. Entre los más utilizados está el sistema de ficheros LUSTRE de Intel, aunque existe una versión de software libre, y el GPFS de IBM, entre otros.

Finalmente, el tercer componente habitual son los nodos de acceso interactivo (zona de frontales en la Ilustración 3). Estos son cada vez más necesarios para poder definir los problemas a ejecutar así como hacer los análisis de los resultados, compilar los programas, controlar las ejecuciones, etc. Sus características suelen ser similares a los de los nodos de computación, con la misma arquitectura del microprocesador pero quizá con más memoria RAM y algo más de frecuencia.

Por supuesto, el número de elementos de la configuración final del computador depende del uso al que se dedicará. Así, para una pequeña compañía todo el sistema podría estar integrado solamente en un nodo o dos. Frente a los grandes centros de supercomputación que tienen millones de núcleos. Por ejemplo, la configuración del Tianhen 2 chino, tiene un total de 3 120 000 núcleos repartidos en 16 000 nodos de cálculo, cada uno de ellos con 2 procesadores Intel Ivy Bridge de 12 núcleos y 3 co-procesadores Intel Xeon Phi de 57. Sumando todos los nodos, tiene 1 404 000 Gigabytes de memoria RAM. Adicionalmente, como nodos de acceso, cuenta con otros 4096 procesadores de diseño propio, cada uno con 16 núcleos. Un gran problema de estos grandes supercomputadores es que, debido a su elevado número de componentes, el consumo eléctrico también es muy importante. En el caso del computador chino es de 17.8 Megavatios (si se le añade la refrigeración, son necesarios 24 MW). Por supuesto, no es necesario tener un computador de este tamaño para poder iniciarse en la computación de altas prestaciones ni tampoco es necesario para la mayor parte de los usos industriales, comerciales o científicos. Solamente en grandes retos científicos o técnicos se podrá aprovechar toda su capacidad.

El Software

Para poder aprovechar el hardware de nuestro entorno de computación, es necesario utilizar el software adecuado. La primera capa de software que se necesita es el sistema operativo, es decir, aquel que se encarga de gestionar el acceso al hardware de cada uno de los nodos y proporcionar servicios a las aplicaciones de usuario. Aunque en los ordenadores personales, y en muchos servidores, el dominante es Microsoft Windows, en el caso de la computación de altas prestaciones el más habitual es LINUX. Aunque Microsoft tiene una versión específica para computación de altas prestaciones (Microsoft Windows HPC), todavía no ha tenido una gran aceptación por la comunidad usuaria. Entre las funciones que realiza el sistema operativo, es importante que gestione eficazmente la memoria RAM, el caché de escritura y lectura de los ficheros, así como la asignación de la CPU para la ejecución de las aplicaciones.

Cuando el computador está compuesto por más de un servidor, como es el caso genérico que se ha presentado, y está compartido por más de un usuario, la asignación de los recursos a un programa suele contar con la ayuda de un segundo elemento de software denominado gestor de colas o balanceador de carga. Este los reparte entre las diferentes solicitudes de ejecución pedidas por los usuarios. Las políticas para este reparto entre las diferentes demandas son bastante complejas y pueden ser desde colas estilo FIFO (la primera petición que llega será la siguiente en ejecutarse) hasta equitativas, priorizadas, con reserva, etc. El objetivo será cumplir con las expectativas del usuario mientras se magnifica la eficiencia del computador y su aprovechamiento, siempre de acuerdo con las reglas de negocio que se definan. Entre los balanceadores de carga más comunes actualmente están el Slurm y Grid Engine, ambos software de código abierto y uso libre, o comerciales como LSF y LoadLeveler.

La siguiente y última capa en esta jerarquía es la aplicación. Esta ha de resolver eficientemente y eficazmente el problema planteado. Se consigue a través de la combinación de dos vías: la buena programación y la utilización correcta de los compiladores. Estos últimos transforman el código fuente del programa en instrucciones que puedan ser ejecutadas en los diferentes elementos del computador (las CPU o los co-procesadores). El proceso de compilación es importante, ya que en función de las opciones que se utilicen, se pueden obtener mejoras muy sustanciales en los tiempos de ejecución del programa, incluso de varios órdenes de magnitud. Claro que debido a que muchas opciones de compilación pueden cambiar internamente el código, es necesario que no solo se ejecute rá-

pidamente sino que los resultados sean los correctos, pasando el proceso de verificación y validación descrito en el primer capítulo.

Por otro lado, la programación correcta de las aplicaciones se ve facilitada por la existencia de *librerías*, bibliotecas de programas o las APIs (*Application Programming Interface* — Interface de Programación de Aplicaciones) que implementan los algoritmos más habituales. En la computación científica y técnica con más arraigo, las operaciones más comunes son las de álgebra lineal que se realizan sobre vectores, matrices o combinación de ambas. Debido a ello, en 1973 R.J. Hanson, F.T. Krogh y C.L. Lawson propusieron un conjunto de funciones que simplificaran la realización rutinaria de esas operaciones matemáticas y que estuvieron disponibles por primera vez en 1977 bajo la denominación de Basic *Linear Algebra Subprograms* (Subprogramas de Álgebra Lineal Básica o BLAS, como se conoce en la actualidad). Las versiones iniciales estaban pensadas y desarrolladas para los códigos escritos en FORTRAN (lenguaje de programación del que hablaremos más adelante) y se han extendido y, sobre todo, perfeccionado notablemente. De hecho, cada fabricante que se introduce en el campo de la computación de altas prestaciones desarrolla su propia versión de estas *librerías*, adaptada a las características específicas de sus procesadores. Con el tiempo, otros algoritmos matemáticos básicos frecuentemente utilizados se han ido estandarizando e incorporando a los paquetes de software de los fabricantes como LAPACK (*Linear Algebra Package*), generadores de números pseudoaleatorios o FFTW (*Fast Fourier Transform in the West*). Sobre estas bibliotecas de funciones básicas, existen otras que encapsulan algoritmos más complejos orientados a la resolución de un tipo de problema específico. A modo de ejemplo, PETSc (*Portable, Extensible Toolkit for Scientic Computation*) es una biblioteca para resolver problemas modelados a través de ecuaciones en derivadas parciales. La ventaja de tener estas API es evidente, ya que evita que cada programador tenga que escribir de nuevo los algoritmos ya conocidos y ampliamente divulgados, teniendo además versiones muy optimizadas. Pero es importante resaltar que se ha de seleccionar adecuadamente el algoritmo a utilizar, verificando que el problema modelado cumple las condiciones matemáticas necesarias.

La programación de las aplicaciones de computación de altas prestaciones requiere un conocimiento especializado importante en el área de interés, muchas veces conseguido solo a través del trabajo conjunto de programadores técnicos, matemáticos y especialistas en el campo que se pretende abordar. Esta colaboración ha pasado a denominarse Ciencia Computacional. Para llevar el modelo conceptual a la computadora es necesario

escribirlo de tal forma que pueda posteriormente ser tratado por los compiladores. El lenguaje más utilizado, debido a razones históricas y su adaptación a la computación numérica y a la sintaxis de las matemáticas, es el FORTRAN (*Formula Translating System*). Propuesto en 1953 por John W. Backus a sus superiores de IBM, ha ido evolucionando con el tiempo hasta su más reciente estandarización de 2008. Dado que es el primer lenguaje de programación orientado a la programación científica y a la existencia de compiladores muy eficaces, los programas escritos en FORTRAN tienen un excelente comportamiento. En los últimos años se han incorporado otros lenguajes de programación, como el C o más recientemente el Python. Incluso, en muchos de los paquetes de software es muy común utilizar dos o más. Además, existen otros paquetes pensados para facilitar el modelado de los problemas o el análisis de datos, como son MATLAB (o sus versiones de software libre como Octave o Freemat) o R para análisis estadístico.

Paralelización

Independientemente del lenguaje de programación utilizado, para aprovechar de forma eficiente la configuración de un computador moderno, es necesario que las aplicaciones puedan utilizar varias CPU dentro de los límites descritos al principio de este capítulo. Las técnicas más habituales son:

- División en tareas independientes. En este caso, el problema a resolver se puede dividir en subproblemas independientes en donde es posible agregar el resultado final. Se ejecutan tantas copias de la misma aplicación como se demanden, no siendo necesario además de que se realice de forma simultánea, ya que no necesitan una coordinación entre ellas.
- Programación multihilo. Se ejecuta solo un programa que puede subdividirse durante la ejecución en pequeños subprogramas o hilos que comparten los datos y recursos de memoria RAM. Cada uno de ellos puede ejecutarse en uno de los núcleos de las CPU dentro de un único nodo[10].
- Programación distribuida. Se distribuyen entre los diferentes nodos del computador varias copias del programa (o programas). Cada una de ellas tiene una parte del problema a resolver, por lo que han de ejecutarse simultáneamente y necesitan coordinarse entre sí. En este caso, la red de comunicaciones entre los nodos

10. Aunque existen soluciones técnicas que permiten la utilización de este modelo de programación usando varios nodos, no es muy habitual su uso.

es un factor muy importante. Por supuesto, este tipo de ejecución también se puede utilizar dentro de un único nodo.

Con la irrupción de las CPU multinúcleo, es cada vez más frecuente que se mezclen los dos últimos modelos, con el objetivo de aprovechar más eficazmente las configuraciones de hardware disponibles. Para facilitar la escritura de las aplicaciones, se utilizan extensiones a los lenguajes de programación que simplifican de nuevo la labor del programador. Para el caso de la programación multihilo, la más frecuente es el estándar OpenMP, creado en 1997 para su utilización en computadores de memoria compartida. Este es un conjunto de directivas de compilación, bibliotecas de programas y variables de entorno disponibles para los lenguajes FORTRAN y C/C++. Las directivas de compilación son extensiones al código del programa que el compilador interpreta cuando genera el ejecutable, en caso de que se quiera que su ejecución pueda realizarse en paralelo. Las bibliotecas o API incluyen funciones o subprogramas que le permiten al programador tener un control más fino durante la ejecución, por ejemplo, preguntando por el número de hilos disponibles en ese momento para adaptar mejor el algoritmo. Finalmente, las variables de entorno permiten definir algunos parámetros que ajustan el programa al hardware disponible o al tamaño del problema a resolver en el arranque de la aplicación, como puede ser el número de núcleos a usar o la política de balanceo de carga en los lazos paralelizados. La ventaja del OpenMP es la facilidad de uso y la rapidez de desarrollo frente a otras soluciones de programación paralela.

Para el caso de la programación distribuida, la solución técnica más extendida es el MPI (*Message Passing Interface*). Su desarrollo se gestó en 1992, culminándose en noviembre de 1993 cuando se presentó en la conferencia *Supercomputing* de ese año y, posteriormente, se publicó su primera versión en mayo de 1994. Es una amplia especificación de interfaces para la programación de aplicaciones, generalmente distribuidas entre los diferentes nodos del computador, de tipo MIMD de Flynn. Es decir, la ejecución coordinada de uno o varios programas, cada uno de ellos con su propio espacio de memoria RAM que intercambian información entre ellos. Lo más habitual es que el programa sea el mismo, teniendo solo un conjunto de datos diferenciado para cada copia que se ejecute, aunque el modelo no impide que sean aplicaciones diferentes. Este paradigma de programación implica que, cuando se utiliza distribuyendo los procesos entre diferentes nodos del computador, las comunicaciones tienen que ser rápidas, para reducir los tiempos de espera y por lo tanto mejorar la eficiencia de la ejecución.

Con la introducción de las CPU tipo multinúcleo, la programación híbrida mezclando OpenMP con MPI se ha hecho cada vez más común. Sin embargo, la dificultad del desarrollo de aplicaciones con MPI o en modelos híbridos ha hecho que se busquen otros paradigmas de programación que permitan una mayor productividad y que no dependan tanto de la configuración final del hardware a utilizar. Es decir, que la misma aplicación se pueda ejecutar de forma distribuida o en memoria compartida sin necesidad de tener este hecho en cuenta en el momento del diseño y desarrollo de la aplicación. El modelo más extendido es el PGAS (*Partitioned Global Address Space o Espacio de Direccionamiento Global Particionado*), en donde se asume que los datos están divididos entre los procesos o hilos, pero que son accesibles globalmente como si estuvieran en memoria compartida. Este paradigma de programación paralela es la base para nuevos lenguajes de programación o extensiones a los existentes, como UPC (*Unified Parallel C*), Coarray FORTRAN o Chapel. Como en otros lenguajes en fase de consolidación, han de demostrar que la programación es más productiva y que la eficiencia en la ejecución es similar a la conseguida con otros más asentados. Además, dada su juventud, tienen el hándicap de no contar con suficientes bibliotecas de programas o API disponibles que ayuden al programador. Por ejemplo, las librerías BLAS comentadas anteriormente y básicas para el desarrollo de muchas aplicaciones científicas y técnicas no han tenido una versión para UPC hasta hace bien poco. Una primera especificación e implementación ha sido realizada conjuntamente por la Universidad de A Coruña y el Centro de Supercomputación de Galicia con el apoyo de HP en 2009 (denominada UPCBLAS).

La utilización de co-procesadores también ha generado una efervescente actividad para poder desarrollar aplicaciones que usen eficientemente estos sistemas y, a ser posible, poder emplear el mismo código en diferentes plataformas. Con la aparición de las GPU como elementos para calcular, la compañía NVIDIA propuso CUDA. Es un conjunto de extensiones al lenguaje de programación C que permite aprovechar las capacidades de las tarjetas gráficas fabricadas por esta compañía (aunque existe también la versión para FORTRAN desarrollada conjuntamente por NVIDIA y la empresa desarrolladora de compiladores PGI). Estas extensiones permiten mover los datos entre la memoria RAM del computador y la propia de estas, así como ejecutar parte del código en C en ellas. A diferencia de lo que ocurría en UPC, cuenta además con las API básicas como cuBLAS (que implementa BLAS) o cuFFT.

Dado que CUDA está diseñada exclusivamente para las GPU de su fabricante, se han propuesto otras alternativas que permiten simplificar la programación, buscando la estandarización, más productividad y flexibilidad. La primera en aparecer fue el OpenCL en 2008. Es un conjunto de especificaciones, API y entorno de ejecución basadas en el lenguaje C con una estructura muy similar a CUDA. La ventaja es que permite la programación paralela no solo en tarjetas gráficas sino en otros entornos hardware como las CPU multinúcleo, los procesadores digitales de señal (DSP), o las FPGA (dispositivos en donde se puede escribir el programa directamente en el hardware).

Más reciente es OpenACC. En este caso está orientado exclusivamente a la utilización de co-procesadores. Se basa en directivas de compilación que facilitan el movimiento de los datos entre el computador anfitrión y el co-procesador así como controlan la ejecución del código en estos últimos. De esta forma, el mismo programa se puede utilizar con o sin ayuda de los co-procesadores. Sin embargo, dado que está diseñado solo para el uso de estos, es más limitado que OpenCL. La ventaja frente a este es que permite usar tanto C como C++ y FORTRAN.

Es problemático decir cuáles de estas iniciativas sobrevivirá y llegará a ser la dominante. Más aún cuando el estándar OpenMP en su última versión ha incluido modificaciones para poder programar los co-procesadores siguiendo el mismo modelo de CUDA u OpenACC.

Y el futuro

El modelo de computación que tenemos hoy en día basado en los conceptos descritos por von Neumann tiene sus limitaciones. A pesar de las mejoras tecnológicas introducidas en los procesadores, la posibilidad de incluir más elementos en ellos gracias a la gran reducción de tamaño de los circuitos básicos, o a la investigación en nuevos materiales, el modelo parece que tendrá un límite no muy lejano, en donde no se podrán seguir mejorando las prestaciones de una sola CPU. No todos los posibles programas que queremos ejecutar permiten una versión paralela, por lo que su mejora implica un cambio en los algoritmos utilizados o, como hasta ahora, una incremento en la eficiencia o velocidad de los procesadores. Otra posibilidad es tener un cambio radical en el modelo, como propone la computación cuántica. Este cambio de paradigma se encuentra todavía en sus albores y está basada en los conceptos de la física cuántica.

En la computación actual, un dato (por ejemplo, un número) se representa internamente como un conjunto de bits (una cifra formada solo por unos

y ceros, como 011 que representa el número 3). Cada conjunto de bits representa un único dato, sobre los cuales hay que operar para obtener la solución deseada, probablemente teniendo que hacer muchas operaciones sobre muchos datos. Pero, ¿qué pasaría si se pudiesen codificar en un solo conjunto todas las soluciones posibles y al operar sobre él una sola vez obtener el resultado buscado? Probablemente la computación sería más rápida y eficiente. Esto es lo que, de forma simplificada, propone la computación cuántica en donde en vez de bits se utilizan qubits (quantum bits, en donde cada «bit» puede ser 0, 1 o la superposición de ambos) aprovechándose de las propiedades cuánticas de la materia en donde todos los estados son posible simultáneamente hasta que se elige uno de ellos (la solución) cuando se opera sobre ellos.

Aunque un computador cuántico de propósito general todavía parece algo lejano, sí existe uno que resuelve problemas de optimización utilizando este concepto. Comercializado por la empresa canadiense D-Wave, tiene un procesador basado en tecnologías de superconductividad que trabaja a temperaturas cercanas al cero absoluto (20 miliKelvins o casi -273 grados centígrados) para poder tener los efectos cuánticos necesarios. En la actualidad puede resolver problemas que necesiten hasta 512 qubits.

Otra posible vía es comprender mejor como funciona nuestro cerebro, que a fin de cuentas es un magnífico y eficiente computador de bajo consumo. Inspirándose en esta posibilidad, IBM ha desarrollado un nuevo tipo de procesador que reproduce el comportamiento de las neuronas y sus conexiones sinápticas, pudiendo emular su comportamiento. Todavía está en fase experimental, pero su introducción representaría también un cambio radical en el paradigma de la computación.

La solución que se impondrá en el futuro, entre estas u otras propuestas que se están investigando, es difícil sino imposible de predecir. Es posible que se encuentren soluciones a las limitaciones del modelo actual o que finalmente aparezca otro paradigma que resulte más eficiente y permita seguir avanzando. Pero mientras no ocurra, lo más importante es aprovechar adecuadamente la computación que tenemos ahora para avanzar en el conocimiento o competir mejor en este mundo globalizado. Son las aplicaciones de la computación lo que realmente cuenta y nos aporta valor.

«Los computadores pueden hacer muchas cosas. Pueden sumar millones de números en un abrir y cerrar de ojos. Pueden burlarse de los grandes maestros del ajedrez»

Richard P. Feynman. «Conferencias sobre computación»

CAPÍTULO 3.
APLICACIONES DE LA COMPUTACIÓN DE ALTAS PRESTACIONES

Desde sus comienzos, la evolución de la computación ha estado guiada por la necesidad de dar satisfacción a las necesidades o las inquietudes de las personas y de las organizaciones. Es una poderosa herramienta que permite encontrar soluciones a cuestiones cada vez más complejas y más rápidamente, favoreciendo que se pueda competir eficazmente en un mundo globalizado. Las posibilidades que proporciona son tan variadas como el conocimiento existente en la ciencia y la técnica. La inventiva de los emprendedores lo transforma en realidades, expandiendo su utilización a un número cada vez mayor de campos empresariales, científicos y técnicos. Hay pocas actividades en donde no se esté utilizando la computación de altas prestaciones, desde la ingeniería hasta la sociología pasando por biología, medicina, física, química, meteorología, climatología, cosmología, finanzas, economía, energía, etc. Dada la variedad y cantidad de aplicaciones existentes, seleccionar un pequeño conjunto de ellas es, cuando menos, arriesgado, pero necesario para que se pueda comprender mejor las capacidades de esta tecnología.

En el primer capítulo del libro se exponían cuatro motivaciones para usar la computación de altas prestaciones: comprender, predecir, diagnosticar y descubrir. En este se ampliará la breve descripción de los ejemplos que

las justificaban y se añadirán unos pocos más para mostrar la manera en que se puede aprovechar la capacidad de la HPC para cada una de esas cuatro motivaciones.

