

Iniciativa de Defensa Estratégica (S.D.I.)

Introducción.
¿Qué es la iniciativa de Defensa Estratégica?
Planteamiento básico
Diseño inicial
Innovación tecnológica en la S.D.I.
Problemas de integración
Niveles de complejidad en la S.D.I.
Software, la piedra angular del proyecto
Especificación
Arquitectura
Desarrollo
Potencia de computación
Pruebas
Tolerancia a fallos
Investigación software
D.L. Parnas y el pesimismo tecnológico
F.P. Brooks, optimismo tecnológico y otras opiniones
Factores de tercer nivel
Conclusiones

Este texto ha sido redactado por Gustavo Alonso en 1990 para el curso de Cibernética y Teoría de Sistemas de la Escuela Técnica superior de Ingenieros de Telecomunicación de Madrid. Se basa en su mayor parte en artículos del IEEE; para evitar las referencias continuas a lo largo del texto, incluimos aquí los números utilizados para desarrollar este tema:

IEEE Spectrum, Septiembre 1985, número monográfico sobre la S.D.I.

IEEE Computer, Noviembre 1986, "Can S.D.I. Software be error free?", Ware Myers, pp. 61-67.

IEEE Spectrum, Noviembre 1988, número monográfico dedicado al coste de los proyectos de defensa.

IEEE Spectrum, Marzo 1989, número monográfico dedicado a la repercusión en la tecnología de los tratados de paz y de desarme.

IEEE Spectrum, Noviembre 1989, "S.D.I. in transition", John A. Adam, pp. 32-38.

1. Introducción

El presente capítulo pretende ser un ejemplo donde aplicar algunas de las ideas que se han visto a lo largo de todos estas Notas. No se pretende hacer ningún juicio de valor sobre el tema si no es desde un punto de vista estrictamente tecnológico y desde la complejidad. Evidentemente la Iniciativa de Defensa Estratégica (S.D.I.) es un tema controvertido, quizá ahora no tanto como lo fue en su momento pero, en cualquier caso, ejemplifica magníficamente varias de las ideas que hemos tratado.

El proyecto de la S.D.I. es de una complejidad formidable y los objetivos que persigue en un amplio abanico de tecnologías son muy ambiciosos. Aún está por ver que pueda manejarse toda esa complejidad. Como veremos, al tratar los problemas que plantea este proyecto nos iremos encontrando con algunos de los problemas que ya hemos estudiado referentes al software y al hardware, además de muchos otros, y con muchas de las propuestas conceptuales que se hacían en la primera parte (niveles, jerarquías, sistemas, incertidumbre, etc.). Lo que tiene de atractivo la S.D.I., desde el punto de vista de nuestro estudio, es que intenta abordar todos esos problemas en un único proyecto, siendo un buen campo teórico donde ejercitar las ideas de complejidad.

Una forma de ver todo el proyecto es a través del modelo de tres niveles de complejidad de Sáez Vacas (referencia a Marcos Conceptuales) y es la que hemos escogido para desarrollar brevemente aquí. Pero no es, en absoluto, la única. Instamos al lector a que intente ver todas las propuestas de la S.D.I. a la luz de diversos modelos y definiciones de complejidad y comparar los resultados que vaya obteniendo con lo visto hasta ahora referente a tecnología. Algunas ideas pueden ser la complejidad relacionada con la incertidumbre (principios de Klir), la simplificación, jerarquías de niveles, definición amplia de complejidad, etc.

2. ¿Qué es la Iniciativa de Defensa Estratégica?

En este primer apartado vamos a tratar de explicar los supuestos de partida de la S.D.I., qué pretende conseguir y cómo. De todo ello, lo que más nos interesa es cómo este proyecto afecta a la innovación tecnológica al plantear una serie de cuestiones críticas, desde el punto de vista puramente tecnológico, que sin duda afectan al desarrollo y a la innovación. Más adelante profundizaremos en la problemática relacionada con el software, aquí pretendemos destacar la gran demanda de tecnología en todos los campos que supone un proyecto como éste.

2.1 Planteamiento básico

El proyecto de la S.D.I. (Strategic Defense Initiative) tiene su origen más inmediato en un discurso de Ronald Reagan, pronunciado el 23 de Marzo de 1983, en el que proponía convertir las armas nucleares en algo "impotente y obsoleto" a través de la construcción de un sistema de defensa capaz de inutilizar misiles nucleares cuando se lanzan contra un objetivo.

Desde entonces, la S.D.I. ha levantado apasionadas controversias tanto en el campo político como en el tecnológico y las interpretaciones de los objetivos que se persiguen son de lo más variado. Quizá la más conocida es la de la "Guerra de las Galaxias" que tiene su origen en uno de los objetivos concretos que persigue la S.D.I., como es la instalación en el espacio de bases antimisil

destinadas a derribar en vuelo las armas nucleares lanzadas contra los EEUU. También se conoce a la S.D.I. como "escudo", centrándose en la idea de que de lo que se trata es de crear una barrera capaz de detectar y detener ataques nucleares. Ambos términos son demasiado simplistas y llevan a confusión, uno por considerar un único aspecto del problema, aunque sea ciertamente espectacular, y otro porque parte de un presupuesto erróneo, ya que la S.D.I. no es perfecta y la cobertura no será completa.

La interpretación que hace el Pentágono también es interesante, en lugar de la destrucción mutua asegurada (Mutually Assured Destruction, MAD) se pretende llegar a una situación de defensa mutua asegurada.

En pocas palabras, la S.D.I. pretende desarrollar un sistema de defensa, que cubra todo el territorio de los EEUU, encargado de detectar y eliminar las armas nucleares lanzadas contra este territorio.

2.2 Diseño inicial de la S.D.I.

La S.D.I. parte de una aproximación por capas al problema de la defensa. Cada una de estas capas está relacionada con alguna de las diferentes fases del lanzamiento de un misil (lanzamiento, postlanzamiento, curso intermedio y fase terminal). El éxito depende, en gran medida, de la capacidad para destruir el mayor número de misiles que sea posible en la fase de lanzamiento (desde que despegan hasta que se dejan libres en la atmósfera). Los misiles que atraviesen este primer sistema de defensa han de ser destruidos en las fases posteriores intentando siempre reducir al mínimo el número de cabezas nucleares que entran en la atmósfera.

Cada una de estas fases y cada tipo de misil requiere un armamento diferente y, a medida que aumenta la complejidad del problema, más exótico. En la fase de lanzamiento hay que distinguir dos tipos de cohetes, los de lanzamiento rápido (que en lanzamientos intercontinentales tardan 1 minuto en superar esta fase) y los de lanzamiento lento (que en misiles intercontinentales supone unos 3 minutos en esta fase). Para derribar los cohetes de lanzamiento lento hay que disponer de bases espaciales de misiles nucleares, para los de lanzamiento rápido se intentan desarrollar láseres disparados también desde bases espaciales. Este mismo tipo de armas, siempre desde bases en el espacio, junto con haces de partículas son las que se utilizan para atacar la segunda fase, la de postlanzamiento.

