

Aplicaciones en Ada: Fuerza Aérea y medicina aeroespacial

JOSE MANUEL DE LA RIVA GRANDAL, Tte. E.T.S.; VICENTE NAVARRO RUIZ, Cte. Médico E.A. y
JULIAN FRAILE BLANCO, Col. Médico E.A.

ORIGENES DE LA INFORMATICA MILITAR

Hardware:

Se suele admitir como el comienzo de la era informática, con la aparición de tarjeta de programa almacenado para telares de Jacquard al comienzo del siglo XIX. En 1930 aparece la primera máquina de calcular, Model 1, desarrollada por Stibiz en los Laboratorios Bell. El primer terminal remoto aparece en septiembre de 1940 en un mitin de la Sociedad Matemática Americana en Dartmouth. Estas máquinas de calcular se utilizaban en cálculos de censos estadísticos, impuestos, sucesos astronómicos y cálculos sobre mareas.

Un paso importante se da con la aparición del tabulador de paso cruzado 601 de IBM. A partir de este momento el desarrollo de los computadores fue impulsado por necesidades militares. Primeramente el control y dirección de tiro, utilizando tanto computadores analógicos como digitales (la máquina analítica Vanevar Bush utilizaba programas para control de tiro en buques de guerra). Estos estudios fueron realizados en el M.I.T. (Massachusetts Institute Technology) y en el Laboratorio de Investigación Balística del Ejército.

Aunque en los Estados Unidos por aquel entonces existía capacidad para haber construido una máquina de calcular con relés electromecánicos o electrónicos, esto no se llevó a cabo hasta el comienzo de la Segunda Guerra Mundial. En esas fechas coexistían los computadores analógicos con los digitales, pero en 1953 la pérdida de terreno de los computadores analógicos era evidente, aunque todavía se utilizaban en algún medio militar, como era el caso del Typhoon de RCA, utilizado para desarrollar los sistemas de guía de los misiles Sparrow y Sidewinder.

En la Segunda Guerra Mundial aparecieron en ambos lados del Atlántico máquinas de programa almacenado La Aiken-IBM Mark I, de la Universidad de Harvard, y la Eckert-Mauchly Eniac, de la Universidad de Pensilvania, por los EE.UU., y Colossus, por Gran Bretaña.

La estimulación militar directa fue evidente, sobre todo en el caso del Colossus, el cual resolvió problemas tan graves en aquel momento como: análisis criptográfico, problemas balísticos y procesos relacionados con la bomba atómica, como la difusión de los gases.

Conviene destacar que mientras que los británicos utilizaban ordenadores con tubos de vacío, más rápidos), los otros países comprometidos en la guerra, como los rusos, alemanes y japoneses, utilizaban máquinas automáticas de calcular mecánicas o electromecánicas.

El Colossus fue muy eficaz en el análisis criptográfico del código secreto militar de los alemanes, por lo que impulsó la construcción de más Colossus, tales como Bombe y Heath Robinson, ambos con relés electromagnéticos y dedicados al análisis criptográfico. Posteriormente, apareció una versión de Heath Robinson llamado "Rube Goldher", el cual era totalmente electrónico (30-80 tubos de vacío) y más tarde el Colossus Mark I, de 1.500 válvulas de vacío. En junio de 1944 vio la luz el Colossus Mark II, de 2.500 válvulas. Es curioso observar que debido a la confidencialidad estos datos no han sido conocidos hasta épocas recientes, cosa que afortunadamente no ocurrió con el Eniac. En efecto, la máquina Eniac fue desarrollada en el Laboratorio de Investigación Balística del Ejército, para solucionar problemas balísticos y resultados electorales y demográficos, desembocando con el tiempo en el Univac I.

En Harvard se desarrollaron los Mark I, Mark II, Mark III y Mark IV, de aplicaciones científicas (tablas de computación de Bessel y funciones de Hankel).

El proyecto Whirlwind elaborado en el M.I.T. era el prototipo de Defensa Aérea computerizada y dio lugar al SAGE (Semi-automatic Ground Environment), de éste se llegó al C² (Comando y Control), más tarde al C³ (Comando, Control y Comunicaciones) y finalmente al C³¹ (Comando, Control, Comunicaciones e Inteligencia).