Comprender

Nuestro cerebro es una poderosa máquina que controla nuestro cuerpo, almacena nuestros recuerdos y conocimientos, genera nuevas ideas, etc. Todo en un solo sistema. Este magnífico computador biológico, altamente eficiente, siempre ha despertado el interés de los científicos más eminentes para intentar comprender cómo funciona. La computación está ahora contribuyendo activamente a desentrañar su funcionamiento en uno de los retos científico-técnicos más ambiciosos del momento: el proyecto Blue Brain. Liderado por el investigador Henry Markram de la Ecole Polytechnique Fédérale de Lausanne en Suiza, con la colaboración de IBM, demostró la posibilidad de ensamblar una pequeña región del cerebro de un ratón y simular su comportamiento. El objetivo final propuesto por el profesor Markram es «*simular los cerebros de los mamíferos con un alto grado de precisión biológica y estudiar los pasos involucrados en la aparición de la inteligencia humana*». Su éxito inicial se ha convertido ahora en uno de los proyectos estrella europeos en investigación llamado Human Brain Project que incluye el desarrollo de un entorno que permite la simulación del cerebro, además de otros cinco: neuroinformática, para la gestión de la información relevante existente sobre el cerebro a todos los niveles, desde el genoma hasta los datos estructurales y funcionales; computacional, en donde ejecutar los modelos del cerebro o acumular la información recopilada; informática médica, que registrará datos de pacientes con diferentes enfermedades relacionadas con el cerebro, como el Alzheimer, para su análisis; computación neuromórfica, para el desarrollo de nuevos sistemas de computación basados en redes neuronales; neurorobotics, en donde se experimenta con robots virtuales conectados a modelos del cerebro.

La simulación del cerebro humano completo todavía es inalcanzable, dado que tiene más de 80 000 millones de neuronas. La zona del cerebro de los humanos de interés para el proyecto es el neocortex, correspondiente a la parte externa y que tiene pliegues para aumentar su superficie. Este está dividido en secciones de aproximadamente 2 milímetros de profundidad y medio de anchura llamadas columnas neocorticales, que a su vez están formadas por una densa maraña de neuronas (unas 60 000 en humanos, 10 000 en ratones). En los trabajos del equipo suizo se han modelizado los diferentes tipos de neuronas, simulando su comportamiento eléctrico como si fuera un cable conductor utilizando el software NEURON desarro-

llado en la Universidad de Yale. Las neuronas posteriormente se enlazan entre sí a través de las conexiones sinápticas hasta construir un modelo de la columna neocortical, el equivalente a un microcircuito. El siguiente paso es comunicar varias columnas para formar un mesocircuito y finalmente construir un modelo completo del cerebro. Dado que la simulación de una sola neurona requiere la resolución numérica de una ecuación diferencial, es un trabajo computacionalmente intensivo que necesitaba el equivalente a un ordenador portátil de 2009. Para hacer la simulación de una sola columna de un ratón, se utilizó inicialmente un computador del tipo Blue Gene de IBM con varios miles de procesadores, colocando varias neuronas en cada uno de ellos. Los avances en la computación y en las herramientas desarrolladas en el proyecto han permitido que se avance hasta obtener microcircuitos de 33 000 células que se han podido conectar para formar mesocircuitos de hasta un millón de neuronas.

Como resultado de estos logros técnicos, se han podido realizar experimentos virtuales para comprender cómo se articula el cerebro. Así, se han conseguido avances en la comprensión de cómo se conectan las diferentes neuronas o cómo se generan las ondas cerebrales que se utilizan en el diagnóstico de algunas enfermedades. El potencial de la tecnología desarrollada es amplísimo, aunque necesita también una gran infraestructura computacional que no está al alcance de todos los investigadores o emprendedores.

No siempre es necesario impulsar un proyecto tan ambicioso o complejo como el descrito anteriormente. De hecho, una gran mayoría de las simulaciones que buscan comprender las causas de un fenómeno no se resuelven en enormes computadores. Un campo en donde la computación de altas prestaciones está muy arraigada es la ciencia de materiales o de la materia condensada. Para comprender los fenómenos observados experimentalmente, y a pesar de que las herramientas experimentales para estudiar los materiales son cada vez más sofisticadas y potentes, es necesario recurrir a la solución numérica de las ecuaciones de la física cuántica, ya que los objetos de interés en este caso son los átomos o incluso solo el subconjunto de los electrones que pasan a estar bajo la influencia de varios de ellos. Conocer cómo se comportan estos electrones en cada caso, permite comprender las propiedades físicas que tiene el material: si es conductor o aislante, si es magnético o no lo es, etc.

Uno de estos materiales de gran interés tecnológico son los óxidos de manganeso, (manganitas) cuya unidad básica está formada por tres átomos de oxígeno, uno de manganeso y otro de lantano. Estos compuestos cambian radicalmente su resistencia eléctrica cuando se le aplica un cam-

po magnético en un fenómeno llamado Magnetoresistencia Colosal. Su adecuado control permitiría desarrollar sistemas de almacenamiento más densos y, por lo tanto, con más capacidad. Si parte de los lantanos de la estructura cristalina se substituyen por calcio, llega un momento en donde el material pasa de ser aislante a ser conductor. El motivo de este cambio no se comprendía hasta que se utilizaron los métodos computacionales. En 2008, un pequeño equipo formado por investigadores de la Universidad de Santiago de Compostela y del Centro de Supercomputación de Galicia encontraron, utilizando el paquete de software Wien2K desarrollado en la Universidad Técnica de Viena y una parte el computador Finis Terrae, que el motivo de dicha transición se debía a la distribución irregular de los átomos de calcio en la estructura cristalina. Las inhomogeneidades del material son por tanto más relevantes para explicar el comportamiento físico del material que la fracción de átomos de calcio en sí.

Predecir

Estamos acostumbrados a ver en televisión, a través de Internet o de nuestros teléfonos móviles el pronóstico del tiempo para los próximos días. La decisión de ir a la playa o no, de hacer una excursión o una comida campestre, de salir a navegar puede depender de la información que nos suministran sobre si estará despejado o lloverá. Pero también es muy relevante para la actividad económica, ya que puede condicionar la proporción de electricidad que se generará por el viento, si es necesario activar planes preventivos para evitar colapsos en ciudades o aeropuertos debidos a la nieve o prevenir a la población por olas de calor. Estamos tan acostumbrados a recibir esta información y asumimos que siempre acertará, que cuando las previsiones fallan aparecen quejas de los afectados.

La previsión del tiempo está basada hoy en día en la experiencia de los meteorólogos junto con la ejecución de modelos de evolución de la atmósfera cada vez más sofisticados y precisos. Los métodos están basados en la resolución de las ecuaciones que describen el comportamiento de los gases y su termodinámica para conocer la evolución de la presión, la temperatura, la humedad o la velocidad de movimiento (es decir, los vientos). Las raíces de la predicción meteorológica se empezaron a vislumbrar a finales del siglo XIX y principios del XX gracias a los trabajos del meteorólogo americano Cleveland Abbe y del noruego Vilhelm Bjerknes, siendo los primeros intentos hechos a mano por Lewis Fry Richarson un pequeño fracaso, ya que al calcular la evolución de la presión atmosférica sobre dos puntos de Europa central, las diferencias entre lo observado y lo medido eran enormes. Sin embargo, con la aparición de los computadores

después de la Segunda Guerra Mundial fue posible plantearse de nuevo la predicción numérica, siendo de nuevo los europeos emprendedores en este ámbito, arrancando en Suecia la primera predicción operacional (es decir, de forma rutinaria) en 1954.

Los modelos computacionales actuales tienen, evolucionados, los mismos principios que propuso Bjerknes. Primero se hace un diagnóstico del estado de la atmósfera a través de los datos existentes, para posteriormente hacer un pronóstico a futuro resolviendo el modelo atmosférico. Actualmente existe una abundancia de datos del estado de la atmósfera diarios, tanto de sensores en tierra como provenientes de satélite, que se asimilan en el modelo para obtener las condiciones iniciales de partida. Posteriormente, se resuelven las ecuaciones de evolución de la atmósfera. Para ello, la zona de interés se divide en pequeñas secciones formando una rejilla o malla. Las secciones horizontales de esa rejilla son cuadrados como los de un tablero de ajedrez con casillas desde 1 kilómetro de lado a varias decenas dependiendo, entre otros factores, del área que se quiere cubrir y de la capacidad computacional existente. Así, las predicciones regionales o locales se suelen producir con mallas pequeñas, mientras los más amplios las tienen más grandes. La ventaja es que las soluciones de los segundos, realizados por centros con abundantes medios tecnológicos y técnicos, se utilizan como entrada en los primeros. Por ejemplo, el European Centre for Medium-Range Weather Prediction realiza una predicción global a 10 días con una resolución horizontal de 40km mientras que la Agencia Estatal de Meteorología utiliza solo 2.5km, pero utiliza los resultados del primero para introducir en la zona estudiada lo que se ha predicho fuera de ella, enlazando la predicción global con la local.

Uno de los obstáculos que tiene la predicción numérica del tiempo es que las ecuaciones a resolver son no lineales. Esto quiere decir que pequeñas variaciones en los valores iniciales pueden generar soluciones muy diferentes y por lo tanto, pronósticos divergentes. Esto llevó en su momento al meteorólogo Edward N. Lorenz en 1972 a plantear su ya famosa pregunta «*¿puede el batir de alas de una mariposa en Brasil generar un tornado en Texas?*» El incremento de la capacidad de los computadores permite hoy abordar este problema solucionando numéricamente varias veces el modelo (habitualmente de peor resolución horizontal), pero cambiando ligeramente las condiciones iniciales. De esta forma, es posible generar decenas de predicciones diferentes y proporcionar en vez de un único valor (*lloverá o no lloverá*), una probabilidad (*existe un 80% de probabilidades de que llueva*).

Las predicciones del tiempo descritas necesitan importantes recursos humanos y, en algunos casos, computacionales. Por ello se realizan mayoritariamente por entidades gubernamentales. Sin embargo, sus predicciones se pueden utilizar para otros fines con menores necesidades de recursos o para temas más especializados, en donde las pequeñas empresas tienen posibilidades de negocio. Desde predicciones personalizadas para una zona determinada, hasta pronósticos de vientos para parques de aerogeneradores. Por ejemplo, una pequeña empresa gallega llamada *4gotas* proporciona información sobre la predicción del tiempo, pero incluye entre sus servicios la posibilidad de hacerla para el día y lugar de un evento determinado, como una fiesta o un concierto al aire libre.

Aunque el pronóstico del tiempo es quizá la aplicación más popular de la computación, existen otros ámbitos en donde la predicción juega un papel importante, como en la medicina o las finanzas. En la primera, su uso se extiende cada vez más desde la extracción de información de los sistemas de imágenes clínicas hasta el análisis de los abundantes datos médicos. Un ejemplo en donde la predicción es fundamental es el tratamiento del cáncer con radioterapia.

El cáncer es una enfermedad que afecta a millones de personas cada año en Europa y con un crecimiento en su incidencia debido al envejecimiento de la población. Sin embargo, el éxito de las terapias hace que cada vez exista un mayor éxito de cura. Entre ellas está la radioterapia, empleada en más del 50% de los casos. Esta técnica consiste en la deposición de radiación en el tumor con un valor superior al que pueden soportar las células cancerígenas, produciendo su muerte. El médico prescribe los valores que se han de suministrar en el tumor. Pero alrededor del mismo puede haber órganos vitales que es necesario proteger, por lo que también marca las cantidades máximas que estos pueden recibir para no dañarlos y, por tanto, generar otros problemas. El tratamiento con radioterapia se puede suministrar de varias formas. Por ejemplo, insertando pequeñas fuentes radioactivas alrededor del tumor (técnica conocida como braquiterapia) o a través de fuentes externas de rayos-X muy intensos o de electrones (denominada teleterapia o radioterapia externa). En cualquiera de los dos casos, es necesario planificar el tratamiento, prediciendo la cantidad de radiación que se suministrará en cada zona, que tendrá que estar dentro de los límites marcados clínicamente. Esta planificación es completamente personalizada y solo sirve para un determinado paciente.

En el caso de la radioterapia externa, para poder suministrar los elevados niveles de radiación en el tumor, es necesario enviarla desde diversos ángulos de entrada para no dañar el tejido del paciente desde la piel hasta

el tumor. La suma de la radiación recibida desde cada ángulo en el tumor ha de ser superior o igual a la marcada por el oncólogo, mientras que la depositada en otros tejidos u órganos ha de ser inferior a los límites marcados. Para predecir cuales son los niveles de radiación recibidos en cada caso, se utilizan modelos físicos que se resuelven numéricamente en estaciones de trabajo del hospital (denominados *Treatment Planning System* o TPS — Sistema de Planificación de Tratamiento). En los tipos de tratamiento más sofisticados, además es posible aplicar técnicas de optimización que encuentren el mejor tratamiento posible.

Para calcular estos tratamientos personalizados, se parte de una Tomografía Computerizada del paciente. Este tipo de imagen hospitalaria proporciona la densidad de sus tejidos en millones de puntos diferentes de su cuerpo. Cada punto informa sobre el valor para un pequeño volumen cúbico llamado *voxel*. En los modelos más sencillos que se utilizan habitualmente en el cálculo de los tratamientos, dado que el cuerpo humano es fundamentalmente agua, se utilizan la deposición en esta de la radiación como valor inicial, corrigiéndose por los valores de la densidad en ese punto. El resultado es satisfactorio para la mayor parte de los casos. En algunos casos más complejos, como cuando hay cerca zonas muy densas (como los huesos) o con aire (como los pulmones), los resultados no siempre son suficientemente precisos para garantizar una correcta planificación. Gracias a las tecnologías de la HPC es posible utilizar ahora otros algoritmos más complejos, pero también más demandantes computacionalmente como son los métodos de Monte Carlo.

Los métodos de Monte Carlo deben su nombre al famoso casino de esa ciudad. Es una técnica de muestreo estadístico propuesta por el matemático polaco Stanislaw Ulam en 1946 y posteriormente desarrollada conjuntamente con John von Neumann, mientras trabajaban en el proyecto Manhatan para crear bombas atómicas. La idea fundamental es utilizar números aleatorios para decidir si ocurre o no un hecho determinado, como una colisión entre dos electrones o si saldrá cara o cruz al lanzar una moneda al aire. Para obtener información útil, es necesario repetir el mismo cálculo miles o millones de veces. En el caso de la teleterapia con rayos-X, se generan millones de fotones (partículas que forman los rayos de luz según las teorías cuánticas). Cada fotón se sigue a través del cuerpo virtual del paciente generado a través de la información suministrada por la Tomografía Computerizada, decidiendo cada poco tiempo si ha colisionado o no con un electrón. Si ambas partículas chocan, se transfiere energía del fotón al electrón, que también puede chocar con otros electrones, perdiendo parte de la energía adquirida inicialmente en cada una

de ellas. Al final, toda la energía perdida por el fotón queda depositada en el interior del cuerpo del paciente. Dado que para obtener el valor total de la dosis es necesario repetir estas simulaciones por Monte Carlo millones de veces, es necesario contar con una modesta infraestructura de computación de altas prestaciones para obtener los resultados que se puedan utilizar clínicamente en un tiempo razonable (inferior a un día de trabajo).

Aunque estos métodos se utilizaron inicialmente para resolver problemas físicos, se han extendido a otras áreas. Una de ellas es el entorno financiero, en donde se emplea para calcular los riesgos de los créditos o de las carteras de inversión o incluso predecir el valor futuro de las acciones en el mercado. Para la medida del riesgo de las carteras de inversión o portfolios, se utiliza una medida denominada *Value-at-Risk* (o VAR). Este indicador calcula la peor pérdida esperada en un futuro cercano definido dentro de un nivel de confianza establecido, suponiendo condiciones normales de funcionamiento del mercado y asumiendo que la composición de la cartera no varía en ese tiempo. Es decir, supongamos que tenemos una conjunto de acciones cuyo valor hoy es de 1000€. Queremos saber cuál es la máxima pérdida que podemos tener dentro de un mes para decidir si podemos soportarla o no. Para ello calculamos el VAR o solicitamos que se calcule, y se obtiene un valor de 80€ con una confianza del 99%. Esto quiere decir que tenemos una probabilidad del 1% de tener pérdidas superiores a 80€ en el periodo de un mes. Para calcular este valor estadístico se utilizan fundamentalmente dos métodos: la simulación histórica y el cálculo basado en Monte Carlo. La primera está basada sobre la información acumulada durante meses o años sobre cada uno de los valores que componen la cartera, estimando a partir de estos el valor futuro. Según los estudios de la consultora McKinsey & Company, se utiliza en el 75% de los bancos. Otro 15% lo calcula utilizando el método de Monte Carlo, más lento pero con algunas ventajas añadidas. El 10% restante combina ambos. Según los datos de la misma consultora, el tiempo de cómputo necesario para la cartera de un banco oscila entre 2 y 15 horas. El largo tiempo de computación es debido a la necesidad de estimar el valor futuro de cada uno de los valores que componen la cartera.

Matemáticamente, el VAR se define como la probabilidad de tener unas pérdidas superiores a una cantidad determinada. Para calcular esa probabilidad es necesario restar a la estimación del valor futuro el que tiene actualmente para cada uno de los activos de la cartera. Esta predicción se computa utilizando el modelo propuesto por los economistas Fischer Black y Myron S. Scholes, que considera que el valor futuro es estocástico

o aleatorio, cambiando de forma similar al movimiento Browniano de una pequeña partícula sobre un líquido, que oscila de un lado al otro sin poder decir en cada momento hacia donde hará el siguiente movimiento. Dada la aleatoriedad de este proceso, los métodos de Monte Carlo se ajustan particularmente bien, pero a costa de calcular muchas veces la evolución desde el momento presente al tiempo futuro donde se quiere hacer la estimación. Para cada simulación, el camino seguido y el valor predicho serán diferentes a los otros, pero permite calcular un valor promedio junto con su error. Independientemente del método utilizado, simulación histórica o Monte Carlo, el cálculo del VAR requiere de una infraestructura HPC para obtener con suficiente antelación el resultado (el VAR a un día se ha de obtener en pocas horas, y desde luego, antes de que se empiece a operar en el mercado financiero).

Requisitos de tiempo de respuesta similares aparecen en el último de los casos seleccionados de predicción y en donde la computación de altas prestaciones presta uno de sus mejores servicios a la competitividad empresarial: la ingeniería. Hoy en día es impensable diseñar un avión, un coche o un tren sin la ayuda de la simulación numérica. Como ya se comentó en el primer capítulo, su inclusión en el proceso de diseño ha permitido reducir en varios años los ciclos de desarrollo de nuevos vehículos a la vez que se mejoraba el confort, se reducía el consumo energético y se aumentaba la seguridad. Todo en uno, gracias a la posibilidad de trabajar con modelos virtuales del producto que se está diseñando y predecir su comportamiento en las condiciones de trabajo reales. Además, al igual que en la planificación de tratamientos de cáncer, es posible utilizar técnicas de optimización para obtener el mejor producto que cumpla unas condiciones preestablecidas, tanto de requerimientos de funcionamiento (por ejemplo, el diseño óptimo de una viga al que se le pide que soporte un peso máximo determinado) como de negocio (que necesite la menor cantidad de material y, por tanto tenga un coste menor, pero manteniendo sus prestaciones). No solo es factible predecir su funcionamiento. También se puede simular el proceso de fabricación en sí, detectando si el producto tendrá defectos o se dañará como consecuencia del mismo, incluso antes de construir los útiles necesarios para hacerlo.

En este tipo de simulaciones, el flujo de trabajo comienza con el dibujo de la pieza o producto utilizando un programa de diseño asistido por computador (CAD), habitualmente en tres dimensiones. El modelo generado pasa a una segunda fase denominada preprocesado en donde se construye una malla como en los casos anteriores (en la Ilustración 4 se puede ver la malla con tetraedros, muy común en este tipo de simulaciones), se

selecciona el tipo de análisis que se quiere realizar (mecánico, termodinámico, electromagnético, etc.), se escoge el tipo de material que forma la pieza (y por tanto sus propiedades físicas) y finalmente las condiciones que definen el caso que se quiere estudiar. El tercer paso es computacionalmente intensivo y consiste en la resolución numérica de las ecuaciones matemáticas. Dependiendo del tipo y tamaño del problema definido, la obtención del resultado puede llevar unos minutos en una estación de trabajo o días en un computador paralelo como el descrito en el capítulo anterior. Una vez obtenida la solución, los resultados se analizan para conocer el comportamiento de la pieza. Si el análisis indica que la pieza no cumple las expectativas o requisitos que se necesitan, se reinicia el ciclo cambiando el diseño o los materiales, hasta que se obtenga un producto conforme a lo exigido. Aunque el proceso aquí descrito es una versión simplificada, sirve para mostrar la contribución de la computación a la mejora de la calidad de los productos y el gran ahorro de costes que significa. Durante su diseño no es necesario construir prototipos físicos y además se pueden realizar multitud de experimentos virtualmente, que de otra forma serían muy costosos (pensemos por ejemplo en una prueba de colisión de un nuevo vehículo, que lo daña o destruye cuando se realiza).