En esta segunda fase, los cohetes se separan en varias cabezas nucleares y en diversos señuelos, lo cual añade más complejidad al problema pues hay que distinguir entre los objetivos reales (cabezas nucleares) y los señuelos. En la tercera fase, durante el vuelo fuera de la atmósfera que dura de 6 a 22 minutos, se utilizan pequeños proyectiles lanzados desde tierra junto con haces de partículas. En la fase terminal se emplean proyectiles de alta velocidad (armas de energía cinética) y cohetes para destruir las cabezas nucleares restantes.

Al contrario de lo que puede parecer, el desarrollo de este tipo de armas no nucleares es el campo más maduro de la S.D.I. y muchas de las tecnologías no son nuevas (excepto los láseres de rayos X). Además aparecen consideraciones nuevas como que es más barato desarrollar estas armas de defensa para destruir misiles que hacer nuevos misiles de ataque.

La utilización de este tipo de armas exige avances tecnológicos muy importantes, pues algunas de ellas han de estar en bases espaciales y las que se disparan desde tierra han de poder guiarse hasta su objetivo. Las propias armas son en sí tecnología punta: tanto las armas de energía directa (láseres y haces de partículas) como las armas de energía cinética (cohetes y rampas de lanzamiento).

Efectividad de la S.D.I.

Un posible planteamiento estratégico del sistema a construir puede ser el siguiente:

Ataque

*1.400 Misiles Balísticos Intercontinentales, ICBM
x 10 cabezas nucleares/misil = 14.000 cabezas
x 10 señuelos/cabeza nuclear = 140.000 señuelos*

Defensa

** Fase de lanzamiento*

5 satélites de vigilancia, 160 estaciones láser de batalla y radar, con una efectividad del 90 %

14.000 x 0,1 = 1.400 cabezas nucleares

140.000 x 0,1 = 14.000 señuelos

** Fase de curso intermedio*

28.000 armas de energía cinética, 20 satélites de vigilancia, con una efectividad del 90 %

1.400 x 0,1 = 140 cabezas nucleares

14.000 x 0,1 = 1.400 señuelos

** Fase terminal*

75 radares, 25 aviones de vigilancia, 140.000 cohetes de energía cinética, con una efectividad del 90 %

14 cabezas nucleares alcanzan su objetivo

2.3. Innovación Tecnológica en la S.D.I.

La Iniciativa de Defensa Estratégica es un enorme reto a la ingeniería y la ciencia. Los problemas que plantea un escenario como el descrito en el apartado anterior son muy complejos y comprenden un abanico muy amplio de tecnologías que han de trabajar de forma integrada. Muestra de ello son los ocho programas en los que se subdividió la S.D.I. a finales de 1983: sistemas; sensores; vigilancia, ataque y tecnologías clave; armas de energía cinética; armas de energía directa; innovación científica y tecnológica; y asuntos externos.

El proyecto de la S.D.I supone una fuerte inversión en investigación y desarrollo mantenida durante bastantes años. Dada la creciente dependencia que tiene la investigación estadounidense del presupuesto de defensa, no son pocos los que defienden que además del problema de la fiabilidad de la S.D.I., han de considerarse todos los avances que se pueden lograr con estas inversiones. Orquestando los mecanismos necesarios de transferencia tecnológica la sociedad puede beneficiarse de los resultados de estas investigaciones.

Esta parece la interpretación de algunos países europeos y gran parte de las compañías implicadas. Un ejemplo de este interés por los proyectos de investigación generados es el de Dinamarca, cuyo parlamento se opone rotundamente a la militarización del espacio pero al mismo tiempo ha dejado bien claro que esto es sólo una resolución, no una ley, y que por tanto las compañías danesas participarán en el proyecto. Esta claro que existe un riesgo enorme de "perder el tren" de la tecnología si no se participa en la investigación y desarrollo de estos sistemas.

El problema con que se enfrenta está lejos de ser sencillo y se han señalado cuatro como los obstáculos principales con los que se va a encontrar:

- 1.- Distinguir las cabezas nucleares de los señuelos (según algunos cálculos un ataque consistiría en unos 3.000 misiles con un total de unas 30.000 cabezas nucleares y 250.000 señuelos).
- 2.- Diseñar software fiable para gestionar las operaciones (se tendrán que tomar cientos de miles de decisiones independientes y sincronizarlas en cuestión de milisegundos).
- 3.- El sistema ha de ser suficientemente robusto como para poder soportar un ataque contra él y seguir funcionando.
- 4.- Ha de ser un sistema suficientemente barato como para que el enemigo no se sienta tentado a superarlo aumentando su capacidad ofensiva.

El primer punto obligará a mejoras importantes en los sensores, radares y mecanismos de detección pues son una parte crucial del sistema. Muchas de las armas empleadas en la S.D.I. requieren una gran cantidad de energía y no se puede desperdiciar disparando contra objetos que no son una amenaza, por otro lado, los sensores que descartan objetivos identificándolos como señuelos se enfrentan a la posibilidad de que haya cabezas nucleares que intentan pasar por señuelos, con lo que la labor de identificación es aún más compleja. A esto se añade el problema de que hay que utilizar la energía exacta, no se puede disparar con menos energía porque no se puede creer que se ha eliminado algo que sigue siendo una amenaza y tampoco se puede derrochar energía aumentando las dosis cuando esto no es necesario.

En 1984 se hizo un experimento en el que se informó que se había conseguido detectar, en el espacio, el calor de un cuerpo humano a 1.600 km. de distancia. Un proyecto como el de la S.D.I. exige poder detectar objetivos entre 20 y 50 grados Kelvin más fríos que la silueta de un cuerpo humano contra el calor de la superficie de la tierra a distancias superiores a los 5.000 km.

Otro punto muy interesante es el número de estaciones necesarias para frenar un ataque. Sobre esto no hay acuerdo pues depende de las suposiciones de partida que se hagan y consideraciones técnicas muy complejas (potencia de los láseres, difusión de éstos, número de misiles enemigos, altitud de las bases, etc.), pero las simulaciones realizadas para un ataque de 1.400 misiles van desde 30 estaciones a 2.400 y proponen desde un crecimiento lineal de las bases con el número de misiles ofensivos a un crecimiento del 40% cuando el ataque se dobla. Además, estas simulaciones han aportado otros datos interesantes como la velocidad de eliminación de misiles en la fase de lanzamiento (10 por segundo) o el tiempo que hay enfocar un misil con un láser para destruirlo (desde 1,7 a 6,6 segundos, dependiendo de la distancia).