Británicos y norteamericanos eran los únicos que al final de la guerra disponían de máquinas de calcular electrónicas. A partir de ese instante, el desarrollo del Hardware fue aumentando hacia mejores memorias, válvulas y dispositivos de entrada/salida. Pero también la arquitectura de las máquinas de programa almacenado y el diseño de lenguajes fue reconocida como crítica.

Software:

Inicialmente, cada máquina estaba acompañada de su propio conjunto de programas, hecho que provocó confusión en el desarrollo del lenguaje y por lo tanto en un progreso débil. Esto duró hasta comienzos de los 50, con la aparición del Fortran, el cual eliminó las camarillas de programadores de ensamblador que monopolizaban las máquinas, siendo éstas accesibles a partir de entonces a la comunidad científica en general. En el desarrollo del Fortran participó ampliamente Backus, el cual también colaboró en el Ina y el Algol 58.

Al principio de los 60, se introdujo el Algol, siendo un dialecto de éste (Algol 60) el que se impuso en la URSS y a través del cual se desbancó a los programadores asistentes involucrados en código máquina, apareciendo en su lugar los programadores de sistemas.

Más tarde apareció el Cobol, a través de la comisión CODASYL, auspiciada por el Gobierno de EE.UU., como respuesta a la falta de estandarización en el complejo mundo de las relaciones comerciales.

Nuevamente el Pentágono, a la vista de que el coste de mantenimiento del software es aproximadamente un 80 por 100 del coste del ciclo de vida de los sistemas defensivos computarizados (el otro 20 por 100 corresponde al hardware), ha promovido otro proyecto en el área de los lenguajes mediante el Ada.

PRESENTE Y FUTURO DE LA INFORMÁTICA MILITAR

(Todo sistema moderno de defensa es críticamente dependiente de los computadores y de su software). Subsecretario de Defensa de EE.UU.

Hoy en día la relación entre la informática y la Defensa sigue siendo estrecha, estando presentes los procesadores en toda clase de armas y equipos informáticos, como carros, aeronaves, barcos, simuladores aéreos, sistemas de defensa aérea, grandes sistemas de transmisiones o C3I (mando, control, comunicaciones e Inteligencia).

Están ya operacionales sistemas de mando e información como el alemán EIFEL para asegurar la defensa aérea del territorio y dirigir los medios de combate de la aviación, soportado por ordenadores Siemens 4004.

Los franceses tienen un sistema similar para la protección aérea de su territorio, basado en el sistema STRIDA, con calculadoras IBM 370. Este sistema goza de amplia cobertura, pues está enlazado al sistema francés de control aéreo civil CAUTA y a las redes NADGE de la OTAN y a la española COMBAT GRANDE.

A juzgar por los proyectos actuales (Strategic Defense Initiative, Eureka), el futuro de la relación entre la informática y la defensa será más vinculante. Desde el punto de vista del hardware existe todo un conjunto de programas de investigación subvencionados por el DoD, a través de DARPA (Defense Advanced Research Projects Agency), tales como semiconductores de arseniuro de galio, fibras ópticas y circuitos integrados de alta velocidad. Por lo que respecta al software cabe mencionar al programa del lenguaje Ada, al proyecto STARTS y a la creación del Instituto de Tecnología del software.

En cuanto al desarrollo de sistemas informáticos de orientación militar cabe señalar el sistema británico ADCIS (Air Defence Command Information System), destinado a optimizar la utilización de los medios de defensa aérea en Europa.

Ada, EN LAS NUEVAS TECNOLOGIAS

*(La promesa de Ada es real, pero los resultados todavía están por llegar).
Braun, C. L., Director de aplicaciones militares de SofTech.*

Ada se está introduciendo rápidamente en las nuevas tecnologías, tales como la Inteligencia Artificial (I.A.), Ingeniería de las Estaciones de Trabajo y los computadores profesionales. Existen actualmente compañías que trabajan en Inteligencia Artificial, que han desarrollado intérpretes y traductores de Ada utilizables en sus máquinas, ya que se espera que en un plazo no muy lejano el DoD será un consumidor de software destinado a I.A. y aunque hoy en día el lenguaje de programación habitual en I.A. es el Lisp, Ada puede ser un modo de favorecer el hardware de I.A. para aplicaciones no I.A.