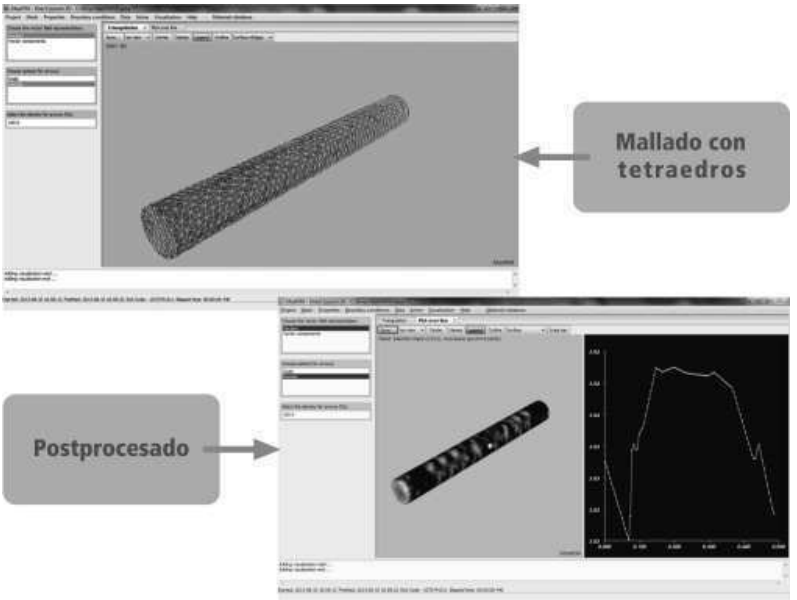


Ilustración 4: Ejemplo de simulación para ingeniería. Campo eléctrico en un cilindro.

Diagnosticar

En el campo de la ingeniería, el diagnóstico consiste en la búsqueda de la causa de los síntomas observados, generalmente de un funcionamiento incorrecto o de un fallo en su comportamiento. Entre otras técnicas, se puede utilizar la denominada *Model-Driven Diagnosis* (es decir, diagnóstico guiado por modelos) en donde se resuelven estos para el sistema que se pretende diagnosticar.

El diagnóstico está continuamente presente en nuestras vidas, sobre todo en el campo de la m istan antecedentes familiares previos y existen síntomas coincidentes. Para obtener estas respuestas, es necesario secuenciar aquellas partes de nuestro ADN que están asociadas al gen o conjunto de genes que están relacionados con la enfermedad, para posteriormente analizarlos en busca de las alteraciones que la producen.

Nuestro ADN está formado por una larga cadena de cuatro aminoácidos (adenina, timina, citosina y guanina) también llamadas bases. El conjunto de nuestros cromosomas tienen unos 3000 millones de bases. Con ellos se codifican las proteínas, que también están formadas por el encadenamiento de otros 23 tipos de aminoácidos. Así, la combinación citosina-guanina-adenina (o CGA) indica que en esa posición de la proteína ha de estar el aminoácido argenina. Por tanto, conocer la ordenación de las secuencias de aminoácidos en el ADN permite identificar si existen cambios en las proteínas. La búsqueda e identificación de esas alteraciones en la codificación es lo que permite el diagnóstico. En la actualidad, el proceso de diagnóstico se ha acelerado gracias a la introducción de las nuevas generaciones de tecnologías de secuenciación (también conocidas como NGS) que permiten obtener más rápidamente la codificación de fragmentos de ADN de cadenas cortas de 200 a 500 bases de las zonas de interés y en paralelo, pero a costa de incluir algunos errores.

Estos procesos de secuenciación producen grandes cantidades de información que es necesario filtrar y ordenar para obtener la cadena de aminoácidos de interés para el diagnóstico. La computación de altas prestaciones contribuye en este proceso de reconstrucción en varias de las fases necesarias. Una de ellas incluye el alineamiento, es decir, buscar el lugar en donde se ha de colocar el fragmento que se ha secuenciado con respecto a la de referencia, que será uno de los cromosomas humanos. Es como colocar una pieza de un puzle que tiene millones de ellas. Este procedimiento informático puede necesitar horas o días para encontrar la localización de cada uno de los miles de fragmentos que se han generado, además de consumir importantes recursos de memoria RAM. Alre-

dedor de estas capacidades aparecen empresas especializadas tanto en la realización del proceso de secuenciación como en el análisis posterior. Por ejemplo, la empresa coruñesa HealthInCode se ha especializado en el diagnóstico de enfermedades a través del estudio genético, principalmente de cardiopatías familiares. Para ello, requiere secuenciar y analizar de 9 a más de 70 genes, dependiendo de la enfermedad que se busque.

Aunque la explotación clínica de la genética avanza rápidamente y adquiere cada día una mayor presencia, hay otras técnicas médicas que también se apoyan en la computación. Una de ellas es el diagnóstico asistido por computador (o CAD, *Computer Aided Diagnosis*). El objetivo es dar apoyo al radiólogo en la toma de decisiones, facilitándole información adicional sobre la imagen que está analizando. Por ejemplo, en un análisis de un cáncer de mama, la detección de las zonas en donde está presente. Aunque el tratamiento computacional de la imagen no está libre de errores, se ha demostrado que con estas herramientas el radiólogo mejora sustancialmente su diagnóstico, no solo en la detección de las zonas microcalcificadas, sino también en su clasificación como benignas o malignas.

El proceso computacional se divide en varias fases que parten de la imagen digital del paciente. Sobre esta se detectan los límites de los órganos y se aplican filtros para incrementar su calidad a efectos de la detección de las lesiones. El proceso puede terminar ahí, suministrando solo información sobre las zonas candidatas, o proseguir para clasificarlas como benignas o malignas, aportar características de las lesiones, etc. Las necesidades computacionales no son tan altas como en el diagnóstico genético. De hecho, muchos sistemas comerciales se venden en una estación de trabajo. Pero los algoritmos involucrados sí son computacionalmente intensos.

Descubrir

Esta rápida visión del panorama de utilizaciones de la computación de altas prestaciones termina con dos casos de relevancia económica y social importante: la búsqueda de nuevos recursos energéticos y de nuevos medicamentos.

La búsqueda de hidrocarburos utiliza la computación de altas prestaciones para localizar posibles bolsas de petróleo que después se han de confirmar haciendo prospecciones físicas. La aplicación de estas técnicas ha producido una reducción muy significativa en los costes de la búsqueda. Así, desde los 15 dólares por barril que eran necesarios en 1979, bajó a esta 6 dólares en 1992. Más de un 50% de reducción.

La principal herramienta que se usa es la exploración geofísica utilizando ondas sonoras. La detección de candidatos se inicia con una toma de

datos sísmicos en el área de interés. Por ejemplo, en las zonas costeras un barco genera impulsos sonoros y recoge con una cadena de micrófonos los ecos devueltos desde el fondo marino. Estos pulsos pueden penetrar en el subsuelo, siendo reflejados y refractados de forma diferente en función del material que lo componga. Dado que el petróleo es un material ligero que asciende a través de los materiales más porosos, es necesario que existan otros que formen barreras en donde se acumule. El objetivo es conseguir un mapa tridimensional del subsuelo a partir de los datos recogidos, para que posteriormente sea analizado por los geólogos. Estos son los que deciden cuales son los posibles candidatos para la perforación. Por tanto, es necesario resolver un problema inverso. Si en el caso de un vehículo, conocemos su forma y los materiales que lo componen y lo que queremos es saber la respuesta que tendrá en las condiciones de trabajo normal (o anormales, por ejemplo, en caso de colisión), en el caso de la exploración geofísica se conoce la respuesta (el eco de los sonidos generados) pero se desconoce el detalle de la forma y material del subsuelo que se quiere explorar. El proceso completo está compuesto de muchas fases diferentes (filtrado de las señales, cálculo de velocidades de propagación, etc.) y es un proceso computacionalmente muy costoso que necesita una potencia de cálculo muy importante. Además, también es intensiva en el uso de información. El tamaño de los conjuntos de datos iniciales que se utilizan puede ser de varios Terabytes (incluso decenas). Finalmente, también incluye la visualización de los resultados para que los geofísicos e ingenieros puedan hacer la interpretación de los resultados. Pero el ratio entre el beneficio obtenido y el coste también es muy elevado.

La utilización de estas técnicas inversas para obtener mapas del terreno no está restringida solo a zonas costeras sino que también se pueden hacer en tierra y con otros objetivos, como la localización de minerales, el control de aguas subterráneas o incluso la planificación de minas. Tampoco se restringe exclusivamente a la localización, sino que se han extendido a otras áreas de la explotación petrolífera como la delimitación del volumen de la reserva, el control de la explotación, la optimización de los puntos de extracción, etc.

Descubrir nuevas reservas de hidrocarburos es importante económicamente, pero encontrar nuevos medicamentos que combatan las enfermedades también lo es socialmente. El proceso para producir un nuevo medicamento es muy largo y extremadamente costoso, por lo que cualquier reducción del tiempo de desarrollo representa un importante ahorro. Típicamente un nuevo fármaco necesita cerca de 15 años desde que se inicia la investigación hasta que se puede comercializar, con un coste entre

800 y 1000 millones de dólares. Además, solo el 1 de cada 5000 compuestos que se prueban llegan a ser comercializados. Como consecuencia de su alto coste, se han ido introduciendo técnicas computacionales que permitan descubrir nuevos candidatos para combatir las enfermedades reduciendo costes e índices de fracasos. Este conjunto de herramientas se denomina *Computer-Aided Drug Design* (CADD).

Una de las primeras herramientas que se utiliza es el *screening virtual*. Anteriormente a la incorporación de estas técnicas computacionales, las compañías farmacéuticas utilizaban robots que probaban experimentalmente los compuestos químicos (entre cien y quinientos mil), lo que representaba varios meses de trabajo. Con la inclusión de las técnicas computacionales, el tiempo para descubrir posibles candidatos se ha reducido enormemente y ampliado su capacidad.

Las especies químicas activas de un fármaco suelen ser pequeñas moléculas que desactivan la función de una proteína específica, a través del cambio de su estructura. El objetivo del *virtual screening* es encontrar posibles candidatos que puedan ser posteriormente analizados y mejorados como medicamento. Para ello, primeramente se modeliza y se analiza la proteína que se quiere desactivar, partiendo de la información cristalográfica o incluso de las secuencias de ADN que la codifican (como se ha visto anteriormente) a través de homologías con otras conocidas. En este paso de modelización se utilizan algoritmos computacionales como los de dinámica molecular. Sobre este modelo, se identifican una zona en donde se puede ligar pequeñas moléculas y que desactiven su actividad biológica. El siguiente paso es encontrar candidatos que puedan encajar químicamente en esa zona, que puedan ser analizados posteriormente y, en algunos casos, mejorados. Esta búsqueda se hace probando computacionalmente si los compuestos existentes en las bases de datos pueden o no encajar en la zona de interés, teniendo en cuenta su forma y composición, así como cierto grado de flexibilidad en la proteína. Frente a los sistemas anteriores robotizados, estos algoritmos de búsqueda de fármacos pueden probar virtualmente muchos más compuestos en el mismo tiempo y a un coste inferior.

En este capítulo se han descrito brevemente diversas aplicaciones de la computación de altas prestaciones dentro del amplísimo panorama y en campos como la ingeniería, la medicina, la energía, las finanzas, etc. En algunos de los casos expuestos se necesitan muy grandes infraestructuras, mientras que en otros solo se emplea una estación de trabajo. Hay otros usos también muy asentados, como en la industria cultural, en donde la

creación de efectos especiales o de películas de animación está basada en el uso intensivo de la computación. En estos casos, la parte artística es la predominante, no siendo imprescindible que las simulaciones encajen con la realidad, como es deseable en la computación científico-técnica. Sin embargo, es un área interesante de negocio en donde contamos en España con una empresa líder (Next Limit Technologies) que ha merecido el premio *Technical Achievement Award* de la Academia de Hollywood en 2008 por su tecnología. Sus simulaciones realistas de líquidos (chocolates, lava, agua) han contribuido a grandes producciones, pero también les ha permitido desarrollar un software que ahora puede ser utilizado en el campo científico y técnico.

En cualquier caso, el acceso a la infraestructura de computación para las pequeñas y medianas empresas puede ser un factor limitante debido a las necesidades de inversión si quieren tenerla en propiedad. Además, algunos usuarios pueden pensar que su uso es demasiado complejo. Por consiguiente, es necesario encontrar soluciones que permitan ocultar la complejidad de la infraestructura y reducir la inversión necesaria (o, mejor, transformarla en un coste variable) en aras de eliminar esas barreras. La prestación de servicios en la nube puede abrir el camino para hacerlo.

«The PC age is giving way to a new era: the utility age»[11]

Nicholas Carr.

«The big switch: rewiring the world, from Edison to Google»

CAPÍTULO 4.
LA NUBE PARA HPC

Estamos tan acostumbrados que no somos conscientes de lo que supone realmente. Cuando accionamos un interruptor, esperamos que se encienda la luz de la sala. Si abrimos un grifo, queremos que salga agua depurada que podamos beber. Al descolgar un teléfono (tanto fijo como ahora con los móviles), exigimos que haya conexión y podamos realizar una llamada (incluso si no hay electricidad en ese momento). Pero llegar a esta situación no ha sido ni fácil ni ha ocurrido rápidamente. Incluso aún hoy, en algunas zonas de nuestro continente, estas infraestructuras básicas no existen, a pesar de que la telefonía se considera un servicio universal. El conjunto de estos servicios públicos esenciales se denomina en el lenguaje de la economía como *utilities*. Además de esta persistencia en el servicio, una de las características comunes es que son servicios de pago por su utilización, frecuentemente con una cuota mensual de mantenimiento más otra parte en función de lo que consumimos. ¿Llegará la computación a ser también un servicio público esencial? ¿Será este siglo XXI la era del servicio público de computación? Probablemente sí, aunque antes seguro que nuestros legisladores piensan primero en Internet como servicio universal al que todos tendremos derecho.

De hecho, Nicholas Carr plantea en su libro «*The big switch: rewiring the world, from Edison to Google*»[12] un paralelismo entre la evolución de la

11. «La era del PC está dando paso a una nueva: la era del servicio público»

12. Existe una versión en castellano titulada «El gran interruptor: el mundo en red, de Edison a Google» editada por Deusto S.A. Ediciones.

red eléctrica y el mundo de la computación. Durante su consolidación como servicio público, las empresas manufactureras pasaron de generar su propia energía a comprarla a los distribuidores, que concentraban cada vez más su producción en grandes centrales eléctricas. Aunque la co-generación, los pequeños huertos solares o las minicentrales hidráulicas son opciones también utilizadas actualmente, el modelo eléctrico sigue basado en grandes infraestructuras de generación.

En el campo de las tecnologías de la información también se está produciendo este cambio gradualmente. En esta era de Internet, esperamos que los servicios que tenemos estén siempre disponibles, como el correo electrónico, la mensajería instantánea, las redes sociales o el omnipresente buscador de Google. Para poder prestar estos servicios de forma eficiente y eficaz, las grandes compañías crean grandes centros de datos que ubican en zonas con acceso fácil a redes de comunicaciones, con costes energéticos bajos y, a ser posible, también salarios reducidos. A fin de cuentas, como Nicholas Carr plantea en su libro «*Lo que la fibra óptica de Internet hace por la computación es exactamente lo que la red de corriente alterna hizo para la electricidad: consigue que la localización del equipamiento sea indiferente para el usuario*»[13].

El paradigma actual dominante en la evolución hacia este servicio público de computación está centrado en las tecnologías de la nube (o más comúnmente *Cloud computing*). La computación de altas prestaciones no es ajena a este movimiento, incluso puede ser realmente la solución para que el acceso y uso por pequeñas y medianas empresas se generalice. Aunque el camino no está exento de obstáculos.

¿Qué es la computación en la nube?

Durante los últimos años se ha intentado encontrar una definición que permita encuadrar el concepto de computación en la nube. Existen dos definiciones que compiten entre sí, aunque parece ser que la estadounidense se va imponiendo. La primera de ellas es utilizada por el Grupo de Expertos de Cloud de la Comisión Europea, que considera la nube como:

«*un entorno de ejecución elástico de recursos que involucra a muchas partes interesadas y proporciona un servicio medido a múltiples granularidades para un nivel determinado de calidad (de servicio)*»[14].

13. Traducción del autor de «What the fiber-optic Internet does for computing is exactly what the alternating —current network did for electricity: it makes the location of the equipment unimportant for the user».

14. Traducción del autor de «*a 'cloud' is an elastic execution environment of resources involving multiple stakeholders and providing a metered service at multiple granularities for a specified level of quality (of service).*»

La segunda es debida al NIST (*National Institute of Standards and Technology*) americano. Esta organización propone que la nube es:

«*Un modelo que permite el acceso bajo demanda, ajustado y omnipresente a través de la red de comunicaciones a un conjunto compartido de recursos de computación configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que pueden ser rápidamente asignados y liberados con un esfuerzo mínimo de gestión o de interacción con el proveedor del servicio*»[15].

La definición propuesta por la organización norteamericana se complementa con cinco características esenciales. Estas acercan ambas propuestas, ya que el NIST considera que entre ellas que también es un servicio medido, típicamente siguiendo un modelo de pago por uso. Las otras cuatro son:

- Autoservicio bajo demanda. Es decir, el consumidor final de los recursos puede asignárselos directamente y automáticamente sin la necesidad de la intervención de otra persona, principalmente empleados del proveedor del servicio. Habitualmente, esta provisión se realiza a través de una aplicación web u otra aplicación ligera proporcionada por el proveedor del servicio, aunque para algunos de las capacidades de la nube (como la creación de infraestructuras virtuales que se describirá más adelante) existen estándares que permiten la automatización del proceso como el *Open Cloud Computing Interface* (OCCI).
- Amplio acceso desde la red de comunicaciones. Los recursos están accesibles a través de la red de comunicaciones y de cualquier tipo de dispositivo que esté conectado a ella como los PC, móviles o *tablets*. Este punto es importante ya que implica la necesidad de estar conectado a la red para tener acceso al servicio, aunque en algunos casos se han desarrollado soluciones que permiten tener parte de la funcionalidad cuando no hay conexión.
- Recursos compartidos. Los recursos informáticos proporcionados por el proveedor al consumidor final son compartidos entre varios clientes simultáneamente. Cada uno de ellos puede demandar una configuración física o virtual diferente que se asigna o libera dinámicamente según sus peticiones. Habitualmente donde se despliega o ejecuta su petición no es importante para el usuario, aunque puede restringirla (por ejemplo, debido a la normativa de protección de datos personales, podría solicitar que el servicio no se prestara fuera de España o Europa para evitar problemas legales).

15. Traducción del autor de «*a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*»

- Elasticidad rápida. Por elasticidad se entiende la capacidad de crecimiento y decrecimiento en la asignación de los recursos. Dentro de esta característica se distingue entre elasticidad horizontal (los recursos se incrementan o disminuyen en número, por ejemplo, pasando de un servidor a dos o viceversa) o vertical (las capacidades de uno de los recursos se cambian dinámicamente, como podría ser añadir más memoria RAM o un mayor número de unidades de procesamiento a un servidor ya en funcionamiento). Esta característica de la nube es quizá una de las más primordiales y novedosas que aporta la tecnología empleada, ya que permite ajustar las necesidades y los recursos demandados en cada momento.
- Medición del servicio. Los servicios en la nube controlan y optimizan el uso de los recursos aprovechando la capacidad de medición del mismo en función del tipo de recurso empleado (por ejemplo, horas de uso de la CPU asignada, o espacio de disco consumido en el caso del almacenamiento, o ancho de banda utilizado cuando se accede a la red de comunicaciones). Aunque el NIST no lo contempla directamente dentro de esta característica esencial, esta capacidad es la que también permite realizar el pago por uso, siguiendo el modelo de un servicio público.

Ambas definiciones concuerdan en lo básico: elasticidad, medición y compartición de recursos computacionales entre varios interesados. La definición del NIST, añade además el autoservicio, que acerca la nube a otros servicios públicos si se proporcionan de forma general. En el Cloud se proponen otros modelos de despliegue.

Modelos de despliegue de la nube

El modelo de la nube reflejado en las definiciones anteriores no presupone que se creará una infraestructura pública de acceso general, sino que está abierta a varias formas de despliegue, como ocurre también en otros servicios públicos esenciales. Se han identificado cuatro modelos de construcción de un servicio en la nube. El primero es el **privado**, es decir, aquél que es instalado por una organización (generalmente grande) para su propio uso interno, pero teniendo las características de autoservicio, flexibilidad y gestión expuestas en la sección anterior. La instalación la puede hacer directamente o puede subarrendarla o desplegarla en la infraestructura de un proveedor externo, pero no está compartida con otras organizaciones. Esta compartición es identificada en el segundo modelo denominado **comunitario**, en donde los recursos se aprovisionan para el uso exclusivo de varias organizaciones o particulares que tienen un interés común. Por

ejemplo, en el paradigma de computación Grid (quizá un precursor del Cloud) se maneja el concepto de Organización Virtual, en donde varias organizaciones (generalmente grupos de investigación de un área científica que colaboran intensamente entre ellos) ponen en común sus computadores e instalan en ellos las aplicaciones de uso más frecuente.