Esto nos lleva a la situación actual de los láseres, los haces de partículas y la capacidad de potencia en las bases espaciales. Con datos de 1985 esta tabla resume muy bien las diferencias entre lo que se tiene y lo que se persigue:

	Objetivos	Actualmente
Brillo del láser	10^{21} vatios/esteroradian	10^{17} W/sr
Energía de partícula	250 megaelectronvoltios	50 MeV
Potencia en el espacio	100-300 megawatios	25 kW
Eficiencia	600 Kilojulios por Kg.	-
Coste	1 dólar por Kilojulio	50 dólares/Kj

Las armas de las que antes hablábamos requieren unas condiciones muy especiales. Los láseres y haces de partículas necesitan mucha energía para ser efectivos y esta energía se la ha de proporcionar la base espacial en la que se instalen. Los láseres, por ejemplo, necesitan unos 100 MW, las rampas de lanzamiento de proyectiles de alta velocidad necesitan pulsos de energía del orden de los gigawatios. Esto significa que hay que mandar mucha masa al espacio y se traduce en un coste muy elevado. Por ello, otro de los objetivos de la S.D.I. es reducir el coste de acceder al espacio, en 1985 este coste era de 1.200 dólares cada libra de peso y se quiere reducir a 10 dólares por libra de peso.

	Objetivos	Actualmente
Potencia básica	0,1-0,2 MW durante 7 años	12 kW durante 4-5 años
Potencia en alertas	2.0-20 MW tiempo indefinido	12 kW durante 4-5 años
Potencia en combate	100-300 MW durante 30 minutos	Ninguna
Condiciones de potencia	100 kV, 1000-3000 A	300 V, 10 A
Eficiencia	3 kW/kg	0,005 kW/kg

Las armas de energía cinética plantean problemas muy interesantes de resistencia de materiales. Los proyectiles se han de disparar a una velocidad superior a los 20 km/s, a esa velocidad las "balas" llegan a fundir el cañón que las dispara, lo que obliga a buscar materiales especiales que soporten esa presión y temperatura. En un experimento en el que se consiguió una velocidad de 11 km/s con un proyectil de 3 gramos, la presión ejercida hizo que el cañón se deshiciera a medida que la bala pasaba por él, el proyectil salió gracias a que su velocidad era tan alta que se mantenía por delante de la onda de choque que destruyó el cañón. Además, el tipo de proyectiles que se quieren conseguir, de unos 3 kg., lanzados a una velocidad de entre 15 y 30 km/s, requieren un pulso de potencia de varios gigawatios, potencia que ha de ser proporcionada por una fuente suficientemente pequeña como para poder ser instalada en el espacio.

Todos estos ejemplos sólo son una muestra de la gran cantidad de avances tecnológicos de importancia que exige desarrollar un sistema como el que propone la S.D.I., y no sólo en armas o en sensores, sino también investigación en fuentes de potencia (química, nuclear, solar, por radioisótopos, etc.), circuitos eléctricos y electrónicos (capaces de soportar esa potencia en un medio extremadamente hostil y con unas limitaciones grandes), aviónica (que en el caso de los proyectiles de energía cinética tendrán que soportar aceleraciones de 100.000 G), ordenadores ópticos (para conseguir la velocidad de proceso adecuada), comunicaciones en el espacio, física de estado sólido (para los sensores y muchos otros componentes del sistema), etc.

2.4. Problemas de integración

Dejando un poco de lado los problemas puntuales que hemos ido viendo hasta ahora, conviene pararse a reflexionar sobre la naturaleza de un sistema de defensa como el que nos ocupa. Aún siendo realmente formidables, los problemas que plantean las armas o ciertas tecnologías concretas son de orden menor comparados con el problema de conseguir que el sistema funcione como un todo. Un sistema de defensa ha de trabajar con un mínimo de coordinación y más aún en este caso en el que la tarea de defensa se divide en varias fases. Se trata de construir un sistema, no una serie de dispositivos de defensa independientes.

En primer lugar han de coordinarse las diferentes fases de la defensa. Los sensores que han detectado los lanzamientos en su primera fase pasan la información a las armas que actúan en esta fase, éstas disparan y se han de recoger datos sobre los impactos conseguidos, qué objetivos han resultado intactos, estimaciones de las rutas, datos sobre objetivos reales y señuelos, etc., si hay tiempo se volverán a reorientar las armas y se volverá a disparar, si no, hay que pasar todos estos datos al control de la segunda fase e informar de qué armas se han dejado disponibles para que se puedan utilizar en fases posteriores y así sucesivamente en cada fase. Y todo esto de forma distribuida porque cada sensor puede actualizar la base de datos de forma dinámica y al mismo tiempo, esa misma base de datos se está consultando para localizar el próximo objetivo. A esto se añade que la base de datos ha de ser redundante y robusta pues daños en un nodo no pueden causar pérdidas de datos.

Algo similar sucede con los sistemas de comunicación, que deben ser capaces de operar en un entorno extremadamente hostil y con unos niveles de radiación y ruido muy elevados. También ha de poder reconfigurarse para asegurar las comunicaciones aunque se produzcan daños en los nodos y poder hacerlo probablemente más de cien veces en pocos minutos. Durante las fases previas a un ataque esto no es tan complejo, en una situación de alarma se puede conseguir que se mantenga un flujo de información adecuado pero en el momento en que empiece el ataque los propios cohetes de lanzamiento destruidos servirán para introducir ruido en el sistema (sobre todo en los sensores) y las cabezas nucleares detonadas en el espacio producirán radiaciones y pulsos electromagnéticos suficientes como para cegar permanentemente a los sensores y cortar las comunicaciones, y eso sin contar con los efectos puramente físicos de una explosión nuclear.

En este entorno se ha de mantener un sistema que funcione coordinadamente tal y como explicábamos antes. Para muchos, la gestión del sistema es la clave de todo el problema de la S.D.I., decidir qué hacer con la información una vez que se ha conseguido. Entre otras cosas se ha de mantener una base de datos con todos los objetivos que existan y la información relevante en cada caso (si es una cabeza nuclear, si es un señuelo, ruta, si ya ha sido desactivado, si está intacto, etc.) ya que no puede repetirse toda la operación de identificación y seguimiento en cada fase y es imposible saber si un misil ha sido ya inutilizado por un láser o si está intacto. También hay que mantener información sobre todos los sistemas de defensa, su situación y estado (cuánta munición queda, qué sensores, computadores y nodos están todavía en funcionamiento, etc.). Coordinar la asignación de armas a los objetivos y mantener la consistencia de la base de datos en todo momento.

El problema de la consistencia es fundamental para conseguir una defensa eficaz pues si una estación con armas resulta destruida sus objetivos han de asignarse a otra estación capaz de alcanzarlos y con suficiente munición como para hacerse cargo de la nueva tarea. También habrá de resolver problemas de datos contradictorios, por ejemplo, un sensor dice que hay 8 misiles y otro dice que hay 12, ¿quién tiene razón?, ¿no es posible que haya 20?, en un sistema de este tipo estas decisiones son cruciales. Y todos estos problemas hay que combinarlos con el hecho de que la base

está distribuida y las diferentes copias han de responder de la misma forma pues no puede permitirse que cada copia dé una solución distinta. Por el momento no existen técnicas que aseguren la fiabilidad de sistemas que utilizan bases de datos asíncronas.

