

컴퓨터技術—未來戰의 要諦

鄭 起 源

1. 防禦主導 未來戰場과 컴퓨터

高度의 尖端技術을 동원한 未來戰爭을 상상할 때에는 數千 킬로미터 밖의 파리라도 꿰뚫을 精密度를 자랑하는 誘導彈에 核彈頭를 장착하여 人口가 밀집된 都市에 터뜨리는 비극적인 장면을 연상하는 경우가 많다. 그러나 尖端技術의 응용은 이러한 悲劇적인 結末만을 향하여 치닫는 것은 아니라 생각된다.

Frank Barnaby는 “自動化된 戰場”에 대한 著述에서 “스마트”한 未來兵器들의 출현으로 공격보다는 방어가 더 효과적인 未來를 내다보고 있다. [1]

戰車部隊를 앞세운 공격군이 國境線을 향하고 있다고 가정하였을 때 이를 無力化시킬 “스마트”한 誘導彈의 防禦가 얼마나 효과적인지를 한번 상상해 보자. 자그마한 誘導彈들이 戰車들을 향하여 소리없이 날아가고 그 각각의 誘導彈은 戰車들을 향하여 진행하다가 그 중 하나를 택하여 砲塔이나 엔진덮개와 같이 가장 약한 부분에 彈을 명중시킨다. 그 誘導彈은 너무나도 “스마트”한 것이라 다른 誘導彈의 標的으로 선정된 戰車를 때리는 일이란 있을 수 없다.

또 30내지 40킬로미터 후방에서 誘導彈이 발사되므로 戰車에 의한 砲射擊圈 밖에 있어 방어측의 위협을 염려할 것이 없다. 이렇게 하여 30萬달러에 달하는 戰車가 불과 數萬달러짜리 誘導彈에 무릎을 꿇게 될 수 밖에 없다.

現代戰에서도 15,000달러 정도인 TOW미사일이 300만달러를 넘는 戰車를 無力化시키는데

어서 만족할 만한 精確도와 信賴度를 보여주고 있다.

앞으로 防禦主導의인 政策이 효과적임을 입증할 만한 징조는 여러가지에서 확인할 수 있다. 地上에서의 對戰車誘導彈, 海上에서는 戰艦을 무력화시키는 對艦誘導彈, 潛水艦을 파괴하는 機雷, 空中에서는 戰鬪機를 격추시키는 對空誘導彈, 그리고 大陸間彈道彈을 명중시켜 파괴하는 誘導彈 등 모두가 적은 비용의 誘導彈으로 高價의 공격무기를 無力化시키는 것들이다. 즉 未來戰爭은 防禦主導의인 戰爭이고 그 主役은 精密誘導彈이라 하여도 무리가 아닐 것이다.

攻擊이 무모한 것이 된다는 위의 發想은 誘導彈에 의한 방어의 成功率에 절대적으로 의존한다. 防禦의 성공률이 낮다면 이를 채택할 필요가 없다. 誘導彈의 성공률은 정밀한 誘導操縱技法과 探索器 機能등에 직접 좌우된다.

또 敵의 공격을 早期에 탐지하고 그 精確한 위치와 速度等 防禦에 필요한 제반 情報를 알아내고 通報하여 주는 기능등 여러가지가 성공률의 밑바탕이 된다. 이러한 諸般技術과 機能은 모두 섬세한 컴퓨터의 도움이 없이는 이루어질 수 없다. 즉 誘導彈의 성능이 “스마트”하다는 것은 誘導彈에 내장된 컴퓨터, 發射統制 컴퓨터 探索器의 정보를 처리하는 컴퓨터, 誘導命令을 계산하는 컴퓨터, 그리고 指揮戰術體系內的 컴퓨터通信網 등이 종합적으로 잘 처리되어서 “스마트”하다는 것이다.

그러나 컴퓨터의 計算이 “스마트”하다는 말을 듣기 위해서는 컴퓨터에 入力되는 情報가 精確하고 完전하여야 하고 또한 그 情報를 처리하는

소프트웨어가 적절하여야 한다. 예를 들면, 레이다가 目標物을 探知하고 追跡하여 컴퓨터네트워크를 이용하여 連續적으로 發射統制用 컴퓨터에 보내면 이를 分析하여 速度, 方向, 位置 등을 계산하고 誘導彈 發射방향을 잡아주며 目標물이 射程圈內에 들어왔을때 發射되도록 制御한다. 이를 위해 컴퓨터間 情報의 送受信 能力과 多量의 情報를 연속적으로 實時間에 高速計算할 수 있는 能力을 갖추어야 한다.