La relación precio/prestaciones de las estaciones de trabajo ha disminuido últimamente lo suficiente como para ser una alternativa a las mainframe, por lo cual las firmas que buscan obtener programadores en Ada para el desarrollo del software de alto volumen en los próximos años, tal es el caso del IBM PC AT, que en el futuro incorporará muchas herramientas de ayuda al desarrollo de programas en Ada.

El futuro a corto plazo del Ada a juicio de los expertos no sólo estará centrado en proyectos militares

promovidos y financiados por el DoD, sino también en los laboratorios de uso científico, en el control de procesos industriales, en la industria de transporte aéreo, en la industria del petróleo, en medicina y en educación. Todas estas áreas necesitan elevada seguridad, eficiencia en su proceso en tiempo real.

En el campo de la educación se está produciendo un incremento de la demanda de productos en Ada. Los compiladores de Ada han empezado a disminuir y esto ha hecho que muchas instituciones educativas puedan acceder al uso del lenguaje.

Por todo esto se espera que Ada reporte beneficios económicos y tecnológicos, siendo actualmente el lenguaje preferido en muchas universidades americanas y europeas.

SIMULADORES DE VUELO EN Ada

(Con Ada la productividad en el software de los simuladores aéreos se multiplica por cinco.)

La simulación aérea mediante el concurso del ordenador es una técnica que permite simular las condiciones reales de vuelo tanto propias como las de los aviones enemigos y las maniobras que ellos efectúan. Para ello, el ordenador presenta en un terminal con pantalla una imagen del avión, su entorno y la mayor parte de las órdenes destinadas al avión, con el fin de reducir los riesgos de confusión que entrañan las comunicaciones fónicas. Este problema está siendo estudiado en los Estados Unidos, Francia y Gran Bretaña, analizando la palabra y reconociendo la voz, pero tiene el inconveniente de requerir gran cantidad de memoria de los microprocesadores, y no se ha generalizado todavía el empleo de este tipo de sistemas.



▲ Figura 1

Mediante este sistema el coste del adiestramiento de los pilotos se reduce espectacularmente por el ahorro de combustible, pasando de unos 5.000 dólares por hora en un ejercicio real con un F-5, a 1.300 dólares la hora en un simulador. Por consiguiente, una hora de adiestramiento en combate utilizando dos aviones reales cuesta unos 10.000 dólares, con el consiguiente ahorro de combustible y la eliminación de posibles accidentes aéreos.

Por otra parte pueden efectuarse unas cinco operaciones de combate por hora, mientras que el simulador permite representar unas veinte, e incluso los factores atmosféricos no influyen en su operacionalidad, con lo cual queda patente la seguridad y el ahorro que supone la utilización de este tipo de simulador.

En marzo de 1985 se utilizó un compilador de la empresa californiana Irvine Computer Science Co. en el simulador F4J. Este proyecto fue el primero en el mundo que tuviese todo un simulador en Ada, y que funcionara con éxito.

Ada OPERACIONAL EN VUELOS CRITICOS

En octubre de 1984 se probó por primera vez la respuesta de un sistema de control con programas escritos en Ada. Dicho software estaba soportado por un computador digital a bordo de un F-15 de McDonnell Douglas, con destino a las FF.AA. de los EE.UU. (figura 1),

La prueba consistió en un vuelo con restricciones de tiempo en el aire, baja altitud y velocidad inferior a 1,2 Mach.

Los programas de control de vuelo están almacenados en una memoria sólo lectura (ROM). El

computador tiene una memoria principal de 27 Kb, pero en el test inicial solamente se utilizaron 5 Kb.

Los tests fueron inicialmente escritos en Pascal, cuyo código objeto fue comparado con el de Ada, resultando equivalentes en cuanto a velocidad. Uno de los tests de respuesta que sufrió el computador de a bordo consistió en generar código de un comando en bucle cuyo tiempo crítico era de 12,5 ms, superándolo en un tiempo de 8 ms. Se trata de un compilador rápido, pues permite al procesador hacer 80 chequeos por segundo de los controles de vuelo.