El tercer caso es el **público**, en donde el servicio está abierto al uso general por cualquier organización o particular. Habitualmente estará gestionado por alguna empresa que busca un beneficio por ello, a través del pago por su uso u otra compensación indirecta, como puede ser la inclusión de publicidad. Aunque también es posible que la organización que acerque el servicio busque un fin social, como puede ser una aplicación web gubernamental de autodiagnóstico energético con el que reducir el consumo eléctrico. En cualquier caso, este modelo es el necesario para acercar la computación de altas prestaciones a las pequeñas y medianas empresas, ya que las libera de las obligadas inversiones en caso de no utilizarlo.

Existe una cuarta posibilidad, el sistema **híbrido**, en donde los dos primeros modelos se apoyan en el entorno público, extendiendo sus capacidades sobre este en caso de necesidad de más recursos. La ventaja es que las nubes privadas o comunitarias pueden tener una infraestructura diseñada para el uso más habitual, creciendo solo en momentos puntuales de carga más intensa, reduciendo de esa manera las inversiones necesarias y los costes operativos asociados. NIST hace una categorización más generalista, basada en la combinación de cualquiera de ellos, manteniendo su independencia pero añadiendo la capacidad de interactuar entre ellos a través de protocolos estándar o propietarios, pudiendo compartir datos, infraestructura o aplicaciones. Por tanto, se acerca más a un modelo **federado**, en donde los diferentes proveedores Cloud proporcionan el servicio de forma autónoma, pero también pueden colaborar con otros, por ejemplo permitiendo sistemas de identificación externos, compartiendo una entrada común para autoservicio o combinando sus capacidades.

Modelos de servicio en Cloud

Hasta aquí se ha considerado la nube o Cloud como una forma de provisión de servicios o, mejor, como un servicio para acceder a recursos computacionales. Por tanto, el objetivo es acceder a una infraestructura computacional para resolver nuestra necesidad. En función de cuales sean nuestros intereses, así será el servicio que demandemos. Es el concepto de cualquier-cosa-como-servicio (*everything-as-a-Service* o XaaS) que ha estado latente en Internet en los últimos años. Sin embargo, en el entorno

de Cloud, se ha ido constriñéndo hasta reducirlo a tres categorías que se apoyan una sobre otra.

La primera de ellas es el modelo de **Infraestructura como Servicio** (en inglés *Infrastructure-as-a-Service* o IaaS). El usuario final o consumidor puede demandar dinámicamente acceso a recursos informáticos básicos, como las CPU, almacenamiento de datos, acceso a redes de comunicaciones, o una combinación de ellas. Por ejemplo, a un proveedor IaaS se le podría solicitar el aprovisionamiento de un servidor con tres CPUs, 10GB de memoria RAM y 200GB de almacenamiento que tuviera un interface de red Ethernet de 10Gbs. Con esta descripción del recurso no existe mucha diferencia con respecto a los ya conocidos proveedores de *hosting* o *housing* que existen desde hace más de dos décadas. La innovación real del Cloud (y quizá la más sobresaliente y que distingue este paradigma) es que dicha petición se puede hacer dinámicamente, en un modelo de autoservicio, escalable y elástico, sin necesidad de la intervención del personal técnico del proveedor. Esta innovadora revolución fue lanzada con éxito comercial por la empresa Amazon, cuando en 2006 abrió sus servicios (todavía dominantes y de referencia en el mercado) de almacenamiento de datos (*Simple Storage Solution* o S3) y computación (*Elastic Compute Cloud* o EC2). Aunque existían precedentes, como la plataforma Sun Cloud, que ofrecía el acceso a servidores tipo Sparc de la empresa Sun (ahora integrada en Oracle) en un modelo de pago por uso, la simplificación del servicio a través de la combinación de las tecnologías de virtualización, de aplicaciones web y de identificación y autorización seguras para crear un computador bajo demanda fue una verdadera innovación.

La siguiente capa es la **Plataforma como Servicio** (*Plaform-as-a-Service* o PaaS). Este nivel de abstracción de la nube está pensado para desarrollar o desplegar aplicaciones específicas que utilizan las herramientas o librerías (API) proporcionadas por el proveedor del servicio. Por ejemplo, Google proporciona una serie de funciones que permiten desarrollar aplicaciones que accedan a sus servicios de mensajería o calendario para crear nuevas soluciones que añadan otras capacidades. En el caso de un entorno Cloud, el programador de la aplicación no debería preocuparse de la gestión de infraestructura, siendo responsabilidad de la API proporcionada por el proveedor, o de las herramientas internas del proveedor, adaptarla elásticamente a la demanda. Por tanto, debería ocultar la infraestructura y su gestión, tanto al desarrollador como al usuario final cuando la aplicación esté en producción, ajustándose dinámicamente para garantizar la calidad del servicio.

Finalmente, la propuesta de más alto nivel es el **Software como Servicio** (*Software-as-a-Service* o SaaS). Todos conocemos aplicaciones que se proporcionan a través de Internet como la mensajería electrónica, calendario, procesadores de texto o incluso de gestión empresarial. Aunque el término elegido se ha adaptado al nuevo paradigma de cualquier cosa como servicio, realmente el concepto no es nuevo y hereda las propuestas de los años 90 del siglo pasado conocidas como ASP o *Application Service Provider* (Proveedor de Servicios de Aplicación). Es, con mucho, la propuesta más usada por el consumidor final, ya sea gratuita o de pago.

Cloud para HPC

Las soluciones en la nube, gracias a su flexibilidad, permiten primero reducir, incluso eliminar, los costes de inversión (o de capital) necesarios para arrancar un proyecto al no necesitar instalar una infraestructura dedicada. Estos se convierten en costes operacionales para el caso de empresas u otras entidades que no ofrezcan los servicios de infraestructura como parte de su línea de negocio. De hecho, existen diversos actores que se pueden beneficiar del Cloud para el desarrollo de su actividad. Entre estos están:

- Usuarios o consumidores finales. Pueden utilizar los servicios en la nube para satisfacer sus necesidades de aplicaciones o de infraestructura. En el primero de los casos, utilizarán los servicios de tipo SaaS. Existen múltiples ejemplos de aplicaciones en el campo de la ciencia y la técnica que están disponibles a través de interfaces WEB y que, para muchos casos sencillos que están limitados en tiempo de ejecución, son muy apreciados. Por ejemplo, el *European Bioinformatics Institute* (Instituto Europeo de Bioinformática) proporciona y mantiene un amplio conjunto de aplicaciones relacionadas con la biología, desde acceso a bases de datos moleculares de interés biológico hasta herramientas de alineamiento de secuencias como las comentadas en el tercer capítulo de este libro. El acceso lo proporciona a través de un interface para navegador de Internet o, para aquellos usuarios más avanzados, a través de servicios WEB que pueden ser invocados desde programas externos.

  Además del acceso a la aplicación en sí, los consumidores finales del Cloud pueden crear dinámicamente infraestructuras virtuales o no (depende del proveedor del servicio), similares a las descritas en el capítulo segundo, adaptadas a sus necesidades puntuales, sin necesidad de tener infraestructura propia. Existen múltiples ejemplos de proveedores que proporcionan actualmente este servicio, como Amazon,

Sabalcore, Nimbix, Oxalya o Gompute. Este modelo se adapta muy bien a aquellas empresas, grupos de investigación u otras instituciones que requieren el uso intensivo de computación en momentos puntuales y acotados. Por ejemplo, en ingeniería es muy común que en un proyecto sea necesario hacer simulaciones del producto o proceso. De la duración total del proyecto, solo una fracción es dedicada a la simulación. Por tanto, puede ser rentable económicamente desplegar un clúster adaptado en el Cloud frente a tener infraestructura propia. Aparte de cuestiones de política interna, existe un punto de inflexión a partir del cual es más rentable utilizar uno u otro modelo que ha de ser convenientemente analizado.

- Revendedores con valor añadido. Gracias a las posibilidades ofertadas por los grandes proveedores de infraestructura y de aplicaciones en Cloud, los más técnicos pueden cubrir necesidades que aporten valor añadido a un mercado vertical que, en muchos casos, no es rentable para el proveedor pero sí para pequeñas organizaciones con conocimiento especializado (pero no desarrolladores de su propio software). Por ejemplo, en el modelo de Amazon que proporciona sus servicios a través de máquinas virtuales es posible configurar estas para que incluyan software específico, actualizado, verificado y validado en el entorno, liberando al consumidor final de esa tediosa labor. La empresa puede incluir esta máquina específica dentro del catálogo disponible en los servicios de Amazon, facturando un sobrecoste al consumidor final por su utilización. Otras empresas, como Cycle Computing, se han especializado en la creación de grandes infraestructuras de cálculo sobre proveedores IaaS.
- Vendedores Independientes de Software. Para pequeñas empresas que desarrollen su propio software que demande computación de altas prestaciones existen varias posibilidades. Pueden proporcionar el servicio tanto en SaaS como en PaaS, si quieren abrir su plataforma al desarrollo de nuevas funcionalidades por terceros, lo cual es muy común en las aplicaciones técnicas. La ventaja es evidente para estas pequeñas compañías ya que permite ofertar su solución mundialmente con una infraestructura reducida si utilizan adecuadamente la nube para contener sus costes.

Por tanto, Cloud para HPC se refiere principalmente a dos de los modelos de provisión: IaaS y SaaS. Al ser modos de uso diferentes y con requerimientos en principio también diferenciados, es imprescindible verlos separadamente.

IaaS para HPC

De la presentación de las tecnologías de la computación de altas prestaciones realizada en el segundo capítulo, es posible inferir que cualquier entorno en la nube necesita de algunas características esenciales. En concreto, como mínimo es imprescindible poder integrar en la infraestructura demandada varios servidores (o uno solo pero con varios núcleos) y almacenamiento masivo eficiente. Si se requiere la utilización de muchos núcleos y estos no pueden ser del mismo servidor, también será necesario contar con una red de comunicaciones de gran ancho de banda y baja latencia. Con esta infraestructura mínima se podrían ejecutar muchos de los problemas que demandan las pequeñas y medianas empresas así como la mayor parte de los grupos de investigación universitarios. Aunque la oferta de Cloud está más centrada en dimensionar recursos para poder ofrecer servicios WEB, las soluciones propuestas para HPC están claramente diferenciadas entre aquellas que utilizan la virtualización de servidores como núcleo del servicio frente a otras más especializadas que basan su diferencia comercial en la configuración de clústeres sin virtualización.

La virtualización de los servidores tiene para el proveedor de infraestructura múltiples ventajas, incluyendo la posibilidad de compartir recursos entre diferentes clientes. Sin embargo, desde el punto de vista del consumidor final de sus soluciones, esta compartición es perjudicial. Entre otros efectos negativos están la baja eficiencia de los interfaces de entrada/salida, la variabilidad en los tiempos de ejecución y los problemas de seguridad.

Es innegable que la virtualización de servidores se ha generalizado como método de consolidación de servicios o de creación de entornos personalizados aislados, con su propio sistema operativo y configuración. Básicamente es una capa de software que emula el comportamiento del hardware, directa o indirectamente. El sistema operativo y las aplicaciones que se ejecutan sobre este software no tienen constancia de si se están ejecutando en hardware real o virtualizado. Al ser una capa de software adicional, estas y las funcionalidades del sistema operativo tienen una penalización, habitualmente pequeña, pero que no es despreciable. Esta es no muy grande en el uso de la CPU (los resultados actuales varían entre el 1 y el 5% de degradación, que podría ser asumible), pero todavía la tecnología no ha avanzado suficiente en la gestión de las entradas y salidas de datos desde y hacia la máquina virtual, tanto en el acceso al almacenamiento como las comunicaciones de red. Ambos recursos son utilizados intensamente por las aplicaciones HPC. Si se utilizan aplicaciones paralelas que requieran más de un servidor, estas verán degradar su eficiencia debido a los problemas en la latencia o en el ancho de banda disponible

(más en el primer caso que en el segundo, por lo que las aplicaciones que tengan que comunicarse frecuentemente tendrán un mayor impacto en la degradación de su rendimiento).

Una segunda consecuencia de la utilización de la virtualización y de la compartición de los recursos es la variabilidad en los tiempos de ejecución de las aplicaciones. Para una misma aplicación y tamaño del problema, se ha observado que los tiempos de ejecución pueden variar apreciablemente, lo cual hace poco predecible cuando se puede obtener el resultado. Existen varias causas que pueden explicar esta variabilidad, incluso para la ejecución en un único servidor: la compartición del nodo y comunicaciones con otros usuarios del entorno Cloud, la sobresubscripción de los recursos hardware, la asignación de diferentes tipos de CPU, la interferencia con los procesos del sistema (y de gestión de las máquinas virtuales), etc. Dado que en los entornos de Cloud habitualmente se paga por el tiempo de uso o reserva de la CPU, la variabilidad implica también diferencias en los costes de las ejecuciones. Probablemente en el futuro, la forma de pago por uso se pueda normalizar de otra forma más transparente y verificable, como ocurre en el caso de la luz eléctrica que se ha tomado como ejemplo.

Finalmente, en algunos casos, principalmente empresariales, existe la preocupación por la seguridad de la información que se utiliza. Pensemos en una empresa que está desarrollando un nuevo producto del que quiere simular su comportamiento bajo condiciones realistas. Subir el diseño al entorno de la nube crea en muchas empresas un sentimiento de desasosiego debido a que piensan que está menos seguro que en sus propias dependencias. Aunque en la mayor parte de los casos es todo lo contrario, existe riesgo de que sea cierto. Solo las medidas técnicas adoptadas por el proveedor pueden garantizar que la información enviada a la nube tenga las medidas de seguridad adecuadas para evitar esa fuga de información delicada. Tanto la selección del proveedor correcto como de la configuración de la infraestructura remota virtual son vitales para reducir este riesgo.

En cualquier caso, la virtualización es una de las posibles soluciones para la prestación del servicio de HPC en la nube, pero no es la única factible. De hecho, los proveedores de IaaS especializados en computación de altas prestaciones suelen utilizar como herramienta de marketing el que no utilizan virtualización, ya que son conocedores de las desventajas que suponen, además de incorporar tecnologías más específicas de HPC como la red de comunicaciones entre nodos (dominada actualmente por la solución de Infiniband).

Sea un modelo u otro, existen otras barreras a tener en cuenta. La primera es que el tamaño sí importa, que el saber ocupa lugar. Las soluciones que utilizan este tipo de computación suelen ser además intensivas en el uso de datos, tanto de entrada como sobre todo de salida. Es muy frecuente tener ficheros de entrada de Gigabytes o incluso superiores y de salida del mismo tamaño o incluso mayores. Esto implica, un tiempo que no es desdeñable en la transmisión de estos ficheros entre la sede de la empresa a la del proveedor del servicio y viceversa, además de ser un coste adicional en caso de que se cobre por el envío o recepción de la información (coste que es necesario incluir cuando se analiza la rentabilidad de utilizar la computación en la nube). Incluso el tamaño de los ficheros puede hacer inviable en tiempo la transmisión de la información utilizando las redes de comunicaciones habituales. Las soluciones propuestas para esta dificultad técnica son variadas. Desde el envío de los ficheros utilizando discos duros u otro dispositivo de almacenamiento por correo postal u otro tipo de transporte comercial (lo cual, como se puede comprobar, no encajaría con la característica esencial propuesta por NIST de entorno de autoservicio la intervención de los técnicos de la empresa proveedora) hasta la realización completa de todo el proceso necesario en la nube, minimizando el trasiego de información inicial y basando la solución técnica en la visualización remota. Esta permitirá el análisis de los resultados obtenidos y la reducción de la información a descargar a un mínimo. En cualquier caso, un entorno productivo de visualización remota puede demandar un ancho de banda de bajada de 4 megabits/segundo por usuario y, a ser posible, latencias también bajas para tener una sensación real de trabajo local.

En algunos sectores más asentados como la ingeniería existe el problema de la gestión de las licencias del software. Las soluciones utilizadas son mayoritariamente comerciales, que suelen tener sistemas de protección y limitación de uso basadas en algún tipo de hardware (como llaves que han de estar conectadas al computador) o de servidores de licencias. Existen propuestas técnicas para solventar esta barrera, tanto ofrecidas por el vendedores del software (por ejemplo, CD-Adapco ofrece su solución Power-on-demand que permite obtener créditos adicionales para el software ya licenciado, que puede ejecutarse en cualquier servidor en la nube que tenga acceso directo a Internet o la solución ElasticLM de gestión de licencias, pensada específicamente para estos entornos dinámicos) como por los proveedores del Cloud, que tienen acuerdos con los primeros para vender la utilización de su software como un valor añadido. En cualquier caso, es un elemento que ha de ser analizado antes de decidir la utilización de la nube como entorno de trabajo, ya que puede resultar una barrera técnica o financiera infranqueable.

Aunque existen dificultades técnicas y operativas para utilizar el Cloud en computación de altas prestaciones, estas son conocidas por los proveedores. Estos están innovando constantemente para ofrecer la mejor calidad de servicio a través de entornos gráficos que simplifican la creación de las infraestructuras computacionales o la utilización de las aplicaciones. Las experiencias existentes indican que la utilización del modelo IaaS es factible para HPC, que cada día las barreras existentes son menores, pero que siempre es necesario hacer un estudio del retorno de la inversión, comparando el uso exclusivo de recursos en la nube con la utilización de servidores locales (teniendo en cuenta todos los costes, como el de personal necesario para la instalación y mantenimiento, las amortizaciones, el consumo de electricidad —nada desdeñable en este momento—, la climatización o la utilización de espacio físico).

SaaS para HPC

La mayor parte de los usuarios, sino todos, de la computación de altas prestaciones están más interesados en los resultados que en la manera llegar a ellos. De hecho, frecuentemente la forma de acceso a la infraestructura computacional, sea en Cloud o no, es una barrera que no están dispuestos a franquear. Por ejemplo, para utilizar muchos centros de computación es necesario aprender a utilizar sistemas de colas, entornos de acceso basados en terminales, etc. Para este tipo de usuarios, la mejor opción es ocultar la complejidad aparente de la infraestructura y centrar el servicio en la aplicación, siguiendo un modelo de Software-as-a-Service.

Las ventajas para ambas partes (consumidor de la solución y fabricante del software) son múltiples. Para el primero, además de poder concentrarse exclusivamente en su problema computacional, se reduce el tiempo necesario para utilizar el software, que está siempre disponible a través de Internet, no es necesario instalarlo ni verificarlo en local, está siempre actualizado a la última versión, con la corrección de errores correspondiente, no necesita comprar hardware adicional ni mantenerlo, etc. Casi lo único que tiene que hacer para poder aprovecharse eficientemente y eficazmente del servicio es validar inicialmente que el software sirve para sus propósitos y formarse en el uso del mismo y en el interface proporcionado por el proveedor, tareas ambas que también tendría que hacer si se instalase localmente. Por otro lado, desde el punto de vista económico, también podría tener una reducción de costes si la facturación se hace siguiendo el modelo de pago por uso, aunque de nuevo esto dependerá de los costes de licencia y mantenimiento fijados por el proveedor del software.

Para el fabricante del software, sobre todo para los más modestos, utilizar estos mecanismos de acceso a su solución le permite ofertarla global-

mente, mantener constantemente actualizado su software, no depender de soluciones de terceros para la gestión de las licencias, obtener información de las funcionalidades más utilizadas por sus clientes (pero no debería acceder o analizar los datos de entrada o salida utilizados para generarles confianza y que no vean el servicio como un riesgo, tal y como se comentó anteriormente. En este entorno técnico no sería admisible el análisis de los datos de entrada o salida por el proveedor), etc. A pesar de las ventajas evidentes, muchos de los grandes vendedores de software técnico se muestran cautelosos para ofertar sus soluciones en este modelo. Una razón para ello es que se produce un cambio en la generación de ingresos. Frente a un modelo de cuotas de mantenimiento anual que les garantiza unos entrada de dinero más o menos constante y predecible, un entorno puro en la nube en modo de pago por uso implica una probable variabilidad. Por ello, la oferta comercial en estos casos suele ser dual: licencias para su uso en las instalaciones locales con posibilidad de ejecución en entornos remotos en un modelo de uso bajo demanda. En pocos casos se ofrece un servicio puramente SaaS directamente por el vendedor del software, pero sí a través de revendedores o de otros proveedores de servicios.