誘導彈에 사용되는 각각의 컴퓨터는 그 使用目的에 따라 서로 다른 여러가지의 까다로운 制約條件들을 만족하여야 한다. 즉, 多數의 컴퓨터들이 各者의 임무를 수행하면서 또한 서로 連結되어 정보를 교환하는 “시스템”의 構成要素로서의 역할이 원활하여야 한다.

그러므로 未來戰爭은 “시스템의인 戰爭”이라고 볼수 있다. 誘導彈의 경우에도 砲隊의 指揮體系下에서 시스템의으로 정보가 처리되어 최종적으로 發射命숨이 집행되는 것이다.

近來 C³I(Command, Control, Communication and Intelligence) 시스템은 시스템內的 情報通信과 “스마트”한 계산을 바탕으로 指揮統制가 이루어지도록 발전하고 있다. 이러한 시스템들은 또 다른 시스템들과 연결되어 방대한 情報通信網이 형성되고 컴퓨터의 應用領域이 확대되어감에 따라 컴퓨터技術에 대한 要求도 확대되고 있다.

컴퓨터의 極小型化, 超高速의 計算速度, 並行處理, 分散處理 및 高信賴度를 유지하는 컴퓨터 하드웨어 및 소프트웨어의 技術所要는 이미 상당한 進전을 보이고 있다. 앞으로 레이저武器, 粒子빔武器(Particle Beam Weapon), 極超短波武器(Microwave Weapon) 및 電磁砲(Electromagnetic Gun) 등의 方向性에너지武器(Directed Energy Weapon)에 대한 研究가 戰略防禦計劃(SDI)에 포함되어 있으므로 이러한 기술응용은 컴퓨터 技術應用과 더불어 防禦主導戰爭의 新技術의 밑바탕이 될것이다. [2]

2. 次期世代 컴퓨터技術

컴퓨터技術의 발전과정을 살펴보면 그 發展速度가 얼마나 빠른지를 實感할 수 있다. 1830年

경 英國의 數學者 찰스베비지(Charles Babbage)가 解析機關(Analytic Machine)을 발표하여 當時 科學者들의 관심을 크게 끌었으나 그의 數學的 概念을 實用的으로 具現시킬 機械技術이 부족하여 實用化되지 못하였다. [3] 이보다 100年 후인 1930年경 獨逸의 Conrad Zuse가 오늘날에도 컴퓨터에 공통적으로 활용되는 二進法 論理體系를 써서 計算機를 만들어 내었다. 이것은 복잡한 數學問題를 사람보다 빨리 풀수 있는 효과적인 컴퓨터였으나 電氣機械式 릴레이(Relay)를 사용하였기 때문에 역시 속도가 느릴 수 밖에 없었다.

英國의 Alan Turing은 디지털컴퓨터製作을 연구한 결과 1943年 COLOSSUS 라는 기계를 만들었는데 2,000개가 넘는 眞空管을 사용한 빠르고 精確한 暗號해독기로서 세계대전 중에 효과적으로 활용되어 終戰時에는 10대나 製作되어 運用되고 있었다.

美國에서는 복잡한 彈道方程式 計算 목적으로 ENIAC(Electronic Numeric Integrator and Calculator)이 1946年 출현하였는데 이것이 世界最初의 電子디지털컴퓨터로 불리워지는 것이다. 18,000餘個의 眞空管과 1,500평방피트의 面積을 차지하는 이 巨大한 컴퓨터는 당시 어느 다른 機械보다도 計算速度가 빠르고 COLOSSUS 가 암호해독 전용기계인 것과는 달리 汎用目的으로 活用될 수 있는 것이었다. 이것이 第一世代 컴퓨터의 출현이다.