Este compilador de Ada está estructurado en dos partes, el primer programa (Front End) analiza el código fuente y genera una representación intermedia, en la segunda pasada traduce el código intermedio en código máquina para cada procesador particular. El compilador de Ada Irvine ha sido reescrito para 8 procesadores, cuatro Z8002, de Zilos, dos para el control de profundidad y dos para las rotaciones.

También se ha utilizado este compilador en los sistemas de vuelo del F-20 de Northrup Co.

Los diseñadores del compilador admiten que su esfuerzo ha sido dirigido en la obtención de un producto rápido, aunque no cumpla las normas de validación de Ada Joint Office.

En efecto, los expertos admiten que se trata del compilador más pequeño y rápido del mercado. Se ha comprobado que es tres veces más rápido que un compilador convencional de Fortran, y de dos a tres veces más rápido que un compilador típico de lenguaje C.

Comparándolo con el líder de los compiladores validados por el Pentágono, el VAX Ada de DEC, éste tiene una velocidad del 10 por 100 de un compilador estándar de Fortran.

OTRAS APLICACIONES MILITARES EN Ada

Todavía es temprana la aplicación masiva del lenguaje Ada en sistemas informáticos, sin embargo, con el patrocinio del DoD, se está difundiendo rápidamente. Para ello ordenó que todos los sistemas defensivos de misiones críticas planeados con posterioridad al 1 de julio de 1984 se implementen en Ada.

Actualmente el Ejército, la Armada y las Fuerzas Aéreas de los EE.UU. tienen en marcha 34 proyectos claves de gran envergadura soportados por Ada. Destacan sobre todo el Programa del Sistema de Información (WIS) del WWMCCS (Worldwide Military Command Control Systems), el AFS-ICCS (Instrumentation Configuration Control Systems), el Joint Tactical Missile Systems y el Submarine Advanced Combat Systems.

Conversión WIS-UNITREP

El proyecto UNITREPorting (UNITREP), es un subsistema del Worldwide Military Command And Control System (WWMCCS). UNITREP tiene por misión el elaborar informes para la Junta de Jefes de E. M. de EE.UU., en función de los datos suministrados por los Comandantes en Jefe de las nueve regiones militares.

La Base de Datos de UNITREP incluye datos de descripción y caracterización de todas las unidades de las Fuerzas Armadas de USA y de algunas fuerzas extranjeras bajo control de los Estados Unidos; también recibe datos de localización, estado y capacidades de las unidades, número de personas y disponibilidad para el combate; datos sobre equipamiento y armas, y el estado de los planes de operación.

UNITREP es un software de comando y control y consiste en un conjunto de módulos, escritos en su mayor parte en COBOL y el resto en rutinas en ensamblador. Los módulos se utilizan para aceptar y validar mensajes, puesta al día de la Base de Datos, responder a preguntas y para generar informes. La base de datos de UNITREP es secuencial indexada y tiene aproximadamente 22 tipos de registros y 500.000 registros pertenecientes a 55.000 unidades. Los programas fuentes sobrepasan las 500.000 líneas de las cuales alrededor de 300.000 son de COBOL dedicadas a la edición de mensajes, puesta al día del archivo y mantenimiento de las tablas internas de los códigos de UNITREP.

La aplicación en Ada consiste en convertir las líneas de COBOL y el sistema de manejo del archivo convencional en Ada y un sistema de manejo de una base de datos relacional. El proyecto fue desarrollado por Naval Ocean Systems Center (NOSC) y SAI Comsystems, de San Diego (California). UNITREP se escogió por ser una aplicación estándar del sistema de información WWMCCS y porque sus comandos no son de gran tamaño y sus algoritmos son de baja complejidad. También porque su integración o dependencia de otros subsistemas WWMCCS es baja.

El prototipo UNITREP/Ada se contruyó en cuatro etapas:

1. Las definiciones de los registros de las bases de datos de UNITREP fueron convertidos en relaciones de un sistema de base de datos relacional.
2. Estas definiciones fueron codificadas usando tipos de registros Ada.
3. Un modelo sencillo de procesamiento de mensajes se escribió y probó sobre un Intel 432, con multiproceso y memoria virtual, siendo el conjunto de instrucciones soportadas por Ada.
4. Implementación de UNITREP en Ada.