Por el contrario, para pequeños proveedores de software así como para empresas de reciente creación intensivas en conocimiento, la utilización de este entorno les permite entrar rápidamente en el mercado a unos costes muy razonables. Al igual que otros servicios de Internet pensados para el público, la combinación de un interface web amigable con una infraestructura de apoyo flexible, como la que puede proporcionar un proveedor IaaS, le permite crecer rápidamente, adaptarse a la demanda, y controlar los costes de infraestructura sin necesidad de cuantiosas inversiones iniciales (evitando el riesgo de hacer una sobreestimación o subestimación de los recursos necesarios).

Ejemplos de servicios HPC en Cloud

Históricamente, los centros de supercomputación ofrecen acceso a sus infraestructuras fundamentalmente para actividades de investigación y desarrollo. Aunque su posicionamiento es anterior al concepto de Cloud y, en el sector académico, la obtención de asignaciones de tiempo es frecuentemente moderada por convocatorias públicas periódicas de proyectos, la forma de ofrecer sus servicios cumple con la mayor parte de los requisitos esenciales propuestos por el NIST: autoservicio, ya que los usuarios demandan la ejecución de sus trabajos a través de sistemas de colas como los descritos en el capítulo 2, que pueden estar priorizados y con limitaciones de uso (por ejemplo, el número de aplicaciones —trabajos— en

ejecución simultáneamente por un único usuario o el número de núcleos o el tamaño de memoria RAM que puede solicitar para cada uno de ellos); amplio acceso desde la red, habitualmente a través de entornos de terminal pero con soporte limitado todavía a otros tipos de interfaces; recursos compartidos como es el conjunto de supercomputadores que gestionan o el almacenamiento masivo que suelen incluir; y servicio medido, que se realiza en la mayor parte de los centros, sean de acceso a través de convocatorias de proyectos para controlar que la asignación máxima de recursos no se traspase, o bien por motivos de facturación por uso. Quizá la escalabilidad rápida de los recursos es el punto más controvertido, ya que el usuario puede demandar diferentes recursos computacionales en cada petición de trabajo dentro de los límites de asignación concedidos en caso de convocatorias públicas de proyectos, pero no puede incrementarlos fácilmente durante la ejecución de la aplicación. La mayor parte de estos centros de supercomputación dependen de las administraciones públicas estatales o regionales, habiendo sido creadas para satisfacer la demanda de colectivos específicos en el área de la investigación académica o institucional (fundamentalmente militar o energética). Sin embargo, durante los últimos años, debido al convencimiento de las administraciones de que la computación es imprescindible en el ámbito industrial y, sobre todo, a la apreciación de la dificultad para acceder a estas tecnologías por parte de las pequeñas y medianas empresas, se han iniciado diversos programas para acercar la computación de altas prestaciones a estos sectores, como el programa INCITE en Estados Unidos o el desarrollado por la infraestructura europea de supercomputación PRACE. Estos casos siguen el mismo modelo que el académico, con convocatorias periódicas de proyectos orientadas específicamente al sector privado y con grandes necesidades de computación. Hay otras experiencias más cercanas a las necesidades cotidianas de las PYME, como es el caso de CloudPYME.

CloudPYME es una experiencia financiada por FEDER a través del programa de cooperación transfronteriza Galicia-Norte de Portugal (POCTEP), liderada por el Centro de Supercomputación de Galicia (CESGA) en colaboración con los centros tecnológicos AIMEN, CATIM y AINMAP, el primero en Galicia y los dos restantes portugueses. Su objetivo es acercar la simulación numérica a empresas manufactureras de esa euroregión, ofreciendo soluciones técnicas que reduzcan las barreras detectadas en estudios realizados previamente: carestía del software, por lo que se apostó por software libre que tuviera las mismas funcionalidades; falta de personal especializado, que se abordó a través de cursos de formación y soporte técnico especializado ofrecido por los centros tecnológicos, siempre en cercanía con las empresas; falta de infraestructura computacional

suficiente, que implicaba bloquear las estaciones de trabajo de la empresa mientras se está haciendo la simulación. La solución técnica adoptada ha estado enfocada al entorno Cloud, que incluye servicios computacionales, de almacenamiento de datos y de ayuda en la utilización de las aplicaciones. La idea principal del proyecto es ofrecer un servicio completo que abarque toda la cadena de valor necesaria para que la empresa adopte la simulación numérica dentro de sus procedimientos internos de desarrollo de un producto o proceso, desde la validación del software para sus necesidades hasta la infraestructura computacional.

La arquitectura computacional del proyecto incluye varias capas que pueden ser utilizadas libremente y bajo demanda por el usuario final. La primera es una versión adaptada de Linux que incluye el software libre seleccionado para la experiencia (Salomé para el desarrollo del CAD en 3 dimensiones y Code_Aster para la ejecución de las simulaciones) y utilidades específicas para conectarse a los servicios Cloud. Esta versión de Linux se puede instalar en los PC de la empresa, arrancar desde una llave USB directamente o utilizarse desde una máquina virtual. A elección del consumidor. La intención es que la empresa utilice esta versión de software para diseñar sus productos y hacer el pre y postproceso de las simulaciones. Pero tampoco es imprescindible su empleo, ya que también puede utilizarla remotamente en la nube, sin necesidad de una instalación local.

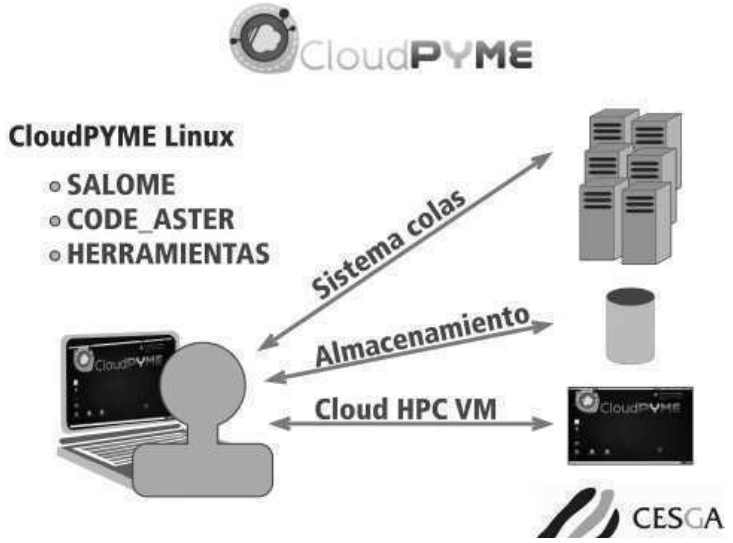


Ilustración 5: Infraestructura Cloud de CloudPYME

Los servicios Cloud ofrecidos (ver Ilustración 5) incluyen un espacio de almacenamiento remoto y seguro en donde guardar los modelos de sus productos y los resultados de sus simulaciones y análisis. Dicho sistema es compartido exclusivamente por los empleados de la empresa, pero no es visible por otras empresas o usuarios del CESGA. Con esta funcionalidad se busca darle a la empresa de que sus diseños estarán a salvo, reduciendo la desconfianza en los entornos Cloud que se comentó anteriormente. Desde la versión de Linux ofrecida por el proyecto se puede acceder a este entorno de almacenamiento directamente, tanto desde las instalaciones de la empresa como desde los servicios computacionales remotos. Estos están compuestos de dos formas de provisión. Por un lado, el usuario puede demandar la creación de una o varias máquinas virtuales (con el mismo software y sistema operativo) siguiendo el modelo IaaS descrito anteriormente. Dichas máquinas virtuales le ofrecen un entorno de escritorio remoto idéntico al que tendría localmente, con visualización interactiva. La petición la realiza a través de un entorno WEB que además le permite controlar el consumo que ha realizado de cada uno de los servicios ofrecidos. Adicionalmente, y para facilitar la ejecución de las simulaciones, también se puede solicitar con una utilidad específica incluida en la distribución de Linux. En este caso, la máquina virtual arrancada se añade automáticamente a la configuración del software Code_Aster para que pueda ser utilizada en las simulaciones de forma transparente. Finalmente, para aquellas ejecuciones más complejas, o porque el usuario así lo quiere, también tiene acceso a los computadores del CESGA, pero a través de los sistemas de colas, sin virtualización, que aportan una mejor eficiencia a costa de cierto tiempo de espera para comenzar la ejecución, al ser un entorno compartido con otros usuarios.

El diseño de esta arquitectura computacional ha buscado reducir las barreras técnicas y psicológicas vistas anteriormente: aporta visualización remota para evitar el trasiego de información innecesaria desde la nube; ofrece medidas de seguridad adicionales para el acceso a los archivos de la empresa que generen confianza; añade herramientas sencillas para facilitar el autoservicio; aporta elasticidad a través de la creación dinámica de máquinas virtuales, que además pueden utilizarse siguiendo un modelo IaaS híbrido en donde la infraestructura de la empresa puede complementarse con la ofrecida por el CESGA cuando es necesario.

A mediados de 2013 esta experiencia había sido utilizada por más de 15 empresas de la Euroregión, demostrando que las capacidades de HPC en Cloud son factibles, aunque no están todavía exentas de dificultades, como toda tecnología novedosa. Pero, aparte del comentado problema

del ancho de banda, la mayor parte no están relacionadas con el servicio Cloud sino con la convivencia de Linux (utilizado mayoritariamente en computación de altas prestaciones, como se ha comentado anteriormente) con Microsoft Windows (dominante en el colectivo de empresas objetivo) y del software elegido (que no es tan amigable como el comercial, aunque tenga capacidades iguales o superiores a este).

El modelo aportado por CloudPYME es una mezcla de los modelos IaaS y SaaS, ya que es posible crear una infraestructura virtual o también se posibilita la utilización de una aplicación específica. Esta es una evolución lógica de los servicios de un centro de supercomputación, que también han seguido la mayor parte de las compañías especializadas en servicios HPC en Cloud. Así, compañías como la sueca Gompute o la francesa Oxalya ofrecen servicios de computación basados en su propia infraestructura compartida o a través de clústeres de servidores dedicados siguiendo el popular modelo de hosting, pero para computación de altas prestaciones. Sobre el primero de ellos, tienen instaladas previamente aplicaciones científico-técnicas tanto de software libre como comerciales (como openFOAM para simulación de fluidos o ANSYS para problemas de multifísica), de las cuales tienen acuerdos con los fabricantes de la solución. Para facilitar a los clientes la ejecución de estas en los servicios compartidos, han desarrollado portales WEB de acceso que permiten la utilización del software en un modelo de SaaS, controlando también la ejecución de los trabajos individuales (envío a la plataforma, control del estado de la ejecución, etc.). Al ser conscientes de la necesidad de interacción y del problema del ancho de banda, el interface WEB incluye la visualización remota. Incluso en el caso de Gompute, han desarrollado soluciones ad-hoc para mejorar la respuesta a través de técnicas de reducción de la latencia (por supuesto, esta no puede ser nunca inferior a la marcada por la velocidad de la luz, aproximadamente 1 milisegundo cada 150 kilómetros de distancia).

Finalmente, también existen servicios IaaS completamente virtualizados para HPC bajo demanda, como los que ofrece Amazon en sus instancias de HPC, que están basados en hardware de última generación con varias CPU multinúcleo conectadas por redes locales Ethernet 10Gigabit. Este tipo de red de comunicaciones permite tener, en caso de utilizar varios nodos o instancias, unas prestaciones eficientes aunque todavía alejadas de las conseguidas en infraestructuras locales.

Un servicio puro de SaaS es el proyecto e-IMRT, que ha desarrollado una solución para la verificación de tratamientos de radioterapia externa utilizando técnicas de Monte Carlo. El proyecto ofrece un interface WEB en donde el radiofísico del hospital sube la información del tratamiento

que ha diseñado en sus infraestructuras locales (véase el tercer capítulo al respecto) en formatos estándares. La aplicación utiliza estos ficheros de entrada, más la configuración definida sobre el acelerador lineal de electrones que se utilizará en el hospital, para generar los mapas de dosis que recibirá el paciente. Esta simulación realista implica la ejecución de cientos o miles de trabajos de simulación, pero de los que el radiofísico no es consciente, ya que se le oculta tanto la infraestructura que está utilizando como el desglose detallado que se ha definido. La simulación completa del tratamiento puede llevar algunas horas, por lo que se le informa del estado de forma sencilla, marcando las fases (tiene cinco) en las que se encuentra la verificación. Al final el proceso, el físico del hospital puede visualizar los resultados y comparar los mismos con los que él ha introducido, buscando si la definición del tratamiento cumple con las expectativas. Es una plataforma software pensada para mejorar la calidad y garantizar la seguridad de la radioterapia externa. El usuario solo se concentra en analizar los resultados, apoyado por herramientas proporcionadas por el entorno, pero no necesita tener conocimientos de computación ni preocuparse de la infraestructura.

El proyecto ha utilizado a lo largo del desarrollo muchos tipos de infraestructuras, desde los servidores locales hasta entornos puramente Cloud. La utilización de servidores locales tiene la ventaja de la eficiencia, pero presenta en algunos casos interferencias entre peticiones de servicio simultáneas (del mismo o diferentes clientes), debido a la compartición y limitación de los recursos, lo que convierte en impredecibles los tiempos de respuesta. Por ese motivo, el proyecto ha experimentado con otras soluciones Cloud puras, en donde para cada petición se despliega un clúster virtual dedicado y dimensionado para el caso específico a simular. De esta forma, no existen interferencias entre peticiones, a costa de un tiempo inicial de arranque. También, dado que son procesos paralelos pero desacoplados, esta infraestructura inicial puede crecer o decrecer en número de nodos (o núcleos de CPU) dinámicamente, aprovechándose de la elasticidad de la nube para ajustarse a las expectativas del usuario final, es decir, para poder devolver la respuesta en el tiempo pactado o previsto previamente. Incluso, utilizando varios proveedores Cloud de IaaS, ha experimentado con configuraciones de clúster virtuales con tolerancia a la caída de uno de ellos, lo que permitiría garantizar el tiempo para devolver los resultados de la simulación, expandiendo dinámicamente la infraestructura computacional sobre el proveedor sobreviviente.

El conjunto de las experiencias presentadas más otras, como el HPCexperiment que se ha desarrollado a escala mundial con decenas de casos de

uso en el sector de la simulación numérica empresarial, demuestra que el Cloud es una alternativa viable para la computación de altas prestaciones dentro de ciertas limitaciones técnicas que todavía es necesario afrontar. Para aquellas soluciones que necesiten una moderada capacidad de computación a un coste razonable, la nube es una alternativa a valorar detalladamente y muy recomendable para empresas pequeñas que necesiten liberarse de los costes iniciales de inversión y mantenimiento, o que tengan solo necesidades puntuales que no justifiquen la instalación local. Para los innovadores que diseñen y programen sus aplicaciones con requerimientos de computación intensiva, la opción de SaaS es realmente interesante. Si además se apoyan sobre proveedores IaaS que les garanticen flexibilidad y robustez, reducen los riesgos del mal dimensionamiento de la infraestructura (en donde existen múltiples ejemplos de fracaso debido a este error), adaptándose dinámicamente a los vaivenes de la demanda del servicio. En cualquier caso, los servicios Cloud para HPC, sea en formato IaaS, PaaS o Saas están todavía en evolución y consolidación, aunque avanzando rápidamente. Pero esto también ocurre en cualquier campo de la técnica y la innovación, que aportan constantemente nuevas soluciones gracias a los esfuerzos en investigación, desarrollo e innovación.

«El único medio de que la industria evite la ruina y el estancamiento es convertir el laboratorio en la antesala de la fábrica»

Santiago Ramón y Cajal

«Reglas y consejos sobre investigación científica»

CAPÍTULO 5.
I+D+I Y COMPUTACIÓN DE ALTAS PRESTACIONES

En sus más de 100 años de historia, los premios Nobel han sido otorgados a españoles en ocho ocasiones, y solo dos de ellas a científicos, recayendo en la literatura los otros seis. Tanto Severo Ochoa como Santiago Ramón y Cajal hicieron grandes contribuciones a la medicina. Todos apostaron por su pasión, a pesar de las dificultades sociales y técnicas de sus épocas. Esta es la forma de progresar, ya que como decía don Santiago «*toda obra grande, en arte como en ciencia, es el resultado de una gran pasión puesta al servicio de una gran idea*». Este aserto se puede aplicar también a los emprendedores, que luchan encomiablemente por sacar adelante la novedosa idea en la que creen, abriéndose un hueco en la difícil economía del mercado global.

Emprendedores e investigadores tienen en la computación de altas prestaciones un lugar en donde la evolución científica y técnica es constante, está siempre en movimiento y surgen nuevas dificultades y necesidades que requieren de nuevas soluciones que permita avanzar, para poder dar respuesta a las expectativas de sus usuarios, o que acerquen las oportunidades a otros nuevos que aprecien sus ventajas. Los retos planteados necesitan de investigación para generar nuevo conocimiento, de desarrollo para demostrar que es aplicable y, sobre todo, de innovación para acercarla a la sociedad. A pesar de su ya larga historia, la HPC está en ebullición, con imperiosas demandas

de novedades que permitan continuar con el gran avance conseguido durante los últimos decenios.

Como veremos más adelante, existen todavía muchos campos en donde es posible aportar. Tantos, que describirlos todos está fuera de las posibilidades y extensión de este pequeño libro. Lo importante es encontrar el nicho en donde nuestros conocimientos y capacidades técnicas y económicas lo permitan. Existen muchos retos tecnológicos a resolver que requieren grandes inversiones. Pero hay otros muchos que es posible abordar solo con ahínco, dedicación y pasión si tenemos el conocimiento adecuado. Incluso, como ocurre en gran parte de las innovaciones, a veces es sencillo y radical cuando se juntan los conocimientos de dos áreas diferentes. Si estamos empezando en un área concreta, como nos recomienda Cajal, lo más sensato es inicialmente apostar por problemas considerados menores, mejor que los grandes. Esto nos aportará el conocimiento para ir avanzando en esa área. Si lo que queremos es innovar, a menos que nuestra especialidad ya sea la computación de altas prestaciones, lo más interesante es mezclar esta tecnología con el conocimiento que tengamos para crear algo nuevo que nos permita competir en la economía globalizada. Es decir, crear nuevos usos, nuevos servicios. Y si ya tenemos una empresa o trabajamos en una entidad innovadora, lo importante es escuchar a nuestros clientes sobre sus problemas e inquietudes, que nos pueden aportar ideas o huecos en donde se necesitan innovaciones. Aunque mucha de la investigación está basada en las mejoras continuas de un producto o proceso, algunas veces se producen cambios radicales que pueden ser muy exitosos si se producen en el momento adecuado. Por ejemplo, la creación del World Wide WEB (WWW) por Tim Berners-Lee en el CERN a principios de los años 90 del siglo XX ha sido una creación que ha cambiado radicalmente nuestra vida en los últimos 20 años. Su gran aportación fue mezclar dos técnicas ya existentes pero hasta ese momento desconectadas: el hipertexto con las tecnologías de Internet.