Uno de los programas que más frecuentemente se citan como ejemplo de que se puede construir software muy complejo es el que utiliza AT&T para la conmutación en líneas telefónicas (conocido como el Sistema Número 5 de Conmutación Electrónica de AT&T). Este sistema software es del orden del millón de líneas de código, distribuido, en tiempo real, y tolerante a fallos.

Sin embargo este sistema se colapsó en Enero de 1990 durante 9 horas, produciendo unas pérdidas a AT&T que están entre los 60 y 75 millones de dólares. Eso sin contar con las pérdidas producidas en todas las compañías que se basan en el teléfono para su funcionamiento como, por ejemplo, las compañías de reservas de vuelos que estimaron catastróficas sus pérdidas. Este suceso ha demostrado una vez más cómo un fallo en un pequeño programa que forma parte de un sistema mucho mayor tiene unas repercusiones gravísimas en todos los niveles superiores. Lo que es en sí mismo un fallo de un elemento del primer nivel de complejidad, se traduce en un colapso total del sistema (segundo nivel) y una implicaciones importantísimas en el tercer nivel.

El fallo se produjo en un ordenador de Nueva York que "creyó" estar sobrecargado y empezó a rechazar llamadas, otros ordenadores se conectaron automáticamente para hacerse cargo del exceso de demanda, pero empezaron a comportarse como el primer ordenador, creyendo estar sobrecargados y rechazando llamadas. Se formó así una cadena de ordenadores que se iban conectando a los estropeados y que, a su vez, se estropeaban. El fallo parece que se produjo debido a una mejora introducida en el sistema que permitía reducir el tiempo entre el marcado y la señal de conexión, esta mejora introdujo un error en el sistema que se propagó a todos los ordenadores de la red ya que todos tenían instalado básicamente el mismo software.

Evidentemente esta es una prueba más de la vulnerabilidad de los grandes sistemas software aunque también habría que estudiar durante cuanto tiempo ha estado funcionando el sistema a pleno rendimiento hasta que ha fallado y tener en cuenta que el error se produjo al intentar cambiar el programa sin respetar las especificaciones iniciales. [Elmer-Dewitt, en la revista Time, número de Enero de 1990]

El problema de la integración de todos los elementos que componen la S.D.I. ha sido visto frecuentemente como su punto más débil y donde se requieren los avances tecnológicos más importantes para conseguir que el sistema funcione. Muchos de los problemas que se plantean son totalmente nuevos para la tecnología e incluso todavía no se comprenden bien.

2.5 Niveles de complejidad en la S.D.I.

La Complejidad del proyecto de la S.D.I. se puede reinterpretar desde la perspectiva del modelo de tres niveles de complejidad [Sáez Vacas, 1983, referencia a Marcos Conceptuales], y quizá sea una de las pocas formas de verlo con la que se pueda abarcar toda la realidad del mismo dada la enorme cantidad de factores relevantes.

Primer nivel de complejidad: Es el nivel de los elementos aislados, en la S.D.I. estos elementos son las armas, las estaciones espaciales, las fuentes de energía, los sensores, sistemas de vigilancia, sistemas de puntería, etc. Todos y cada uno de ellos plantean formidables problemas a la tecnología y a la ciencia, pero no conviene olvidar que son sólo el primer nivel de complejidad. Prueba de ello es que la mayor parte de la controversia no está en este nivel donde se asume que las metas que propone la S.D.I. son alcanzables en un plazo razonable de tiempo. Por otro lado, este nivel es el más maduro de la S.D.I., se conocen gran parte de todos los dispositivos, ya se está experimentando con ellos y se dispone de una base teórica bastante amplia y experiencia suficiente como para confiar en que se conseguirán los objetivos establecidos.

Segundo nivel de complejidad: Del que ya hemos comentado algunas ideas cuando hablábamos de los problemas de integración del sistema de defensa. El segundo nivel de complejidad es la complejidad sistémica, que aparece cuando los elementos aislados (armas, sensores, satélites, estaciones de seguimiento terrestres, radares, etc.) se combinan para formar un sistema, un todo completo que ha de funcionar coordinadamente. La complejidad de este nivel es mucho mayor que la de los elementos aislados y es el problema fundamental de la S.D.I. Es en este nivel donde se han producido las críticas más fuertes, argumentando que si bien los elementos de primer nivel son factibles, la coordinación de todos ellos es imposible dado el tamaño, entorno y las condiciones en las que ha de sobrevivir el sistema.

Tercer nivel de complejidad: Que tiene su origen en la interacción entre el sistema tecnológico y la sociedad. En la S.D.I. este nivel de complejidad juega un papel fundamental, por todas las consideraciones políticas, sociales, económicas y éticas que plantea. Gran parte de las críticas a la S.D.I. se hacen también en este nivel, dejando de lado el aspecto puramente técnico, mostrándose así una vez más la importancia del factor humano y social en la tecnología. Como ejemplo de cuestión política están los tratados internacionales que prohíben la construcción de muchas de las armas que propone la S.D.I.; como problema social está el hecho de que el escudo de la S.D.I. no va a ser perfecto y detrás va a haber toda una sociedad que cree estar defendida; como problema económico basta citar el presupuesto inicial del proyecto, 26.000 millones de dólares, y el coste total, estimado entre varios cientos de miles de millones y un billón de dólares.

Cualquiera que sea el tema que se toque referente a la S.D.I. es conveniente no perder de vista este mapa que nos proporciona el modelo de tres niveles. La complejidad no está sólo en la tecnología, sino también en nuestra capacidad para comprender el problema globalmente y disponer de alguna referencia que nos permita un ordenamiento coherente de todas estas ideas.

3. Software: la piedra angular de la S.D.I.

Gran parte de lo que hemos tratado hasta ahora está relacionado con diferentes tecnologías, láser, física de partículas, satélites, balística, radar, sensores infrarrojos, etc. pero todavía no hemos mencionado la más importante, el software. En este apartado vamos a desarrollar alguna de las características propuestas para el sistema de la S.D.I. y podremos comprobar cómo todo se orquesta alrededor del software. La mayor parte de las críticas y las controversias, de carácter técnico, relacionadas con la S.D.I. tienen que ver con el software. A pesar de que el término "ingeniería software" es ampliamente utilizado, hay todavía muchos que opinan que el diseño y desarrollo de software tiene más de artesanía que de ciencia o disciplina de la ingeniería. La falta de unos principios generales básicos, universalmente aceptados y válidos es una de las primeras barreras con

las que se enfrenta todo desarrollo software y más si es de la envergadura del necesario para construir el sistema que propugna la S.D.I.