Según los autores del proyecto, quedó demostrado que el lenguaje Ada es muy útil en software de comando y control, resaltando su potente definición de tipos y el buen tratamiento de las excepciones.

Dicho proyecto se halla operativo desde abril de 1985.



▲ Figura 2

AFS-ICCS

Otra de las principales aplicaciones en Ada es el desarrollo de software del Instrumentation Configuration Control Systems (ICCS) para el Centro de Desarrollo de Motores en estación de la Fuerza Aérea de Arnold (USA).

El proyecto ICCS tenía como objetivo fundamental el desarrollo de software que soportara las células de pruebas de motores a reacción y de misiles. Dichos programas tendrían un ciclo de vida de 15 a 25 años y sus especificaciones (bajo el influjo de la filosofía FORTRAN) definían el manejo de un amplio rango de instrumentación y sistema de adquisición de datos. Cada célula de prueba tendría su propia Base de Datos, se preveía la construcción de controladores de interface para los diferentes ordenadores que adquiriesen datos. Finalmente las especificaciones fueron estructuradas bajo MIL-STD-483, e incluidas en la carta NASSI-Sniderman.

El proyecto fue encomendado a Digital Electronic Systems (DES), la cual decidió utilizar el lenguaje Ada, como lenguaje de desarrollo. Se formaron equipos de programadores especialistas en FORTRAN, PASCAL y COBOL, que comenzaron a estudiar la sintaxis de Ada. Esto se hizo en tres partes: primero, una etapa de conocimiento y familiaridad con el nuevo lenguaje, después se pasó al desarrollo de software según las especificaciones ICCS y finalmente la implementación del mismo.

El proyecto ICCS se esperaba que requiriese 80 hombres/mes, incluyendo 10 hombres/mes para solucionar los posibles problemas del compilador. Se producirían 150.000 líneas de código y varias miles de líneas adicionales de código de prueba.

Las lecciones que se obtuvieron en la realización del proyecto fueron las siguientes: Los comentarios son innecesarios, los identificadores largos son muy descriptivos, los procedimientos y funciones pequeñas no necesitan tests, están mejor preparados para trabajar con Ada los especialistas en PASCAL, las comunicaciones entre los programadores se debe hacer utilizando la terminología del Manual de Ada.

También en el área de la normalización se han hecho esfuerzos utilizando el lenguaje Ada. Los circuitos multiplex de distribución de datos avioelectrónicos (MIL-STD-1553B) y la normalización de los conjuntos de programas destinados a los equipos electrónicos y a la dirección de tiro de las armas aéreas (MIL-STD-1750A), utilizan Ada con el terminal TCT (Tactical Computer Terminal), que pronto será puesta en servicio en el Ejército de los EE.UU. para formar parte del Manoeuvre Control System, capaz de comunicar con el sistema alemán HEROS. (El sistema C3I HEROS está destinado a facilitar las tareas de los estados mayores alemanes del Ejército, la Marina y la Aviación, así como los de la OTAN.)

En el Reino Unido, la sociedad Systems Designers está desarrollando nuevas herramientas de programación, tales como el Analyst, el cual contribuye a formular los deseos de los usuarios de manera aprovechable. El Analyst es un puesto de operador en el que se aplican técnicas de I.A. Permitirá definir especificaciones generales, incluso para proyectos complejos que comprendan numerosos subsistemas, contribuyendo a reducir los efectos del cambio de un elemento del proyecto.

La versión actual del Analyst está operativa en Pascal, y la misma sociedad está desarrollando una versión en Ada del mismo sistema para su centro de programación Prospective.

CESELSA & Ejército del Aire

En nuestro país la empresa Ceselsa es la que ocupa un lugar puntero en el desarrollo de proyectos en lenguaje Ada. Los simuladores de los aviones Casa 101, Harrier y el F-18 (fig. 2) fueron diseñados en un principio en Ada, aunque posteriormente el lenguaje de programación final fue el Pascal.