Definiciones: Investigación, Desarrollo e Innovación

Coloquialmente se habla exclusivamente de investigación cuando nos referimos a cualquier tipo de actividad que busca encontrar o generar novedades. Sin embargo, seamos empresa, emprendedor o incluso profesional de la «investigación», es fundamental que sepamos en donde estamos. La OCDE y Eurostat han trabajado durante años para clarificar los conceptos, generando dos guías denominadas Manual de Frascati (referido a investigación y desarrollo) y Manual de Oslo (que aborda lo

que es innovación o no lo es, sobre todo para su aplicación a los estudios estadísticos en ese ámbito). Las definiciones se han ido puliendo, hasta alcanzar rango de reglamento en la Unión Europea.Las utilizadas actualmente son:

- **Investigación fundamental** se refiere a «*los trabajos experimentales o teóricos emprendidos con el objetivo primordial de adquirir nuevos conocimientos acerca de los fundamentos subyacentes de los fenómenos y hechos observables*». En algunos casos se le añade la coletilla «*sin perspectivas de aplicación práctica o uso*».
- **Investigación industrial** se considera aquella «*investigación planificada o estudios críticos cuyo objeto es la adquisición de nuevos conocimientos y técnicas que puedan resultar de utilidad para la creación de nuevos productos, procesos o servicios, o contribuir a mejorar considerablemente los existentes. Incluye la creación de componentes de sistemas complejos que sean necesarios para investigación industrial, especialmente la validación de tecnología genérica, salvo los prototipos*». Su diferencia principal con la anterior es que la búsqueda de nuevos conocimientos tiene un propósito claro.
- **Desarrollo experimental**. Se conceptualiza como «*la adquisición, combinación, configuración y empleo de conocimientos y técnicas ya existentes, de índole científica, tecnológica, empresarial o de otro tipo, con vistas a la elaboración de planes y estructuras o diseños de productos, procesos o servicios nuevos, modificados o mejorados. Entre las actividades podrá figurar la elaboración de proyectos, diseños, planes y demás tipos de documentación siempre y cuando no vaya destinada a usos comerciales*».
- **Innovación**. La más cercana al mercado, ya que se refiere a «*la introducción de un producto (bien o servicio) o de un proceso, nuevos o significativamente mejorados, de un nuevo método de comercialización, o de un nuevo método organizativo en las prácticas internas de la empresa, la organización del lugar de trabajo o las relaciones exteriores de dicha empresa*». En algunos casos, se recalca la necesidad de que la mejora sea sustancial. Por ejemplo, se utiliza frecuentemente en las convocatorias públicas de proyectos la siguiente definición: «*actividad que tiene como resultado la obtención y puesta en el mercado de nuevos productos, servicios o procesos, o mejoras sustanciales de los ya existentes, de modo que presenten características o aplicaciones que difieran sustancialmente de los existentes con anterioridad*». Mientras en investigación es fácilmente distinguible si se ha generado o no conocimiento nuevo, aseverar si las caracterís-

ticas de un producto o proceso difieren sustancialmente no siempre es sencillo.

En el mundo occidental existe actualmente una gran presión para que la investigación tenga una rápida aplicación en la sociedad. De hecho, en el área de la salud se habla de investigación traslacional, que busca acelerar el traspaso de nuevos conocimientos al área clínica y por tanto a la salud de los ciudadanos. En cualquier caso, en esta cadena de valor de la innovación, la secuencia empieza con la investigación cuyos resultados pueden ser aplicados a corto plazo en unos pocos casos. Para ello se realizan desarrollos experimentales que, en caso de ser exitosos técnicamente pueden ser introducidos en el mercado. Cuanto más cerca de la investigación básica, más riesgo de tener resultados negativos. Por el contrario, cuanto más próximo a la comercialización, más seguro se está de que técnicamente funciona, pero las inversiones y los riesgos de fracaso económico son también mayores. Frecuentemente se da el caso de que un prototipo experimental funciona correctamente, pero el análisis de mercado indica que no será exitoso, por lo que no termina el proceso. Cuanto más alejados estemos del resultado final, también es necesario más tiempo, por lo que la selección de la línea de trabajo ha de tener en cuenta este hecho. Como comentaba un representante de una gran empresa de software en una reunión sobre las necesidades de investigación y desarrollo en tecnologías Cloud: «*No interesa que investiguéis en lo que se necesita dentro de seis o doce meses. Nosotros tenemos la capacidad de hacerlo en ese tiempo o menos. Lo realmente importante es investigar sobre lo que viene detrás del Cloud dentro de tres o cinco años*». Claro que si lo supiéramos, y tuviéramos una mágica bola de cristal que adivinara el futuro fehacientemente, probablemente nadie lo diría e intentaría explotarlo directamente, ¿no?

Entonces, seamos profesionales de la investigación o emprendedores, necesitamos encontrar el área en donde podemos contribuir teniendo en cuenta nuestra experiencia, conocimientos y el plazo temporal que nos marcamos para ello. La selección puede ser sencilla, ya que existe una abundancia de ámbitos en donde la computación de altas prestaciones y sus aplicaciones necesitan de contribuciones. Aquí solo describiremos brevemente algunas de ellas, divididas en tres categorías: infraestructura, aplicaciones y nuevos usos.

I+D+I en infraestructuras de HPC

Desde el año 1965, cuando el co-fundador de Intel Gordon E. Moore predijo que el número de componentes en un circuito integrado se doblaría cada dos años (conocida como Ley de Moore), nos estamos constante-

mente beneficiando en todos los entornos de la informática. Pero aunque se aumente el número de componentes, se ha llegado a un límite en la frecuencia a la que trabajan los procesadores, debido al consumo energético excesivo. Como consecuencia, la solución ha sido utilizar la misma superficie para incluir más núcleos (más unidades de procesamiento) y añadir más funcionalidades, ya que el tiempo necesario para hacer una única operación ya no se puede seguir reduciendo. Como consecuencia indirecta, los grandes centros de computación que quieren alcanzar el máximo de operaciones matemáticas por segundo necesitan juntar millones de núcleos en un solo computador. Se estima que una máquina Exaflop (1 000 000 000 000 000 000 operaciones por segundo) tendrá entre 50 y 200 mil nodos, con varias decenas de millones de núcleos de cálculo (probablemente en entornos híbridos que incluyan CPU multinúcleo y co-procesadores) y consumirá una enorme cantidad de energía, cercana a los 20MW. La previsión es que esté funcionando la primera alrededor del año 2020. Tomando estas estimaciones como realistas, los retos tecnológicos son enormes a nivel de infraestructura y de desarrollos básicos en hardware.

Claramente, el primero es que se necesitan nuevos procesadores que puedan realizar más operaciones por segundo y que consuman mucho menos que los actuales. Damos por supuesto que este tipo de investigación y desarrollo seguirá siendo dominado por los grandes fabricantes de las CPU, en donde cada día juegan un papel más relevante aquellos que han estado en el mercado de los procesadores de muy bajo consumo que se utilizan, por ejemplo, en teléfonos móviles. La necesidad de obtener grandes prestaciones con muy bajo consumo hace que ambos mundos estén convergiendo y hay muchos proyectos en el área de hardware dedicados a la HPC que están ensayando estas posibilidades, como el proyecto europeo Mont-Blanc o el nuevo servidor Moonshot de HP. Europa, debido a su liderazgo durante años en el diseño de la CPU para entornos embebidos, puede jugar un papel relevante en los próximos años si aprovecha esta oportunidad.

También en el área más cercana al hardware, otro importante hueco para la investigación es la mejora en el ancho de banda y la latencia relacionados con todos los sistemas de entrada y salida de información, tanto a la memoria principal o RAM como a otros dispositivos como los interfaces de red o el almacenamiento. En el primero de los casos para poder alimentar eficientemente la CPU con los datos que necesita y, por consiguiente, poder mantener de forma sostenida los cálculos. El segundo es necesario para reducir al máximo los tiempos perdidos en las comunicaciones entre los procesos de una aplicación paralela distribuida. Finalmente, la entrada y

salida al almacenamiento necesita mejorarse dada la cantidad de información que se analizará o se generará en grandes computadores. Las soluciones que se encontrarán, imprescindibles para estos enormes sistemas, también incrementarán sustancialmente la eficiencia de los computadores más pequeños, como ha ocurrido hasta ahora. Aunque probablemente la mayor parte de estos avances serán realizados por grandes compañías, por lo menos a nivel de explotación comercial.

El problema de consumo energético no solo se ha de atacar desde el cambio en los componentes de los computadores. El calor generado por estos ha de controlarse adecuadamente en todo tipo de instalaciones, desde las más pequeñas a los grandes centros de computación. En la actualidad, un armario de computadores puede consumir alrededor de 30kW. En el pasado, cuando los consumos por cada uno de estos era mucho menor, el aire era suficiente para poder ventilar y extraer el calor producido. Ahora es necesario buscar otras soluciones, con una tendencia a utilizar nuevamente el agua como elemento refrigerante (por su mayor capacidad calorífica, casi 3.500 veces superior a la del aire para el mismo volumen) . Incluso se han desarrollado sistemas que sumergen los ordenadores en baños de aceite. Si sigue la tendencia de incrementar la densidad por armario, como es previsible, serán necesarios nuevos sistemas de refrigeración robustos y fiables.

Un segundo problema es qué hacer con el calor, la energía, que se ha extraído. Tratarla implica también consumo eléctrico adicional, y por lo tanto, incremento de los costes operacionales y medioambientales. ¿Se puede reutilizar esa energía de forma eficiente? La respuesta es sí. Por ejemplo, existe ya algún caso en donde se emplea para calentar el agua de una piscina. Es un campo en donde se está innovando constantemente por la necesidad de llegar a una situación de equilibrio. Un área todavía poco explorada es la posibilidad de convertir directamente el calor en electricidad para realimentar el computador.

La siguiente capa que necesita innovaciones es el sistema operativo que controla el acceso al hardware del computador. Teniendo cada vez más elementos de cómputo (tanto más núcleos en las CPU como más co-procesadores), la gestión de los procesos e hilos ha de ser mucho más eficiente, ha de minimizar las interrupciones de las aplicaciones de cálculo y los tiempos de espera en los accesos a los datos que están en la memoria RAM. Estos mejoran si están en la memoria más cercana a la CPU que los demandará (lo que se denomina localidad). El sistema operativo ha de gestionar adecuadamente la asignación de las unidades de procesamien-

to a los procesos para incrementar la localidad todo lo posible de forma transparente a la aplicación.

Es posible que la gestión se delegue en los balanceadores de carga que tengan una visión global del estado de la computadora cuando está compuesta de más de un nodo. Los retos son los mismos, incluyendo también la posibilidad de que se incluya en las tomas de decisiones el acceso a los datos de entrada necesarios así como la cercanía temporal, en términos de latencias, para las aplicaciones paralelas distribuidas.

Para administrar correctamente un entorno computacional son necesarias herramientas de monitorización y contabilidad eficientes y eficaces. Estas son necesarias tanto para los grandes computadores como para la gestión de los Centros de Proceso de Datos tradicionales o los de nueva construcción, e incluso más decisivos para los proveedores de servicios Cloud. Aquí se abren nuevos retos y también nuevas oportunidades para las pequeñas y medianas compañías que puedan producir soluciones escalables. Por un lado, se necesitan nuevos paradigmas para controlar una infraestructura que puede tener varios millones de componentes, en los que el tiempo medio de fallo (es decir, que alguno de los nodos que componen el sistema global tendrá algún error hardware) será de horas o menos. La abundancia de información a recibir hace casi imposible que un humano pueda procesarla adecuadamente, por lo que serán necesarios mecanismos inteligentes intermedios que puedan tomar decisiones de forma autónoma. Por otro lado, es necesario incluir en esta monitorización otros parámetros que no son habituales, referidos al consumo energético a todos los niveles (desde el nodo hasta casi cada operación realizara por la CPU) o la eficiencia de las aplicaciones (por ejemplo, controlando e informando sobre los tiempos de acceso a la memoria RAM, de escritura en disco o de parámetros de rendimiento de la propia aplicación). Recientemente los sistemas de monitorización y gestión también se han extendido a la infraestructura al completo, incluyendo los sistemas de refrigeración, sensores, etc. con el objetivo de reducir el consumo energético global y garantizar que el sistema en conjunto sea robusto (denominados DCIM o *Data Center Infrastructure Management*). Es un área joven en donde todavía existe un amplio abanico de posibilidades en la innovación, tanto desde la recogida de la información como en el análisis de la misma. Por último, no se ha de dejar a un lado la seguridad y confianza, desde la detección de usuarios no autorizados hasta la identificación de usos indebidos.

En los nuevos paradigmas de computación como el Cloud para HPC son necesarias mejoras sustanciales en la virtualización, de tal forma que las

pérdidas de rendimiento existentes actualmente (por ejemplo, en los sistemas de entrada y salida de datos) se minimicen, además de permitir una mayor flexibilidad en las configuraciones de cada entorno virtual (por ejemplo, en la escalabilidad vertical permitiendo una mejor gestión de los incrementos y decrementos de los recursos asignados, sea número de CPU o cantidad de memoria RAM). Pero también existe el modelo que permite evitar la virtualización a través de dividir físicamente una computadora de forma dinámica (técnicamente en inglés se denomina *bare-metal*, aunque la mayor parte de los proveedores que lo comercializan en la actualidad no lo realiza automáticamente). Para ello es necesario que se pueda actuar sobre los nodos y la red o redes de interconexión para aislar el nuevo clúster generado de las influencias de otros, así como instalar y configurar adecuadamente el sistema operativo con las personalizaciones del cliente. Es decir, lo que ahora se está haciendo en entornos Cloud virtuales será necesario reproducirlo de manera eficaz y eficiente, sin perder las capacidades de elasticidad que caracterizan al Cloud en formato IaaS que son muy beneficiosas.

Finalmente, la visualización remota es un campo en donde se necesitan nuevas soluciones, debido a varios factores: será casi imposible mover eficientemente y a un coste razonable los datos generados por las aplicaciones en los grandes supercomputadores; por la tendencia de las grandes empresas a consolidar los entornos informáticos de desarrollo de productos, de tal forma que todos los ingenieros y diseñadores solo acceden a aplicaciones remotas; a la mayor presencia del Cloud y la existencia de múltiples dispositivos de visualización, desde estaciones de trabajo, portátiles, *tablets* o teléfonos móviles. En los tres primeros dispositivos, para que el usuario o trabajador sea productivo y se sienta confortable, es imprescindible que el comportamiento de la aplicación parezca que es el mismo, independientemente de si la está ejecutando en su estación de trabajo o en los servidores remotos. Dado que es imposible reducir la latencia por debajo de un valor determinado, que viene dado por la distancia física, y a las limitaciones de ancho de banda que puedan existir, son necesarias técnicas que oculten o aminoren ambos problemas. La diversidad de dispositivos de visualización requiere una adaptación de las aplicaciones a las características finales, tanto en capacidades gráficas como en interfaces hombre-máquina (del ratón y teclado de las estaciones de trabajo al dedo que utilizamos en las *tablets* o *smartphones*).

El objetivo de todas estas mejoras es prestar un servicio que se adapte adecuadamente a las expectativas de los usuarios, cada vez más exigentes. La evolución futura de la computación de altas prestaciones, y su cada

vez mayor implantación en los procesos productivos de las empresas, hace que existan unas características clave que han de inspirar las actividades de investigación, desarrollo e innovación:

- Flexibilidad, a la inclusión de componentes adicionales (frente a las configuraciones monolíticas existentes mayoritariamente hace unos años, en donde todos los servidores que componían una computadora eran iguales, se ha pasado a configuraciones híbridas, a las que incluso se les añaden a lo largo de su vida útil nuevos servidores, con tipos de CPU y configuraciones diferentes), a cambios en su forma de uso con el tiempo y a la coexistencia de aplicaciones con diferentes requerimientos.
- La adaptación al fallo (o en inglés, *Resilience*). Las computadoras han de ser más maleables, recuperándose rápidamente en caso de fallo y afectando mínimamente al servicio (por ejemplo, limitando el impacto sobre las aplicaciones que se estuvieran ejecutando en el momento del fallo, que idealmente tendría que ser nulo).
- Fiabilidad (*Reliability*), es decir, reducir al máximo la posibilidad de fallo.
- Disponibilidad (*Availability*), definida como el porcentaje de tiempo que la computadora está realmente en operación con respecto al tiempo total que tendría que estar operativa. La disponibilidad ha de ser la máxima posible, idealmente cercana a la que se alcanza en las redes de comunicaciones actualmente.
- Facilidad de mantenimiento (Serviceability). La instalación, configuración, operación, solución de problemas, etc. ha de ser sencilla y ágil.
- Eficiencia, tanto en el consumo energético global (de la computadora y de la infraestructura adicional necesaria para su correcto funcionamiento) como en la ejecución de las aplicaciones.

I+D+I en aplicaciones de HPC

Poco valor puede aportar cualquier infraestructura si no se puede explotar adecuadamente. Si en un puerto diseñado para atracar grandes buques solo es utilizado por pequeños cargueros, la inversión realizada estará parcialmente desaprovechada. Lo mismo pasará con los grandes supercomputadores si no existe un software que pueda extraer todas las capacidades puestas a su disposición. La ventaja adicional es que cualquier mejora introducida también beneficiará a su ejecución en los computadores más modestos. Por tanto, el software y las herramientas para su desarrollo y control posterior, son una pieza clave para el avance futuro de la computación de altas prestaciones.

Si al principio de la sección anterior hablábamos de la Ley de Moore y como había contribuido a la mejora sustancial de la velocidad en que se desenvuelven las aplicaciones, los algoritmos básicos que se utilizan en algunas de las áreas explicadas en el capítulo segundo han contribuido muy significativamente a que las soluciones se alcancen más rápidamente. Por ejemplo, durante un periodo de 35 años, el cálculo del potencial electrostático inducido por una distribución de carga eléctrica ha mejorado tanto en sus necesidades de memoria (100 veces menores) como de operaciones matemáticas a realizar (con cuatro órdenes de magnitud de diferencia). El resultado es que la introducción de nuevos algoritmos (cinco en total) ha contribuido tanto como la Ley de Moore durante ese tiempo. Ahora, dado el cambio acelerado del hardware que se está viviendo, la investigación en nuevos algoritmos y herramientas de apoyo al desarrollo son fundamentales para aprovechar las nuevas configuraciones. Pero el camino no es sencillo y existe una importante brecha temporal desde la introducción del nuevo hardware y la explotación eficiente del mismo por las aplicaciones. Este hecho está contrastado en el programa que se utiliza para generar la lista TOP500 de la que hablamos anteriormente. Este está fuertemente ligado a la potencia de cálculo de la CPU, pero no mide adecuadamente el comportamiento global del computador. El resultado es que muchas aplicaciones solo pueden extraer una mínima parte de la capacidad de cálculo posible debido a las limitaciones de los otros componentes, como el acceso a la memoria o las necesidades de comunicaciones entre procesos.

Hay otros dos factores que influyen también de forma importante en las aplicaciones: la necesidad de que el mantenimiento de los códigos desarrollados sea sencillo durante periodos de 30 o más años, en donde suceden muchos cambios en el hardware, y que los resultados sean correctos independientemente del tipo de computador que se esté utilizando (que puede variar en muchos de sus componentes, como la CPU, el sistema operativo o el compilador). La reescritura completa de un código es poco habitual. De hecho, las pequeñas y medianas empresas que explotan software comercial no suelen tener capacidad para cambiar los núcleos de sus soluciones frecuentemente. De media, solo lo pueden hacer una vez cada 20 años. De hecho, algunas de las aplicaciones que se han convertido en estándares de facto en algunas áreas, han sido desarrolladas a finales de los años sesenta del siglo pasado y no han cambiado radicalmente desde entonces[16] .

16. Por ejemplo, la aplicación Gaussian de referencia en química computacional fue presentada en 1970 y su arquitectura básica no ha cambiado significativamente desde entonces, aunque se actualiza frecuentemente.

Si la contribución europea a la mejora energética de los procesadores podría ser significativa de aprovechase adecuadamente, la situación en el software podría ser mucho más sustancial, ya que nuestra posición es mucho más fuerte, con importantes núcleos de desarrollo de software científico-técnico y de herramientas de apoyo. Además, su comercialización es probablemente más sencilla, permitiendo que se haga desde cualquier lugar, siempre que se aporte un valor significativo, sin necesidad de infraestructuras importantes de comercialización y apoyo, como ocurre con el hardware.

Finalmente, antes de describir algunas de las mejoras que necesitan investigarse en el ámbito de la HPC, comentar que muchas de ellas son comunes a otras áreas de la informática, ya que los retos a los que se enfrentan son similares, desde la paralelización hasta el control del consumo energético.

En el panorama descrito de evolución futura del hardware, con un mayor número de componentes, sistemas multinúcleo y heterogéneos, el mayor reto es conseguir algoritmos que puedan explotar adecuadamente estos nuevos computadores. Para ello, la primera condición es que pierdan el mínimo tiempo posible esperando por datos, transfiriéndolos entre sus elementos (por ejemplo, de la memoria RAM del computador a la de los co-procesadores) o sincronizándose con otros procesos. Los nuevos algoritmos han de reducir por tanto al máximo esas necesidades de transferencia de información o han de poder solapar las comunicaciones con los cálculos de forma eficiente. En el caso de los procesos distribuidos, la sincronización entre ellos ha de ser mínima. Este es un campo de trabajo en donde la creatividad es fundamental y pequeños grupos de investigadores motivados pueden hacer contribuciones muy relevantes.