3.1 Especificación del proyecto

La importancia del software en un proyecto como el de la S.D.I. es fundamental. Generalmente se tiende a prestar más atención a los aspectos espectaculares de la S.D.I. (armas en el espacio, proyectiles de energía cinética, rayos láser, etc.) que a los ordenadores, las comunicaciones y el software que controla todo el sistema. Ya hemos hablado

de los problemas que plantean las armas, su puesta en el espacio, la energía necesaria y de la complejidad de conseguir sensores que cumplan los requisitos exigidos. Pero, a pesar de ser problemas formidables hay uno mayor, cómo controlar todas esas armas, sensores, satélites, estaciones terrenas, etc., para conseguir que el conjunto se comporte como se desea.

Ese control estará a cargo de un sistema software de entre 10 y 30 millones de líneas de código fuente, ejecutado a razón de 100 millones de operaciones por segundo, capaz de sobrevivir al menos durante 10 años de forma autónoma y sin fallos generalizados y con un tiempo medio entre fallos de dos años. Este es el verdadero problema de la S.D.I.

A este enorme reto que se le plantea a la ingeniería software hay que añadir la necesidad de desarrollar herramientas para poder escribir el código, analizadores, simuladores dinámicos, baterías de pruebas, especificaciones para el desarrollo y todos los elementos que requiere un desarrollo software de envergadura.

3.2 Arquitectura

Un grupo de trabajo de la S.D.I. ha señalado que si no se puede construir el sistema de ordenadores y el software para controlar los sensores y las armas, entonces las características y situación de las armas y los sensores es una elucubración puramente académica. Hay que prestar mucha más atención a los problemas de la complejidad del software y a las pruebas a realizar.

De acuerdo con estas ideas, se ha recomendado una arquitectura abierta y distribuida. Las razones para que sea distribuida son varias, la vulnerabilidad de la opción centralizada (si falla el centro coordinador falla todo), un sistema de defensa no necesita estar altamente coordinado (que en un sitio caigan misiles es independiente de que estén cayendo en

otro y las decisiones a tomar son prácticamente las mismas pero independientes) y un sistema distribuido requiere un software no tan complejo (un sistema centralizado ahorra hardware pero

Problema: *Coordinación de un sistema formado por un número muy elevado de elementos y que ha de sobrevivir en un entorno altamente hostil.*

Solución: *Un programa software que controle y coordine el sistema de defensa.*

Inconvenientes: *La enorme complejidad de un programa con varias decenas de millones de líneas.*

Problema: *Construir un sistema poco vulnerable y que reduzca de alguna forma la complejidad del software necesario.*

Solución: *Optar por un sistema distribuido.*

Inconvenientes: *La solución distribuida es menos fiable en cuanto a su comportamiento y aumenta en mucho el problema de las comunicaciones.*

aumenta mucho la complejidad del software para coordinar todos los elementos y se reducen las posibilidades de hacer pruebas, pues éstas habría que hacerlas con todo el sistema). Por otro lado, una arquitectura distribuida necesita que se realicen más disparos, un 20 % más, para destruir la misma cantidad de misiles que un sistema centralizado.

El que el sistema sea abierto es una necesidad, dado que se irán desarrollando diferentes modelos de sensores, armas, comunicaciones, computadores, nuevos nodos, etc. que habrá que ir integrando en el sistema. Un sistema abierto que permita la inserción e integración de elementos inesperados y modificados.

3.3 Desarrollo

En cuanto a la forma de desarrollar el software, se ha descartado el modelo en cascada por su falta de realismo y por la necesidad de volver continuamente a las fases iniciales del proyecto. Parece que se prefiere desarrollar varios prototipos de sistemas software de control en diferentes organizaciones, prototipos realizados con un esfuerzo relativamente pequeño (25 hombres/año).

Problema: Diseñar y desarrollar un sistema de gran envergadura que funcione a la primera y hacerlo dentro de unos plazos lógicos

Solución: Desarrollo siguiendo la metodología de prototipos para reducir al máximo los errores en el software.

Inconvenientes: El alto coste que supone la construcción de varios prototipos del mismo programa.

Estos prototipos irán creándose de acuerdo con abstracciones y aproximaciones simuladas de los sensores y de las características de las armas. Luego se irán ampliando para substituir las aproximaciones por elementos reales y, a medida que el sistema crece en tamaño y capacidad, se aumentará el nivel de realismo y detalle en las simulaciones.

3.4 Potencia de computación

Una consecuencia directa de tener tan enorme cantidad de código y los requisitos en cuanto a velocidad de ejecución es que el software se habrá de ejecutar, de acuerdo con la capacidad actual, en supercomputadores como el Cray X-MP. Un problema adicional es que estas prestaciones sólo se pueden alcanzar, de momento, con el ordenador en tierra, los ordenadores que se llevan al espacio dan prestaciones un orden de magnitud más bajas debido a las condiciones del entorno en el que trabajan.

Para conseguir el rendimiento apropiado se han de utilizar múltiples ordenadores o diseñar máquinas mucho más rápidas a través de avances en el hardware. Se espera que esto sea posible y siga la tendencia de la tecnología a reducir el tamaño, peso y consumo de los ordenadores y a aumentar su velocidad.

Problema: Ejecución masiva de software con velocidades del orden de 100 millones de operaciones por segundo, esta velocidad puede ser del orden de varios gigaflops para los sensores.

Solución: Nuevas arquitecturas de ordenadores. Computadores ópticos.

Inconvenientes: Las nuevas arquitecturas, por el momento, aumentan en gran medida la complejidad.

Un aspecto importante es utilizar esta potencia para simplificar las tareas, no hacer el software más complejo para compensar la falta de hardware. Además, un incremento en la potencia de los ordenadores puede permitir construir depuradores y simuladores que permitan mejorar en gran medida la calidad del software producido.

3.5 Pruebas

Otro gran caballo de batalla de la S.D.I. son las pruebas. El sistema no podrá probarse por completo hasta que sea el momento de utilizarlo y para entonces no puede haber fallos. La validez de un sistema de defensa radica en la confianza que se deposite en él y por tanto, dada la dificultad para hacer pruebas, hay que elegir la arquitectura y los medios de desarrollo adecuados para facilitar al máximo esta tarea.

Problema: *Se trata de un sistema que, por su naturaleza, no va a poder probarse hasta el momento en que entre en funcionamiento, momento en el cual no habrá tiempo para correcciones.*

Solución: *Construcción modular del software para asegurar la fiabilidad de todas las partes, prototipos para disminuir el tipo de errores, investigación en metodologías de simulación.*

Inconvenientes: *La bondad de las partes no garantiza que el todo funcione correctamente, las herramientas de simulación suelen costar varias veces más que el propio sistema que se construye.*

Un primer paso es elegir un sistema distribuido, por las razones ya comentadas. El tener elementos relativamente independientes evita que haya problemas de coordinación y facilita que se puedan comprobar las partes por separado. La simulación también jugará un papel importante para poder comprobar el sistema bajo diferentes condiciones. Y también será importante realizar pruebas continuas durante toda la vida útil del sistema.