그 후 트랜지스터回路的 普遍化로 第二世代 컴퓨터가 제작되기 시작하였는데 트랜지스터化로 因하여 回路에서의 發熱量이 眞空管에 比하여 크게 감소하였고 計算速度가 크게 개선되었으며 運用의 융통성이 증대되어 컴퓨터 應用分野가 크게 擴散되기 시작하였다.

또한 高級프로그래밍言語와 그 言語로 만든 프로그램을 機械語로 變換시켜 주는 컴파일러의 出現으로 컴퓨터 프로그래머라는 職業이 생기기 시작하였으며 컴퓨터 使用人口가 크게 늘어나는 契機가 되었다.

컴퓨터 關聯技術은 급속한 발전을 거듭하여 1970年代에는 第三世代로 進入하였다. 이는 電子工學의 발달로 컴퓨터에 集積回路(IC)를 사용함으로써 컴퓨터의 크기가 大幅 작아졌고 消耗

電力도 크게 줄었으며, 따라서 컴퓨터의 信賴度가 크게 증가되었다.

멀티프로그래밍(Multiprogramming)技法을 사용하여 컴퓨터의 活用度를 높였고 소프트웨어 開發技法도 발전하기 시작하였다. 멀티프로세싱(Multiprocessing)概念이 컴퓨터設計에 반영되어 동시에 여러 使用者가 實時間處理를 가능하게 하였고 一括作業만 처리할 수 있었던 것이 온라인實時間處理를 할수 있게되어 컴퓨터 터미널의 이용이 급증하게 되었고, 따라서 資料通信의 必要性이 증대되었으며 결과적으로 컴퓨터들 間의 資料通信網 構成 및 資料通信用 機器와 소프트웨어가 크게 발전하였고 普遍化되게 되었다.

지금은 第四世代 컴퓨터時代에 있다고 한다. 電子工學의 급속한 발달로 LSI, VLSI, VHSIC등이 컴퓨터製作에 활용되므로 컴퓨터의 高速化, 大容量記憶, 分散處理 및 並列處理技術이 발전하였다.

소프트웨어 開發環境도 自動化를 위한 개발도구(Software Tool)들을 개발보급함으로써 소프트웨어 開發 生産性을 높이고 品質을 向上시켜 소프트웨어 整備度(Maintainability)를 改善시킬 수 있게 되었다.

그러나 第四世代는 第五世代를 향한 中間段階에 불과하다고 하여도 過言이 아니다. 이것은 第三世代와 비교하여 理論的인 改革이 없고 다만 “생각하는 機械”로 불려질 혁명적인 第五世代컴퓨터를 出現시킬 디딤돌 역할만 담당한다는 관점에서이다.

第五世代 컴퓨터는 사람들이 思考能力을 가진 컴퓨터를 바라는 人類의 꿈을 實現시키고자 하는 노력의 產物이다. 즉 人工知能技術의 集約的表現이라 하겠다. 특정한 프로그래밍言語가 아니라 自然語를 쓰고 音聲으로 컴퓨터와 對話한다든지, 컴퓨터에 내장된 知識을 활용하여 여러 가지 문제를 해결한다든지, 人間의 視覺 및 觸覺을 대신할 센서들을 이용하여 事物의 모양과 位置를 알고 空間에서의 作業을 制御한다든지, 人間의 能力을 컴퓨터에 投入시키도록 노력하는 것이다. 이러한 第五世代 컴퓨터의 꿈을 實現시키기 위하여 日本에서는 구체적으로 長期計劃下에 研究開發을 진행하자 先進國들은 技術優位를 日本에 빼가지 않으려고 競爭의 대열에 들어

서게 되었다.

次期世代 컴퓨터開發에 관한 代表的인 研究事例는 美國의 國防省산하 DARPA(Defense Advanced Research Project Agency)에서 주관하는 戰略計算計劃(Strategic Computing Program)과 美國의 여러 컴퓨터 關聯會社들이 投資하여 설립한 MCC(Microelectronics and Computer Technology Corporation)의 프로젝트들, 英國의 Alvey Committee에서 主管하는 Program for Advanced Information Technology, 歐州共同體의 ESPRIT(European Strategic Programme for Research and Development in Information Technology), 그리고 日本의 第五世代 컴퓨터開發프로젝트와 같이 通產省의 政策主導下에 ICOT(Institute for New Generation Computer Technology) 研究 등이다. [4]

프랑스와 독일에서도 ESPRIT에 참여하는 동시에 독립적으로 研究體制를 갖추어 연구중이다.