Dado que los computadores son y serán cada vez más complejos, la simplificación de su programación será una condición indispensable para su aprovechamiento. La paralelización automática de aplicaciones es un camino en el que se sigue investigando, pero con pocos resultados exitosos en los últimos años. Quizá solo se han conseguido avances significativos en aquellos algoritmos que se pueden dividir fácilmente en tareas independientes. Como consecuencia, esta simplificación de la programación pasa por encapsular los algoritmos básicos en bibliotecas de funciones o subprogramas que maximicen la productividad del investigador y permitan su rápida evolución manteniendo sus funcionalidades. Si ya existen interfaces estándares, como las BLAS comentadas anteriormente, el trabajo de investigación e innovación se concentra en adaptarlas a las nuevas configuraciones hardware sin que los cambios afecten a los resultados y, a

ser posible, se puedan utilizar sin cambios en los ejecutables. Existen áreas de la computación de altas prestaciones maduras en donde estos núcleos de cálculo se han identificado y casi estandarizado, pero hay otras de más reciente incorporación en donde es probablemente necesario hacer todavía ese trabajo de campo.

La simplificación de la programación pasa también por la existencia de herramientas de apoyo que permitan incrementar la productividad del programador-investigador y comprobar el correcto funcionamiento del software. Por tanto, es imprescindible mejorar los depuradores de código, que han de soportar la heterogeneidad de los nuevos computadores; los analizadores de códigos paralelos, que han de poder recoger mucha más información y han de escalar a entornos mucho mayores que los actuales; las herramientas de análisis de la eficiencia de las aplicaciones, que además de integrar los diferentes tipos de elementos hardware, han de incluir nuevas métricas como el consumo energético.

Para acortar los tiempos de desarrollo y ocultar en lo posible la complejidad cada vez mayor del hardware, permitiendo a su vez la adaptación de la aplicación en cada ejecución al entorno en donde se realiza, como ya se Comentó en el segundo capítulo, se están investigando nuevos paradigmas de programación como PGAS y lenguajes asociados como UPC, Chapel o Coarray FORTRAN. Su aceptación e implantación, si consiguen introducir mejoras significativas, será lenta debido a la competencia con otras soluciones que han demostrado durante años que son eficientes y eficaces. A pesar de la antigüedad del FORTRAN, en áreas como la química computacional, materiales o ingeniería, sigue siendo el lenguaje de programación dominante, solo desbancado en algunos casos por el C después de muchos años de adaptación y competencia leal. Sin embargo, sí existe un importante hueco para la irrupción de herramientas y lenguajes de alto nivel que incrementen la productividad en la generación de nuevos modelos que utilicen los núcleos básicos de cálculo y también faciliten el análisis, procesado de datos y la visualización, como ha ocurrido con herramientas como MATLAB, el paquete estadístico R o el lenguaje de programación Python.

Las innovaciones no están restringidas al desarrollo de las aplicaciones. También son requeridas en el control de las ejecuciones de estas y su adaptación al entorno en donde se efectúa. La eficiencia en la explotación del hardware puede variar incluso sobre el mismo computador debido a múltiples causas, desde una asignación de las CPU incorrecta hasta la compartición de las redes de comunicaciones o de los sistemas de almacenamiento. Las herramientas de monitorización y contabilidad a nivel de

aplicación serán cada vez más demandadas, incluso imprescindibles en entornos Cloud. Basándose en esta información o en la que proporcione el sistema operativo respecto a la asignación de hardware, el software ha de poder adaptarse dinámicamente a la nueva situación, con la mínima intervención del usuario final. El objetivo es responder a las expectativas del usuario, devolviendo un resultado correcto en el tiempo que él estima que es necesario. Es decir, mantener una calidad del servicio.

Esta adaptación del software también ha de ocurrir respecto a su robustez. Las aplicaciones han de responder adecuadamente al fallo, detectándolo y reaccionando al mismo. Como ya se ha comentado anteriormente, la mayor integración de los componentes y su elevado número hará que los tiempos de medios entre dos fallos de un computador disminuya significativamente. Con o sin el apoyo de la infraestructura, la aplicación ha de poder recuperarse o, al menos, poder arrancar de nuevo desde un estado anterior, sin perder todo el trabajo ya realizado. Parece casi descartado que las herramientas existentes de *checkpointing*, en donde se guarda el estado de la aplicación en un punto determinado de ejecución en los sistemas de almacenamiento, pueda emplearse en los nuevos grandes sistemas computacionales sin afectar muy negativamente a su rendimiento. O se desarrollan nuevas soluciones imaginativas para utilizar está técnica o será necesario encontrar otras alternativas. Pero el tiempo ya gastado en la ejecución no debería perderse totalmente en ningún caso, como hoy no aceptamos que una base de datos pierda algunos de sus registros por un fallo del sistema. Casi sería un escándalo, sino una falta administrativa.

Finalmente, dentro del amplio abanico de posibilidades y necesidades, existen tres que merecen comentarse. La primera es la aspiración de conseguir la integración de modelos en múltiples escalas temporales y espaciales. ¿Qué se quiere decir con esto? Supongamos que se está simulando el comportamiento de una viga. Actualmente, su simulación es macroscópica, es decir, el material será más o menos homogéneo, con propiedades físicas que son válidas a una escala de milímetros o quizá de alguna fracción de estos (esta sería la macroescala). Sin embargo, si disminuyéramos la escala espacial, como se puede apreciar a través de un microscopio, su estructura interna sería granulosa, con un comportamiento físico diferente. Es lo que se denomina mesoescala. Si se incrementara la resolución del microscopio, se llegaría a ver la estructura a nivel atómico, con interacciones y configuraciones ya reguladas mayoritariamente por la física cuántica. El deseo y la aspiración de muchos investigadores e ingenieros es poder integrar en una sola simulación las tres escalas de forma eficiente. Pero este deseo puede ser incluso imprescindible en otras áreas,

en donde además de la escala espacial, los efectos han de tener en cuenta también el tiempo, como es la biología, en donde hay procesos que duran nanosegundos que son fundamentales para comprender otros que tienen duraciones muy superiores.

La segunda es la irrupción de nuevas áreas en la computación de altas prestaciones que no han alcanzado todavía una madurez como en las más clásicas. Son una oportunidad para los investigadores de esa área de trabajo así como de especialistas en HPC y pequeñas y medianas empresas.

Para terminar este terceto, son necesarias también contribuciones relativas a la confianza que se pueda tener en los propios algoritmos que se utilizan. Las técnicas de verificación y validación del software han de adaptarse a las nuevas infraestructuras y, si es posible, simplificarse. Además, dado que los resultados son siempre una aproximación, la estimación de su incertidumbre es cada vez más necesaria.

Existen todavía múltiples necesidades, retos y oportunidades relacionados con el software. No es posible describirlas todas y, en cualquier caso, seguro que siempre quedaría alguna. Sí es posible relatar motivaciones que han de inspirar la generación de nuevas invenciones, que se pueden integrar en unas pocas:

- Productividad del programador y usuario. En el primer caso, las herramientas han de facilitarle el desarrollo y mantenimiento del software, adaptándose fácilmente a las nuevas configuraciones del hardware cada vez más grande y complejo. En el segundo, ha de poder utilizar estas aplicaciones de forma fiable y sencilla, con el menor conocimiento del hardware y su configuración (incluso, sin saber exactamente donde o como se ejecuta), pudiéndose concentrar en su problema técnico.
- Mejora en la escalabilidad y su gestión. Los actuales códigos, excepto en muy pocos casos, no pueden aprovechar el gran número de las CPU o su heterogeneidad. Son imprescindibles soluciones que no sacrifiquen la productividad comentada anteriormente pero sí permitan el aprovechamiento de todas las capacidades de los computadores, independientemente de si son grandes infraestructuras o son modestos entornos virtuales en Cloud.
- Reducción del consumo energético. Al igual que en el hardware, las aplicaciones pueden, y tienen que, contribuir a la reducción del consumo energético.
- Adaptación al fallo. El software ha de adaptarse, con o sin la ayuda de la infraestructura subyacente, al fallo del hardware. Además, sería

deseable también que pudiera detectar también sus propios fallos y reaccionar en consecuencia.

- Eficiencia. A pesar de que es una competencia entre la facilidad de mantenimiento durante muchos años y la evolución del hardware, el software utilizado en la computación de altas prestaciones no puede renunciar a ser eficiente en su ejecución, extrayendo el máximo rendimiento de las capacidades existentes.

Aunque estas motivaciones son importantes, la contribución de la I+D+I no ha de olvidarse de aportar, como ya se ha comentado, a las nuevas áreas que se están incorporando a la HPC, así como otros usos y modelos de utilización que se empiezan a demandar.

Nuevos usos del HPC

En el primer capítulo de este libro se describía el concepto de computación de altas prestaciones y posteriormente se comentaba que existían otros usos de los computadores que no encajaban exactamente en el modelo presentado. Entre estos, recientemente ha explotado el análisis de grandes volúmenes de datos (o *Big Data*). Esta es un área de la informática en efervescencia que cada vez demanda más recursos debido al crecimiento imparable de la información acumulada y que se caracteriza por las tres Vs: **volumen**, es decir, la cantidad de información acumulada es enorme, de Terabytes o muy superior; **velocidad**, la incorporación de nueva información ocurre muy rápidamente dado el gran número de elementos de entrada (desde sensores, cada vez más baratos, hasta usuarios de los sistemas, cada vez en mayor número); **veracidad**, dado su volumen y velocidad de cambio, la confianza en la información no se puede presuponer (pensemos en sensores que no están correctamente calibrados o envían incorrectamente la información de medida, o un usuario de una aplicación de redes sociales que envía un mensaje falso). A estas características identificadas para *Big Data*, en una infraestructura de HPC diseñada para prestar servicios externos o, incluso dentro de grandes compañías, hay que sumarle la variedad. Frente a un tipo de datos y organización de los mismos casi únicos, los centros de servicios han de adaptarse a múltiples formatos y herramientas de análisis. Con estos condicionantes, a la que hay que sumar que los usuarios quieren explotar la información y tener resultados casi instantáneamente, las infraestructuras utilizadas durante años en HPC son ideales para resolver sus necesidades. Sin embargo, las necesidades de la computación de altas prestaciones y de análisis de grandes volúmenes de datos son diferentes e interfieren significativamente. En algunos casos, como los sistemas de visualización, pueden ser puntos en

común muy útiles, pero la convivencia de ambos usos en una sola computadora ha de contar con soluciones innovadoras a corto plazo.

Por otro lado, esta demanda de respuesta inmediata también es deseada en algún otro campo en donde se requiere una urgencia en la respuesta. Podríamos denominarla «computación instantánea», en donde los tiempos desde que se realiza la petición de ejecución de la aplicación hasta que empieza a proporcionar información válida ha de ser mínimo (por ejemplo, pensemos en la simulación de la propagación de un incendio u otra situación de emergencia en donde un PC o portátil no es suficiente). Normalmente los computadores de altas prestaciones están gestionados para maximizar su utilización e incluso algunas de las aplicaciones requieren días o semanas de ejecución. La coexistencia de esta computación tradicional con una demanda inmediata requiere de soluciones que resuelvan las expectativas de los usuarios de ambos modelos de uso. ¿Quizá el Cloud sea una solución en este caso?

Finalmente, la computación, siguiendo el modelo de Software-as-a-Service comentado en el capítulo anterior, tiene que garantizar una mínima calidad, independientemente del número de usuarios que concurrentemente estén solicitando el servicio. Las páginas WEB son un ejemplo de lo que puede ocurrir cuando no se dimensiona adecuadamente la infraestructura que lo proporciona. Se puede sobreestimar, con unas inversiones improductivas o, por el contrario, subestimar trayendo como consecuencia fallos inaceptables que pueden hundir el negocio. Además, se han incorporado nuevos dispositivos como los teléfonos inteligentes o las *tablets*, que están dominando las ventas en el mercado de consumo. La HPC no es diferente y no puede estar ajeno a estos movimientos. Ha de innovar en los modos de gestión de la infraestructura, su forma de acceso y los interfaces de las aplicaciones para adaptarse a estos cambios.

Como se ha visto a lo largo de este capítulo, a pesar de la larga historia de la computación de altas prestaciones, las oportunidades de innovación son muy importantes y relevantes. Dado que es y será imprescindible computar para competir, la inversión en su evolución a través de la investigación, desarrollo e innovación es imperativa y no es justificable la pérdida de oportunidades que abre.

«La capacidad o falta de capacidad de las sociedades para dominar la tecnología, y en particular las que son estratégicamente decisivas en cada periodo histórico, define en buena medida su destino»

Manuel Castell,

«La era de la información. Vol.1 La sociedad red». 1996

CAPÍTULO 6.
LAS OPORTUNIDADES

Empezaba este pequeño ensayo con la pregunta de por qué el 97% de las empresas que utilizan la computación de altas prestaciones la consideran imprescindible para su competitividad. Después de leer los capítulos anteriores, cada lector habrá sacado sus propias conclusiones y habrá visto las oportunidades que puede aportar a su ámbito de trabajo. No en vano, las innovaciones más radicales y efectivas suelen venir del cruce de caminos entre dos conocimientos, dos áreas dispares, frecuentemente alejadas. Una que busca la solución a un problema y otra que la tenía, pero para otros campos. El contacto multidisciplinar es, por tanto, el frutal en donde hay que buscar las oportunidades. Es la forma de trabajo de la Ciencia Computacional que ya se comentó anteriormente.

A pesar de que realmente la computación de altas prestaciones ha estado ligada desde sus orígenes a la ciencia y la técnica, no ha tenido un protagonismo en la sociedad como debiera. Su uso ha estado restringido hasta hace bien poco a grandes empresas, instituciones públicas y universidades, con insuficiente presencia en otros ámbitos, como las pequeñas y medianas empresas. Con los cambios en el hardware que se utilizan, pasando de computadores con arquitecturas específicas como los procesadores vectoriales a los clústeres de computación fabricados

con componentes ya existentes en el mercado de consumo, la tecnología está disponible a precios asequibles. El coste del software comercial sigue siendo una barrera para algunas pequeñas empresas, pero que se puede soslayar al existir versiones de software libre, menos amigables pero igualmente efectivas. En cualquier caso, las soluciones a ambos obstáculos pueden venir de la utilización del reciente paradigma de computación en la nube o Cloud, en donde las inversiones de capital pueden convertirse en gasto operativo, limitando el riesgo para los pequeños negocios, sobre todo los emergentes.

Las consultoras de análisis de tendencias que han examinado el campo de la HPC durante los últimos años han encontrado que, excepto con un pequeño parón cuando comenzó la crisis en la que estamos inmersos, ha crecido a un ritmo constante. Lo más interesante no es el pasado, sino el futuro. La consultora IDC predijo en julio de 2013 que el mercado global de HPC crecería un 7% hasta 2016, donde termina su proyección. La definición que utiliza esta consultora es diferente al utilizado aquí y este crecimiento está influenciado por la irrupción del procesado de grandes volúmenes de datos o *Big Data*, pero en cualquier caso muestra la fortaleza con la cual se está introduciendo esta tecnología en todos los ámbitos de nuestro entorno productivo. Estamos en un momento decisivo, con varios cambios radicales en marcha: dispositivos móviles como smartphones y tablets, *Big Data* y HPC. Tenemos que dominar las bases tecnológicas de estos tres movimientos para no perder el momento histórico en el que se encuentra el mundo. Y aprovecharlo para impulsar una mejora sustancial de nuestra calidad de vida.

Pero, ¿dónde está el negocio? ¿Cómo y dónde se puede emprender una nueva actividad exitosa? La respuesta ha de venir de un análisis propio en donde se identifique qué sabemos hacer, qué demanda o demandará la sociedad o la economía de lo que hemos localizado, y con qué necesitamos complementarlo. La cadena de valor de la computación de altas prestaciones es amplia y seguro que existe un lugar en donde los conocimientos e inquietudes del emprendedor tienen cabida. En el modelo descrito en el primer capítulo caben muchas habilidades diferentes: la capacidad de modelar el problema matemáticamente, el conocimiento de programación científico-técnica para llevar el modelo a un código de ordenador fiable y eficiente, la formación para diseñar los experimentos que puedan verificar y validar la solución computacional construida y la experiencia para ejecutar la aplicación resultante eficazmente. Pero la principal habilidad es la capacidad para identificar el problema y tener la voluntad de solucionarlo, ya que muchas veces simplemente se asume sin más.

Las oportunidades individuales son tan numerosas y particulares que no es posible enumerarlas. En cambio, sí es factible el describir como algunos colectivos pueden aprovechar el resurgir de la computación de altas prestaciones.

Un primer grupo lo componen las pequeñas y medianas empresas de la manufactura, incluyendo también a las microempresas. Las grandes compañías explotan a diario las capacidades de modelado y simulación para diseñar y mejorar sus productos y procesos productivos. Las empresas de menor tamaño también pueden beneficiarse de esta tecnología si apuestan por ello. Los costes actuales de inversión en infraestructura han decrecido gracias a la utilización de hardware de uso común. Además, se ha mostrado en los capítulos anteriores que es posible utilizar infraestructura virtualizada en la nube para poder reducir aún más los costes, haciendo el gasto únicamente cuando es necesario para el proyecto. La inversión en software sigue viéndose como una barrera, pero también existe soluciones de código libre de alta calidad, quizá no tan amigable, pero de igual rendimiento y funcionalidad (en algunos casos superior) que el comercial y que permite introducir estos procesos de simulación en la empresa. La formación del personal es aquí clave, pero en cualquier caso, necesaria para competir en el futuro. Y si no se puede utilizar esta técnica básica industrial dentro de la empresa, siempre existen servicios externos que estarían más que dispuestos a ayudar. Lo que no es aconsejable es quedarse parados y dejar pasar esta oportunidad de producir mejores productos que la competencia, a menor coste y de mayor calidad. Estas tres ventajas han sido demostradas ya por las empresas más competitivas, sean grandes o pequeñas.

Si el conocimiento está más cercano al hardware y las infraestructuras de apoyo, todavía hay mucho recorrido. Claramente, desarrollar las nuevas CPU, memorias RAM o discos duros no está al alcance de muchas empresas. Sin embargo, como la HPC se basa cada día más en electrónica de uso común, cualquier empresa de fabricación de dispositivos que utilicen procesadores tiene capacidad de introducirse en este mercado. Si además tiene experiencia en electrónica de bajo consumo eléctrico y en fabricación de series cortas, tendrá una parte del camino andado y podrá diversificar su negocio. La alianza con expertos en computación será clave para el desarrollo del producto, sobre todo para darle una credibilidad en el mercado. No solo es necesario tener un buen producto, sino que hay que demostrarlo.

Pero esta área cercana a las infraestructuras tiene también una segunda vertiente de interés: la gestión del consumo eléctrico. Los grandes centros

de computación han invertido mucho tiempo y esfuerzo en tratar de reducir los costes eléctricos, con gran éxito en la mayoría de los casos, obteniendo retornos de la inversión en poco tiempo. El reto es ahora trasladar las técnicas aplicadas a otros entornos, como los centros de trabajo de las pequeñas y medianas empresas. Es previsible que el coste de la electricidad siga subiendo en el futuro, con lo que las inversiones en la eficiencia energética serán rentables y cada día más demandadas. Finalmente, existe un nicho de mercado asociado a la administración, monitorización y control integral de los centros de datos, tanto grandes como pequeños. Para los grandes, serán necesarios nuevos conceptos y herramientas para poder abordar instalaciones de millones de unidades de procesamiento y elementos diferentes, incluyendo las condiciones climáticas.

Los servicios en la nube de infraestructuras virtualizadas o no, específicas para HPC son un campo todavía poco explotado. Hay aún escasos proveedores especializados en esta área, que por sus características necesita de soluciones alejadas de las convencionales, como se ha visto en el cuarto capítulo. También es un área en donde se requieren innovaciones que es necesario investigar y desarrollar, siendo por tanto un ámbito de trabajo tanto para el emprendedor como para el investigador.

Sin embargo, aunque las infraestructuras y el hardware son nichos importantes de negocio, el gran baúl de las oportunidades está en el software. El colectivo de desarrolladores está de enhorabuena. Por un lado, crear una infraestructura computacional, grande o pequeña, es más sencillo que posteriormente sacarle todo el partido posible. La evolución del hardware en los últimos años ha sido vertiginosa, pero los creadores de aplicaciones necesitan más tiempo para comprender todas sus capacidades y obtener el máximo rendimiento. Por ello, muchos de los códigos actuales tendrán que ser reescritos o adaptados en los próximos años. Esto será necesario tanto para las aplicaciones comerciales o las libres de uso general, como para aquellas internas de las compañías que han desarrollado soluciones propias durante los últimos años. Muchos de estos cambios requerirán de expertos externos de programación HPC, escasos en este momento. También son imprescindibles, como se ha mostrado en el capítulo anterior, nuevas herramientas que faciliten la programación y algoritmos que puedan ser eficientes en los futuros computadores.