3.6 Tolerancia a fallos

La historia del software muestra que en cualquier sistema hay fallos y más si se trata de uno con la envergadura y complejidad del que nos ocupa. En la S.D.I. los fallos son críticos dada su naturaleza, por eso es crucial reducirlos al mínimo y diseñar todo para tolerar fallos. En lo que atañe al hardware tolerante a fallos hay ya una amplia experiencia y es un campo bastante conocido donde se han hecho avances importantes, sin embargo, el software tolerante a fallos es un campo desconocido donde se ha avanzado muy poco.

La complejidad de tolerar fallos tiene mucho que ver con algunos de los puntos que hemos ido mencionando. El más importante es el hecho de que las pruebas no pueden ser completas. Muchos programas se construyen por el método de prueba y error, mejorando sus prestaciones después de cada intento.

Problema: *Hay que asumir que habrá errores, además el sistema de defensa puede ser atacado y quedar inutilizado alguno de sus componentes.*

Solución: *Construcción de sistemas que soporten fallos, utilizar redundancia, duplicar los sistemas.*

Inconvenientes: *La tolerancia a fallos del software es muy poco conocida, sólo se pueden tolerar fallos que se han previsto con antelación y no se conoce perfectamente el sistema.*

Esta metodología no puede utilizarse en la S.D.I. pues el programa ha de diseñarse para estar libre de errores desde un principio, una práctica inexistente, pues generalmente se confía en una fase de pruebas y de operación experimental en la que los errores se van depurando.

Para solucionar este problema, se ha pensado en la arquitectura abierta y distribuida y en el desarrollo de varios prototipos independientes, ya que la probabilidad de que distintos equipos de programadores cometan el mismo error parece remota. La integración de los diferentes prototipos permitirá reducir la tasa de errores en el programa. También se pretende mejorar las técnicas de tolerancia a fallos a través de sistemas redundantes y producir herramientas más perfeccionadas de análisis y depuración.

3.7 Investigación software

Todos estos planteamientos requieren un esfuerzo importante de investigación y mejora en el campo de la ingeniería software. Muchas de las propuestas realizadas para solucionar los problemas que plantea la S.D.I. son de técnicas y metodologías que se encuentran en su estado inicial y lejos de ser seriamente efectivas. Además de investigar en los problemas que plantea la computación masiva, herramientas de simulación, fiabilidad y tolerancia a fallos, se han señalado seis áreas principales de investigación:

- 1.- Técnicas de verificación matemática: Para la verificación formal de programas, al menos de módulos relativamente pequeños y garantizar así su fiabilidad.
- 2.- Lenguajes de especificación: Que permitan reducir los errores, ya que se obliga al programador a seguir una especificación estricta que asegura la coherencia semántica a medida que se desciende en el nivel de detalle de la programación.
- 3.- Computación paralela, concurrente y distribuida: Dadas las características de la S.D.I. gran parte de los problemas van a estar en estas tres líneas. Actualmente existen bastantes sistemas que explotan las ventajas de estas arquitecturas pero la capacidad para utilizarlos está todavía poco clara. El paralelismo, la concurrencia y la distribución de recursos son, por el momento, fuentes adicionales de complejidad.
- 4.- Equipos de desarrollo: Que podríamos denominar desarrollo cooperativo de software, tomando como base las nuevas redes de comunicación y las potentes herramientas de desarrollo que existen. El problema es cómo gestionar estos equipos y encontrar nuevas formas de estructurar el trabajo de programación.
- 5.- Entornos software: La eficiencia de la programación depende en gran medida de la disponibilidad de entornos de alta velocidad y con una amplia gama de herramientas de desarrollo. Hay que profundizar en el desarrollo de entornos software y estudiar cómo afectan al estilo y forma de programar.
- 6.- Mantenimiento: La S.D.I. introduce un problema totalmente nuevo en el campo del mantenimiento. El sistema ha de estar en funcionamiento permanentemente. Las modificaciones en el software son, hoy por hoy, peligrosas y se ha de investigar mucho en este terreno para garantizar los niveles de seguridad que exige la S.D.I.

4. D.L. Parnas y el pesimismo tecnológico

D.L. Parnas formaba parte del Grupo de Organización de la S.D.I., en 1985 presentó su dimisión alegando razones de profesionalidad pues no consideraba ético que le pagaran por algo que él consideraba que no se podía hacer. Ese mismo año publicó un artículo en "American Scientist" titulado "Aspectos Software en los Sistemas de Defensa Estratégica", en el que enumeraba todas las razones por las que creía que no se podría construir un sistema como el de la S.D.I. Desde entonces se toma este artículo como referencia básica de los que consideran la S.D.I. como un imposible tecnológico.

Los argumentos de Parnas se pueden agrupar en cuatro puntos principales:

Complejidad del Software: El proyecto generará un programa de varios millones de líneas de código, esto significa un número muy elevado de estados posibles en un sistema muy poco regular, la verificación con la herramientas actuales es imposible y no se conseguirán avances importantes en este aspecto en el plazo de tiempo que propone la S.D.I. La tarea de construir un sistema de tal envergadura está más allá de nuestra capacidad como programadores, construir tal sistema requeriría cambiar todos los hábitos convencionales del desarrollo software. La complejidad del software se orquesta alrededor de tres puntos fundamentales: el tamaño del programa, el funcionamiento en tiempo real (el software más complejo que existe) y la concurrencia (paralelismo y distribución de recursos, fuentes de complejidad porque todavía no se dispone de métodos apropiados para programar en estos sistemas). Por otro lado, es un sistema cuyas especificaciones dependen del enemigo, que puede cambiarlas sin previo aviso modificando el entorno de funcionamiento de tal manera que el sistema sea inservible. Por último, se trata de un sistema que no es simplificable, dada su naturaleza no se puede reducir la complejidad permitiendo un mayor número de errores, no hay posibilidad de compromiso, ha de funcionar al cien por cien.

Complejidad del Sistema Soporte: Los elementos a integrar son muy variados y en número muy grande, ya hemos visto que se puede estar hablando de miles de estaciones espaciales. La coordinación será extremadamente difícil en los momentos cruciales y el sistema ha de ser capaz de reconfigurarse automáticamente a gran velocidad sin perder capacidad. La base de datos ha de estar permanentemente actualizada, con datos coherentes y permitir accesos múltiples a gran velocidad. No se tiene experiencia en sistemas de semejante envergadura y con unas constricciones temporales y de tolerancia a fallos tan estrictas y no hay posibilidad de probarlo antes de que entre en funcionamiento.

Complejidad Algorítmica: Han de construirse algoritmos que solucionen un problema del que no se conoce el tamaño con exactitud. Se necesitarán ordenadores de alta velocidad capaces de responder en tiempo real y con una eficiencia muy alta a un ataque que por el momento es desconocido y que además puede cambiar con el tiempo (por ejemplo, puede que se utilicen modelos antiguos de cohetes como señuelos multiplicando por dos o tres el número de objetivos que se presentan al sistema y no se puede suponer que esos cohetes viejos no contienen cabezas nucleares). A todo esto se añade el problema de escribir especificaciones para un problema que no se conoce exactamente.