En la actualidad dicha compañía, en colaboración con el Ejército del Aire, está desarrollando los dos grandes proyectos en Ada: ALERCAN, SAFTA, y diversos subproyectos para la Agencia Espacial Europea (ESA). ALERCAN es un proyecto que consiste en la defensa aérea de Canarias que pasará a integrarse en la futura Ala de Alerta y Control número 2 del Ejército del Aire. SAFTA engloba un conjunto de Sistemas Automáticos de Control de Tráfico Aéreo para Aviación Civil, con destino a los aeropuertos españoles de Sevilla y Barcelona entre otros. Por lo que respecta a los proyectos de la Agencia Europea del Espacio, existen varios aún sin adjudicar y otros en fase de desarrollo, los cuales tratan de evaluar el lenguaje Ada.

Los recursos físicos para poder desarrollar dicho Software son diversos, destacando de entre ellos el ordenador MV 10000 de Data General y entre los recursos lógicos se dispone del compilador Ada de Rolm/D.G.

Ada y Sanidad

En los últimos cinco años se ha asistido a una proliferación desordenada y anárquica de los sistemas informáticos en la Sanidad española. La falta de un Plan Informático Sanitario Nacional, la escasa formación informática del usuario final y la agresiva política comercial de los fabricantes del sector, están llevando a nuestra Organización Sanitaria Nacional y a nuestros Hospitales a un grave confusiónismo en los recursos físicos y lógicos. Sin embargo, este fenómeno de heterogeneidad de equipos y lenguajes de programación afecta también a otros países, incluyendo a los propios EE.UU. Baste como muestra el toque de alarma recientemente expresado por J. Hennon (6). "En todo el Ejército Estadounidense se encuentran calculadoras. Nadie conoce el número de las que están en servicio... Los equipos suministrados solamente por Honeywell tienen un costo superior a los 2.600 millones de dólares, y no se trata del proveedor más importante".

En la Sanidad Militar y en Medicina Aeroespacial se impone en aras de una óptima relación costo/eficacia la implementación de equipos homogéneos y con un lenguaje de programación común que los haga interoperativos. Solamente con el diseño de un Plan Informático Sanitario de Defensa será posible alcanzar estos objetivos. El Ministerio de Defensa Español ha detectado ya esta problemática y tiene puesto en marcha un ambicioso plan de Informática Sanitaria Integral en el que se contempla la informatización interna de cada Unidad Sanitaria y su interacción e interrelación de los Hospitales Militares a través de una Red de Transmisión de Datos.

El Ada supone una herramienta de inestimable valor para superar los obstáculos de diversificación del Software. Sin embargo, hasta la fecha no se conoce ninguna realización ni proyecto en curso en el mundo occidental de aplicaciones médico-aeronáuticas sobre Ada.

En el dossier "Informática Médica en el Hospital del Aire" (7), se trazaron las directrices sobre las que podría discurrir la informatización interna de un Hospital Militar. Esta doctrina, al cabo de dos años, sigue teniendo vigencia. La incorporación del lenguaje Ada a dicho plan supondría el conseguir los objetivos propuestos en un plazo más corto, así como la posibilidad de incorporarse al mercado mundial de componentes lógicos sanitarios que está por llegar.

Es importante hacer mención en este proceso de informatización a gran escala al Centro de Instrucción de Medicina Aeroespacial (CIMA), con misiones altamente específicas actuales y en un futuro inmediato. ■

BIBLIOGRAFIA

(1) A.W. Burks, "From ENIAC to the Stored-Program Computer". A history of computing in the twentieth century. Academy Press, N.Y., 1980.

(2) B. Knobe, "Flight Languages: Ada vs. Hal/s". Guidance and Control. Vol 4. No. 1, 1981, pp. 35-40.

(3) D. E. Kunth, T. Pardo, "The early development of

programming language Stan-CS-76-562", Stanford University Computer Science Department, 1976.

(4) P. Wallich, "Software 85", IEEE Spectrum. January 1985, pp. 50-52.

(5) D. Klein, "A buyer's guide to Ada procurement", Defense Electronic, January, 1985, pp. 101-109.