Por otro lado, tanto las áreas tradicionales del uso de la HPC como las recientes incorporaciones como la bio-sanitaria o las finanzas, tienen una larga lista de necesidades de soluciones computacionales. La inclusión más que probable de la computación de altas prestaciones en el diagnóstico y tratamiento de pacientes abre un abanico de posibilidades enorme

para la innovación, con el traslado de las investigaciones en marcha en universidades y centros de investigación a la clínica diaria. Este modelo ya ha sido demostrado anteriormente en el sector de la ingeniería, en donde muchas de los paquetes comerciales más utilizados salieron de esos laboratorios y crecen con nuevas opciones con su ayuda. La ventaja es que ahora, gracias a las nuevas formas de distribución de software es posible desarrollarlo y comercializarlo desde cualquier parte del mundo. La utilización del modelo de software como servicio en la nube ofrece la posibilidad de abarcar un mercado global. Lo importante es tener una buena solución que sea útil y eficaz. De hecho, una fracción importante del software científico-técnico comercial es de pequeñas y medianas compañías con conocimiento especializado. Pero es importante recordar que el software para HPC ha de contar con las verificaciones y validaciones correspondientes, con lo que la colaboración con todos los actores de la cadena de valor es muy importante. Aunque a la vez hace más atractivo e interesante el trabajo.

Un último colectivo que está de enhorabuena es el de la investigación. Como se ha visto en el capítulo anterior, los retos que hay que batir en los próximos años para alcanzar los objetivos marcados son muy importantes. Tanto si el investigador está dedicado a la investigación relacionada con el hardware como si trabaja en software o nuevos algoritmos, las posibilidades de hacer contribuciones básicas y aplicadas relevantes está a su alcance. También es posible contribuir desde otras áreas de la ciencia, aportando a los grandes retos que necesitan computación, considerada en la actualidad como un laboratorio más. La colaboración entre especialistas de diversas áreas será decisiva para avanzar en estos retos científicos, como puede ser el Human Brain Project.

En los diversos campos existen ejemplos de pequeñas compañías que han encontrado su nicho de mercado, dado que el conocimiento necesario es muy especializado y, habitualmente, poco rentable para las grandes compañías, que muchas veces las subcontratan para sus propios proyectos. Existen casos conocidos en el sector como la compañía francesa CAPS que ha desarrollado y comercializado un compilador eficiente basado en el estándar OpenACC. Desde Granada, la compañía CATÓN se ha especializado en todo lo relacionado con la HPC, incluyendo la eficiencia energética de los centros que alojan los computadores y la monitorización a todos los niveles. En prestación de servicios de computación en la nube ya se ha hablado de Gompute u Oxalya, ambas dedicadas a la prestación de servicios HPC.

La prueba de que no es necesario estar en grandes capitales para comercializar buen software podría ser el conjunto de soluciones de interface de usuario para HPC de la pequeña compañía italiana NICE, radicada en la ciudad de Asti en el Piamonte, con dos productos robustos, uno de control de las ejecuciones (EigenFrame) y otro de visualización remota, orientada a la ingeniería, que como se vio anteriormente es un campo que necesita mejoras. Otro ejemplo, pero relacionado con el campo de química y desde Santiago de Compostela es Mestrelab Research, con una solución que domina el mercado de análisis de datos de Resonancia Magnética Nuclear. Estos son solo algunos casos en donde las empresas han encontrado donde aplicar sus conocimientos apoyados en o para la computación de altas prestaciones, pero existen muchos más que cubren un amplio espectro con éxito.

En resumen, con la información que tenemos actualmente y que se ha expuesto a lo largo del libro, se puede concluir que la computación de altas prestaciones es un mundo de posibilidades y de oportunidades. Es una tecnología con mucho pasado, pero con un brillante fututo. Es un conocimiento que tiene cada vez más demanda, tendrá cada día una mayor presencia en nuestra economía y nos afectará más en nuestra vida diaria. En consecuencia, es necesario dominarla y explotarla, ya que si ahora es importante, en el futuro no computar será no competir.

Bibliografía

Para cada capítulo se incluye una serie de referencias bibliográficas que permiten ampliar su contenido. No es una referencia bibliográfica exhaustiva y detallada, sino aquella seleccionada por su interés o relevancia. Incluso hay duplicidades entre capítulos, con el fin de facilitar su localización por el lector cuando haya atraído su interés durante o después de la lectura.

Capítulo 1

Comisión Europea. *High-Performance Computing Europe's Place in a Global Race. COM(2012) 45 Final*, 2012.

Joseph, Earl C, Steve Conway, Chris Ingle, Gabriella Cattaneo, Nathaniel Martinez, and Cyril Meunier.

A Strategic Agenda for European Leadership in Supercomputing: HPC 2020 — IDC Final Report of the HPC Study for the DG Information Society of the European Commission, 2010.

Oberkampf, W. L., and Ch. J. Roy. *Verification and Validation in Scientific Computing*. New York: Cambridge University Press, 2010.

González Redondo, Francisco A. «Leonardo Torres Quevedo (1852-1936) 2a Parte. Automática, Máquinas Analíticas.» *La Gaceta De La RSEM 8*, no. 1 (2005): 267–293.

Torres y Quevedo, Leonardo. «XIX. — Ensayos Sobre Automática.— Su Definición. Extensión Teórica De Sus Aplicaciones.» *Revista De La Real Academia De Ciencias Exactas, Físicas y Naturales* Tomo XII (1913): 391–419. http://www.rac.es/ficheros/Revistas/REV_20100220_03288.pdf.

Rodríguez, Xerardo. «Ramón Verea García – El Inventor De La Calculadora.» *Galicia Única* – Revista Digital Independiente, n.d. http://www.galiciaunica.es/gente/?page_id=455.

Capítulo 2

Hennessy, J.L., and D.A. Patterson. C*omputer Architecture. A Quantitative Approach*. Morgan Kaufmann, 2011.

Flynn, Michael J. «Some Computer Organizations and Their Effectiveness.» *IEEE Transactions on Computers* C-21, no. 9 (September 1972): 948 960.http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5009071.

Dongarra, Jack. V*isit to the National University for Defense Technology Changsha, China*, 2013. http://www.netlib.org/utk/people/JackDongarra/PAPERS/tianhe-2-dongarra-report.pdf.

OpenMP Architecture Review Board. «OpenMP Application Program Interface», 2013.

Chapman, Barbara, Gabriele Jost, and Ruud Van Der Pas. *Using OpenMP: Portable Shared Memory Parallel Programming*. Vol. 10. The MIT Press, 2008.

Message Passing Interface Forum. «MPI: A Message-Passing Interface Standard». High-Performance Computing Center Stuttgart, 2012.

Gropp, William D, Ewing L Lusk, and Anthony Skjellum. *Using MPI: Portable Parallel Programming with the Message-passing Interface*. Vol. 1. the MIT Press, 1999.

IEEE Computer Society,. «IEEE Std 754TM-2008 (Revision of IEEE Std 754-1985), IEEE Standard for Floating-Point Arithmetic», 2008.

Goldberg, David. «What Every Computer Scientist Should Know About Floating-point Arithmetic.» *ACM Computing Surveys 23*, no. 1 (March 1, 1991): 5–48. http://portal.acm.org/citation.cfm?doid=103162.103163.

Vajda, András. *Programming Many-Core Chips*. Boston, MA: Springer US, 2011. http://link.springer.com/10.1007/978-1-4419-9739-5.

Capítulo 3

Lery, Thibaut, Mrio Primicerio, Maria J Esteban, Magnus Fontes, Yvon Maday, Volker Maday, Volker Mehrmann, et al. *FORWARD LOOK Mathematics and Industry Success Stories*, 2011.

«AEMET. Predicción Numérica Del Tiempo», n.d. http://www.aemet.es/es/idi/prediccion/prediccion_numerica.

Lorenz, Edward W. «Does the Flap of a Butterfly's Wings in Brazil Set Off a Tornado in Texas?» In *139th Meeting of AAAS*, 1972. http://web.mit.edu/lorenzcenter/about/LorenzPubs/Butterfly_1972.pdf.

Lynch, Peter. «The Origins of Computer Weather Prediction and Climate Modeling.» *Journal of Computational Physics 22*7, no. 7 (March 2008): 3431–3444. http://linkinghub.elsevier.com/retrieve/pii/S0021999107000952.

Berry, Romain. «An overview of value-at-risk: part ii – historical simulations var.» *Investment Analytics and Consulting*, no. December (2008).

———. «An overview of value-at-risk: part iii – monte carlo simulations var.» *Investment Analytics and Consulting*, no. March (2009).

Black, Fischer, and Myron Scholes. «The Pricing of Options and Corporate Liabilities.» *Journal of Political Economy 81*, no. 3 (January 1973): 637.

Glasserman, Paul. *Monte Carlo Methods in Financial Engineering*. Springer, 2003.

«Directory of Computer-aided Drug Design Tools», http://www.click2drug.org/.

Dalkas, Georgios a, Dimitrios Vlachakis, Dimosthenis Tsagkrasoulis, Anastasia Kastania, and Sophia Kossida. «State-of-the-art Technology in Modern Computer-aided Drug Design.» *Briefings in Bioinformatics* (November 12, 2012).

Doi, Kunio. «Computer-aided Diagnosis in Medical Imaging: Historical Review, Current Status and Future Potential.» *Computerized Medical Imaging and Graphics: the Official Journal of the Computerized Medical Imaging Society* 31, no. 4–5 (2007): 198–211.

Giger, Maryellen L., Heang-Ping Chan, and John Boone. «Anniversary Paper: History and Status of CAD and Quantitative Image Analysis: The Role of Medical Physics and AAPM.» *Medical Physics* 35, no. 12 (2008): 5799.

Neidle, Stephen, and Roderick E Hubbard, eds. Structure Based Drug Discovery. *An Overview*. RCS Publising, 2006.

Halperin, Inbal, Buyong Ma, Haim Wolfson, and Ruth Nussinov. «Principles of Docking: An Overview of Search Algorithms and a Guide to Scoring Functions.» *Proteins* 47, no. 4 (June 1, 2002): 409–43.

Hines, M L, and N T Carnevale. «The NEURON Simulation Environment.» *Neural Computation* 9, no. 6 (August 15, 1997): 1179–209.

Markram, Henry. «A Brain in a Supercomputer», 2009. http://www.ted.com/talks/henry_markram_supercomputing_the_brain_s_secrets.html.

Kearey, Philip, Michael Brooks, and Ian Hill. *An Introduction to Geophysical Exploration (Tercera Edición)*. Blackwell Science, 2002.

Johnston, David H. (Editor). *Methods and Applications in Reservoir Geophysics*. Society of Exploration Geophysicist.

Yilmaz, Öz. *Seismic Data Analysis.* Society of Exploration Geophysicist.

Capítulo 4

Carr, Nicholas. *The Big Switch: Rewiring the World, from Edison to Google*. Primera Ed. W.W. Norton & Company, Inc., 2008.

Mell, Peter, and Timothy Grance. The NIST *Definition of Cloud Computing. Special Publication 800-145*, 2011.

De Alfonso, Carlos, Miguel Caballer, Fernando Alvarruiz, and Germán Moltó. «An Economic and Energy-aware Analysis of the Viability of Outsourcing Cluster Computing to a Cloud.» *Future Generation Computer Systems 29*, no. 3 (March 2013): 704–712. http://linkinghub.elsevier.com/retrieve/pii/S0167739X12001720.

Autores, Varios. *The Ubercloud Experiment: Compendium of Case Studies*. Tabor Communications Inc, 2013.

Calheiros, Rodrigo N., Adel Nadjaran Toosi, Christian Vecchiola, and Rajkumar Buyya. «A Coordinator for Scaling Elastic Applications Across Multiple Clouds.» *Future Generation Computer Systems 28*, no. 8 (October 2012): 1350 –1362. http://linkinghub.elsevier.com/retrieve/pii/S0167739X12000635.

Expósito, Roberto R., Guillermo L. Taboada, Sabela Ramos, Juan Touriño, and Ramón Doallo. «Performance Analysis of HPC Applications in the Cloud.» *Future Generation Computer Systems 29*, no. 1 (January 2013): 218–229. http://linkinghub.elsevier.com/retrieve/pii/S0167739X12001458.

Gentzsch, Wolfgang. «HPC as a Service – Key to Mainstream HPC or Wishful Thinking?», 2012. http://www.digitalmanufacturingreport.com/dmr/2012-08-28/hpc_as_a_service_%E2%80%93_key_to_mainstream_hpc_or_wishful_thinking_.html?page=1.

Gomez, Andrés, Luis M. Carril, Raul Valin, J.C. Mouriño, and Carmen Cotelo. «Experimenting Virtual Clusters on Distributed Cloud Environments Using BonFIRE.» In *Fire Engineering Workshop*. Ghent (Belgium), 2012. http://www.ict-fire.eu/events/fire-engineering-workshop.html.

González-Castaño, Diego M, Javier Pena, Faustino Gómez, Araceli Gago-Arias, Francisco J González-Castaño, Daniel a Rodríguez-Silva, Andrés Gómez, Carlos Mouriño, Miguel Pombar, and Manuel Sánchez. «eIMRT: a Web Platform for the Verification and Optimization of Radiation Treatment Plans.» *Journal of Applied Clinical Medical Physics / American College of Medical Physics 10*, no. 3 (January 2009): 2998. http://www.ncbi.nlm.nih.gov/pubmed/19692983.

Gupta, Abhishek, Laxmikant V Kale, Filippo Gioachin, Verdi March, Chun Hui Suen, Bu-sung Lee, Paolo Faraboschi, Richard Kaufmann, and Dejan Milojicic. *The Who , What , Why and How of High Performance Computing Applications in the Cloud (HPL-2013-49)*, 2013. http://www.hpl.hp.com/techreports/2013/HPL-2013-49.pdf.

Metsch, Thijs, and Andy Edmonds. «Open Cloud Computing Interface - Infrastructure.» *Open Grid Forum* GFD-184 (2011).

———. «Open Cloud Computing Interface - RESTful HTTP Rendering.» *Open Grid Forum* GFD-185 (2011).

Nyrén, Ralf, Andy Edmonds, Alexander Papaspyrou, and Thijs Metsch. «Open Cloud Computing Interface - Core.» *Open Grid Forum* GFD-183 (2011).

Schubert, Lutz, and Keith (Editors) Jeffery. Advances in Clouds. Research in Future Cloud Computing, 2012.

Schubert, Lutz, Burhard Neidecker-Lutz, and Keith (Editors) Jeffery. *The Future of Cloud Computing. Opportunities for European Cloud Computing Beyond 2010*, 2010.

Schubert, Lutz, Burkhard Neidecker-Lutz, and Keith (Editors) Jeffery. *A Roadmap for Advanced Cloud Technologies Under H2020*, 2012.

Tordsson, Johan, Rubén S. Montero, Rafael Moreno-Vozmediano, and Ignacio M. Llorente. «Cloud Brokering Mechanisms for Optimized Placement of Virtual Machines Across Multiple Providers.» *Future Generation Computer Systems 28*, no. 2 (Februa-

ry 2012): 358–367. http://linkinghub.elsevier.com/retrieve/pii/S0167739X11001373.

Yelick, Katherine, Susan Coghlan, Brent Draney, Lavanya Ramakrishnan, Adam Scovel, Iwona Sakrejda, Anping Liu, et al. «The Magellan Report on Cloud Computing for Science» (2011).

Capítulo 5

Frascati Manual 2002. OECD Publishing, 2002.

EUROSTAT, and OCDE. *Manual De Oslo. Guía para la recogida e interpretación de datos sobre innovación*, 2005.

Reglamento (CE) No 800/2008 de la comisión De 6 De Agosto De 2008 Por El Que Se Declaran Determinadas Categorías De Ayuda Compatibles Con El Mercado Común En Aplicación De Los Artículos 87 y 88 Del Tratado (Reglamento General De Exención Por Categorías), 2008.

Comisión Europea. *Community framework for state aid for research and development and innovation (2006/C 323/01)*, 2006.

Communications Networks, Content and Technology Directorate-General Unit CONNECT A3 – Complex Systems and Advanced Computing. *Towards a Breakthrough in Software for Advanced Computing Systems*, 2012.

Lecomber, David, Ian Phillips, Francesc Subirada, François Bodin, Jean Gonnord, Sanzio Bassini, Giampietro Tecchiolli, et al. *ETP4HPC Strategic Research Agenda Achieving HPC Leadership in Europe*, 2013.

Aloisio, G., M. Cocco, J. Baldasano, J. Biercamp, R. Budich, M.A. Foujols, M. Hamrud, et al. *European Exascale Software Initiative Working Group Report on Weather , Climate and Solid Earth Sciences*, 2011.

Andre, Jean Claude, and Philippe Ricoux. *European Exascale Software Initiative D3. 3 Working Group Report on Industrial and Engineering Applications: Energy , Transports*, 2011.

Ashworth, Mike, and Andrew Jones. *European Exascale Software Initiative D4 . 6 Working Group Report on Scientific Software Engineering*, 2011.

Berthou, J.Y. *European Exascale Software Initiative. Deliverable D5.6 Final Report on Roadmap and Recommendations Development*, 2012.

Cappelo, Frank, and Bernd Mohr. *European Exascale Software Initiative D4 . 4 Working Group Report on Software Eco- System*, 2011.

Duff, Ian, and Andreas Grothey. E*uropean Exascale Software Initiative Report D4 . 5 Working Group Report on Numerical Libraries , Solvers and Algorithms*, 2011.

Duranton, M., D. Black-Schaffer, S. Yehia, and K. De Bosschere. *Computing Systems: Research Challenges Ahead The HiPEAC Vision 2011/2012*, 2011.

Foray, Dominique, Paul A David, and Bronwyn Hall. *Knowledge Economists Policy Brief n ° 9 Smart Specialisation – The Concept*, 2009.

Huber, Herbert, and Riccardo Brunino. *European Exascale Software Initiative D4 . 3 Working Group Report on Hardware Roadmap , Links and Vendors*, 2011.

Joseph, Earl C, Steve Conway, Chris Ingle, Gabriella Cattaneo, Nathaniel Martinez, and Cyril Meunier. *A Strategic Agenda for European Leadership in Supercomputing: HPC 2020 — IDC Final Report of the HPC Study for the DG Information Society of the European Commission*, 2010.

Mohr, Bernd. European Exascale Software Initiative D4 . 7 – *Synthesis of the Four Working Groups*, 2012.

Sutmann, Godehard, and Jean-Philippe Nominé. *European Exascale Software Initiative D3 . 5 Working Group Report on Fundamental Sciences (Chemistry , Physics)*, 2011.

European Exascale Software Initiative Working Group Report on Life Science and Health Activities, 2011.

Sawyer, Mark, and Mark Parsons. *A Strategy for Research and Innovation Through High Performance Computing*, 2011.

———. *Challenges Facing HPC and the Associated R & D Priorities: a Roadmap for HPC Research in Europe*, 2012.

Asanovic, Krste, Bryan Christopher Catanzaro, David A Patterson, and Katherine A Yelick. *Technical Report No. UCB/EECS-2006-183:The Landscape of Parallel Computing Research: A View from Berkeley*, 2006.

Capítulo 6

Duranton, M., D. Black-Schaffer, S. Yehia, and K. De Bosschere. *Computing Systems: Research Challenges Ahead The HiPEAC Vision 2011/2012*, 2011.

García Tobío, Javier, Ignacio López Cabido, Carlos Fernández Sánchez, and Javier Cacheiro López. «Energy Efficiency Policy at CESGA», 2013. https://www.cesga.es/es/biblioteca/downloadAsset/id/731.

Joseph, Earl C, Steve Conway, Chris Ingle, Gabriella Cattaneo, Nathaniel Martinez, and Cyril Meunier. *A Strategic Agenda for European Leadership in Supercomputing: HPC 2020 — IDC Final Report of the HPC Study for the DG Information Society of the European Commission*, 2010.

Woodie, Alex. «IDC Forecasts 7 Percent Annual Growth for Global HPC Market.» HPC Wire, 2013. http://www.hpcwire.com/hpcwire/2013-07-09/idc_forecasts_7_percent_annual_growth_for_global_hpc_market.html.

Andrés Gómez Tato (Lugo 1966) es Dr. en Física. Tiene más de 12 años de experiencia en computación de altas presataciones, dirigiendo el departamento de Aplicaciones y Proyectos del Centro de Supercomputación de Galicia y más de 20 en el sector TIC. En su actual puesto, asesora a investigadores y empresas sobre las posibilidades de la computación de altas prestaciones y su implantación en nuevos servicios, incluyendo su despliegue en Cloud. Además es miembro del Grupo de Expertos sobre Investigación en Computación Cloud de la Comisión Europea, en donde se han propuesto a la Comisión las necesidades de investigación, desarrollo e innovación en este ámbito dentro del programa Horizonte 2020.