Complejidad de gestión: Dado el tamaño del sistema la gestión del mismo es crucial para obtener un funcionamiento adecuado. Las técnicas de gestión de software están poco avanzadas y al no existir ciclo de vida (el programa no se puede probar, las mejoras no se hacen de acuerdo con rendimientos anteriores, etc.) el esfuerzo a realizar para que el sistema sea fiable a la primera será enorme. Parnas asegura que aún teniendo suficientes conocimientos de ingeniería software y disponiendo de todos los recursos que pidiera no podría hacerlo, ni ahora ni en los próximos 20 años.

5. F.P. Brooks, optimismo tecnológico y otras opiniones

A pesar del panorama que describe Parnas no todo el mundo considera imposible construir el sistema de la S.D.I., el ejemplo más característico es F.P. Brooks, un especialista reconocido en ingeniería software, que, llamado a dar su opinión sobre el tema ante una comisión del Senado, afirmó que no veía ninguna razón seria por la que no pudiera construirse el sistema. De hecho, existen sistemas ya construidos que permiten interceptar un misil y que utilizan más de un millón de líneas de código y hasta 30 elementos entre estaciones terrenas, satélites y bases de lanzamiento. Por otro lado, se están haciendo serios esfuerzos por mejorar las técnicas software empleadas en los sistemas de defensa para poder abordar el problema con mayores garantías de éxito.

También se argumenta que existen muchos sistemas que sin haber conseguido la perfección, funcionan correctamente, en el sentido de que son efectivos y fiables. Que un programa contenga errores no quiere decir que no vaya a realizar la tarea para la que está diseñado, en el caso de la S.D.I. un error no es lo mismo que un agujero en el escudo de protección. El grupo del que Parnas formaba parte concluyó sus trabajos en 1985 afirmando que los recursos de computación y el software de gestión del sistema estaban dentro de lo que puede conseguir la tecnología hardware y software en los próximos años. Recientes avances como las investigaciones en ordenadores ópticos (más ligeros y más rápidos que los convencionales), redes neuronales (para paliar la complejidad del software utilizando técnicas de Inteligencia Artificial) y nuevas arquitecturas para proceso de alta velocidad, conceden un cierto margen de optimismo a los que creen que se puede construir un sistema como el propuesto.

Una opinión muy generalizada es que la S.D.I. no debe verse como un sistema perfecto sino como un sistema que haga que cualquier enemigo se lo tenga que pensar antes de lanzar un ataque, al saber que gran parte de él puede ser detenido (y dar tiempo así a un contraataque que no será capaz de parar). Esto es ya una consideración de tercer nivel pero las repercusiones técnicas son importantes. Como Brooks explicó ante la comisión del senado, el problema fundamental es decidir cuál es el grado de perfección que ha de conseguir el sistema (con un rendimiento del 99,9 %, un ataque de 10.000 armas conseguirá que pasen 10 bombas nucleares). El grupo de Estudios Tecnológicos para la Defensa cree que se puede construir un sistema con una efectividad entre el 90 y el 99 % para mediados de la década de los 90.

Una afirmación que también tienen en cuenta muchos es que un sistema defensivo al menos garantiza que un ataque no será tan destructivo, mientras que la construcción de sistemas ofensivos (como los que se han estado construyendo hasta ahora) lo único que garantiza es la destrucción total del atacante.

Hoy por hoy se han alcanzado algunos de los objetivos que se habían planteado pero la gran mayoría de ellos dentro de lo que hemos denominado primer nivel de complejidad. Así, por ejemplo, se han hecho experimentos con éxito en interceptación de misiles. Pero la mayor parte de la crítica sigue centrándose en el software de control de todo el sistema, sobre todo a la vista de los recortes presupuestarios que han afectado especialmente a esta parte del programa. Parece extenderse la idea de que los avances en ingeniería del software son demasiado lentos como para poder abordar los problemas que plantea la S.D.I. en un futuro cercano y es casi una verdad totalmente aceptada que es muy alta la probabilidad de que se produzca un fallo catastrófico debido al software en la primera batalla real. Las líneas de investigación que se consideran más prometedoras son una muestra palpable de ello: sensores a base de superconductores a baja temperatura, técnicas de fabricación de compuestos con mayor resistencia a los impactos, reconocimiento de imágenes no convencionales, interferometría de amplitud y resolución en el nivel de subpixel, microondas de alta potencia, etc.

6. Factores de tercer nivel

Además de todos los que hemos estado viendo y comentando, existen muchos factores de primordial importancia para el desarrollo de la S.D.I. que están dentro de la categoría que denominamos tercer nivel de complejidad. Estos factores se escapan del ámbito puramente tecnológico e introducen problemas de carácter muy variado, social, económico, ético, etc. Al igual que sucede con toda la tecnología, no podemos concluir un estudio sobre un proyecto tecnológico sin considerar este tipo de factores, que en muchos casos son la clave de éxito.

El primer factor que se considera en los proyectos tecnológicos, aparte de los puramente técnicos, es el económico. En este punto la S.D.I. no es diferente de ningún otro proyecto. El presupuesto inicial estimado de la S.D.I. para los cinco primeros años fue de 26.000 millones de dólares, el coste total no parece estar claro pero las estimaciones van desde los varios cientos de miles de millones hasta más de un billón de dólares. Uno de los argumentos que preocupan más económicamente es que un proyecto como el de la S.D.I. no es una inversión única, el mantenimiento del sistema, una vez construido y en funcionamiento, puede costar entre 50 y 200 mil millones de dólares al año. Como referencia se puede tomar el coste del bombardero B-1B: 28.200 millones, desde la investigación inicial hasta la construcción de 100 unidades. Otra referencia interesante es el "Mosaic Sensor Project", que comenzó hace más de once años y su objetivo era producir sensores infrarrojos con un coste de 25 céntimos (de dólar) cada unidad y poder utilizarlos masivamente. El coste actual es todavía 3.000 dólares cada sensor.

Con este presupuesto muchos temen que la mayor parte de las investigaciones se desvíen hacia tecnologías más maduras, aunque menos efectivas, de forma que se puedan obtener resultados con mayor seguridad y justificar el gasto ante la opinión pública. Esto ha ocurrido de hecho y de momento se ha abandonado el trabajo en integración -el software de control de todo el sistema- favoreciendo ciertos campos en los que ya había resultados (interceptación de misiles, rayos láser y satélites). Una decisión política (y por tanto de tercer nivel) que puede afectar decisivamente al futuro tecnológico de la S.D.I.

Otros problemas que merecen consideración, pero sobre los que no nos vamos a extender, son los tratados internacionales que viola el proyecto de la S.D.I., los problemas de seguridad (demasiados

participantes para asegurar la confidencialidad de las investigaciones) y problemas de transferencia tecnológica (dificultad para emplear en la industria los descubrimientos y avances que se hagan).

Pero, sin duda, uno de los factores de tercer nivel que más han influido en el futuro de la S.D.I. está la actual situación política mundial. Siete años después de ser propuesto, la caída del muro de Berlín, el derrumbamiento hasta el momento pacífico del bloque del Este, el fin de la guerra fría y de muchas de las tensiones internacionales y la fuerte tendencia al desarme de las dos superpotencias han cuestionado un proyecto como el de la S.D.I.

Es justo mencionar que también ha habido algunos problemas tecnológicos que no parecían abordables y que condujeron a reducir el presupuesto y centrarse en objetivos más realistas. Aunque en opinión de algunos ésta fue la puntilla final a la S.D.I., ya que si los verdaderos y grandes problemas estaban en el software y la integración de todo el sistema sería ahí donde habría que hacer el mayor esfuerzo de inversión y, en cambio, se ha preferido favorecer las investigaciones más específicas en armamento y más relacionadas con la física (recordar lo que se decía sobre la importancia de conseguir algún resultado inmediato).

Y aunque la desaparición del proyecto de la S.D.I. puede dar la razón a algunos de los críticos, tampoco es tan fácil que desaparezca. Y aquí es donde entran en juego en toda su relevancia los factores de tercer nivel. Ya mencionamos anteriormente, en un cuadro, la importancia que tiene en la investigación el presupuesto de defensa de los EEUU. Suprimir radicalmente esta fuente de financiación de las Universidades puede provocar muchos y graves problemas. Prueba de ello es el siguiente cuadro sobre la financiación de las universidades en EEUU:

Universidad	Proyectos de Defensa	Proyectos de la Industria
Johns Hopkins U.	357,4	7,8
Georgia Institute of Tech.	52,1	23,6
M.I.T.	45,8	35,1
Carnegie Mellon U.	43,0	16,1
Stanford U.	40,8	11,0
Austin, Texas U.	37,0	3,2
Southern Ca., L.A. U.	35,1	10,0
San Diego, california U.	23,7	5,9
Urbana-Champaign, Illinois U.	18,6	11,4

Datos de 1987, en millones de dólares USA
[IEEE Spectrum, Noviembre 1989, p. 61]

Otro tema de candente actualidad es que el fin de los proyectos de defensa significa que muchas grandes compañías tendrán que reducir plantilla en todos los niveles de la organización: Hughes Aircraft Co. despedirá 6.000 trabajadores, Lockheed Corporation ha reducido su plantilla en más de 8.000 trabajadores, Grumman Corporation suprimirá unos 3.100 puestos de trabajo, Northrop Corporation entre 2.500 y 3.000, Rockwell International aproximadamente 4.000, TRW unos 1.000, Unisys cerca de 2.000, General Electric 4.000, etc.

Las cifras son suficientemente expresivas como para que cualquiera se dé cuenta del problema que conlleva reducir los presupuestos de defensa cuando la universidad y la industria del país dependen en gran medida de ellos. Ahora quizá se puede entender mejor a los que proponen que el problema de que la S.D.I. funcione o no es secundario, que lo importante es tener fondos para investigar y que

al proyecto se le llame como se quiera. También es importante resaltar que gran parte de esos puestos de trabajo son de personal administrativo y de baja cualificación, lo que dificulta volver a encontrar empleo.

En lo que a los ingenieros atañe, parece que no es tan sencillo cambiar de proyectos de defensa a proyectos comerciales normales. La diferencia de planteamiento entre los dos casos es muchas veces muy grande y básica e impide que un ingeniero de defensa se pueda adaptar a la industria comercial. Por ejemplo, en defensa el presupuesto no es tan importante como las prestaciones, en la industria el presupuesto es vital; la producción en defensa es de pocas unidades con tendencia a sobrediseñarlas, en la industria comercial la producción ha de ser masiva y de productos muy ajustados; la tecnología usada en defensa es tecnología punta, mientras que la industria comercial hay más interés por tecnología estándar y por mantener bajo el coste.

¿Cómo aunar, pues, todos estos problemas con los que veíamos al principio sobre satélites, armas en el espacio y tecnología punta?. Una vez más se pone de manifiesto la importancia del tercer nivel y se prueba que efectivamente es el nivel superior pues son las cuestiones de este nivel las que decidirán el futuro de la S.D.I. al margen de cualquier consideración tecnológica.

7. Conclusiones

Lo que más nos interesa resaltar de todo lo visto en este capítulo es cómo un proyecto de la envergadura e importancia del de la S.D.I. puede servirnos como base para estudiar una aplicación práctica de las ideas de la complejidad.

No hay sistema conocido donde los problemas de integración, jerarquías, complejidad, incertidumbre, sistemas, fiabilidad y coordinación sean tan importantes y se planteen a una escala tan grande. Desde el punto de vista de la tecnología, la S.D.I. es un reto importantísimo porque implica una innovación radical en muchos campos pero también es una incógnita enorme dados los problemas que plantea.

Para entenderla en toda su extensión y significado hemos de recurrir a las herramientas conceptuales que hemos ido viendo en capítulos anteriores. Sólo a través de ellas se puede llegar a percibir la naturaleza de todas las implicaciones que presenta y su complejidad global. A través del modelo de niveles hemos desarrollado un mapa de la complejidad de la S.D.I. que nos ha permitido profundizar en los aspectos tanto tecnológicos como sociales.

Tecnológicamente hablando, la S.D.I. recoge la innovación y el desarrollo de los más diversos campos y pretende integrarlos en un único sistema. Es muy interesante estudiar las demandas que plantea en cuanto a potencia de proceso, bases de datos dinámicas, software distribuido, tiempo real, tolerancia a fallos, diseño de software a gran escala, etc. y cómo todos estos campos se han de integrar, considerando todas las implicaciones y consecuencias de cada decisión pues el hardware va a condicionar grandemente al software, el software a la tolerancia a fallos, la tolerancia a fallos a las pruebas, las pruebas a la arquitectura elegida, la arquitectura a la fiabilidad y robustez del conjunto, etc.

Desde el punto de vista de la complejidad, la S.D.I. plantea muchos de los interrogantes que se pretenden resolver a través del estudio de la complejidad. La existencia de varios niveles de diseño,

la coordinación de los diferentes sistemas, el gran número de elementos, la redundancia necesaria, la incertidumbre del diseño o la complejidad de la simulación son conceptos que hemos tratado profusamente en la primera parte de estos apuntes y que aquí se pueden ver en un ejemplo práctico concreto como problemas reales y directamente relacionados con la tecnología.

Y para terminar, recordemos lo que se decía en la introducción, instando de nuevo al lector a que sobre las ideas que se le dan aquí sobre tecnología y un proyecto concreto, intente aplicar las herramientas conceptuales que se han ido desarrollando a lo largo de todos estos apuntes y saque sus propias conclusiones sobre su aplicabilidad y validez.