이러한 計劃들은 대부분 政府次元에서의 강력한 뒷받침으로 수행되고 있다는 점과 人工知能과 소프트웨어工學의 기술의 개발에 초점이 맞추어져 있다는 점이 共通的이라 하겠다. 참고로 美國 國防省主導의 戰略計算計劃(SCP) 研究內容을 살펴보면 그 一次的인 목표는 無人地上車輛(ALV: Autonomous Land Vehicle), 操縱士를 지원하는 專門家시스템(Pilot Associate), 그리고 艦隊指揮센터의 戰場管理시스템(Fleet Command Center Battle Management)이다. 이를 위한 세부과제들은 다음 分野들에 집중되어 있다. [5]

가. 多重處理시스템構造(Multiprocessor System Architecture)

나. 自然語理解(Natural Language Understanding)

다. 專門家시스템技術(Expert System Technology)

라. 音聲理解(Speech Understanding)

마. 視覺理解(Computer Vision)

바. 計算技術下部構造(Infrastructure)

사. 마이크로電子技術(Microelectronics)

이상의 研究課題分野들을 검토해 보면 次期世代의 컴퓨터技術이 軍事的으로 어떻게 응용될 것인가를 짐작해볼 수 있는데 1985年 1月 美國 國防省에 보고된 技術開發戰略에 대한 咨文내용을

보면 더욱 확실하다. [6] 즉,

- 가. 戰鬪시뮬레이션 (Warfare Simulation),
- 나. 戰鬪評價/戰場管理 (Battle Assessment/Battle Management),
- 다. 次世代컴퓨터 및 電子戰 (New-Generation Computer and Electronic Warfare),
- 라. 無人車輛 (Autonomous Vehicles),
- 마. 彈道誘導彈防禦 (Ballistic Missile Defense),
- 바. 操縱士支援 (Pilot Associate),
- 사. 병참관리 (Logistics Management),

등이 가장 效果를 볼수 있는 未來戰場的 技術應用分野라는 것이다. 여기에서도 컴퓨터技術이 人工知能技術과 연합하여 軍事的 應用的 核心技術로 擧擡을 수 없음을 확실하게 한다.

3. 軍用컴퓨터 計算能力的 來日

軍用裝備는 信賴度가 높고, 튼튼하며, 혹독한 環境에서도 잘 動作하여야한다. 이는 컴퓨터의 경우에도 例外가 될수 없다. 특히 戰略防禦計劃에서와 같은 “별들의 전쟁”에서는 宇宙空間에서 컴퓨터動作이 강력한 放射線에 견디는 技術(Radiation Hardened Electronics)을 필요로 한다. 극단적인 狀況에서는 核爆發의 영향까지도 고려하는 設計技術을 필요로 하는 것이다.

그러나 이러한 副次的인 要件보다도 重要한 것은 컴퓨팅파우어, 즉 計算能力이다. 計算能力에 대한 所要는 軍事的 應用에서 항상 크게 마련이고 또 날이 갈수록 증대되고 있다. 이러한 所要를 충족시키기 위하여 컴퓨터의 小型化 및 高速化는 피할 수 없는 추세이다.

VHSIC(Very High Speed Integrated Circuit)나 高速 VLSI(Very Large Scale Integration)技術은 앞으로 數年內에 定着될 展望이다. 이는 곧 高速 小型 軍用컴퓨터가 한장의 카드에 제작되어 나올 수 있음을 뜻한다. 즉 4 MIPS (Million Instructions Per Seconds)의 處理速度를 가진 CPU와 192 K Word(16 bits) 容量의 CMOS RAM 을 한장의 카드에 담은 標準化된 軍用컴퓨터인데 그 크기가 12 inch³, 무게가 0.5 lb 이고 5볼트 直流電源으로 電力消耗가 5 watt 에 불과한 것으로서 現在 市中的 CMOS/VLSI 를 쓴 1 MIPS 컴퓨터 제작기술에서부터 추산해 볼때

충분히 가능한 것이다. [7]

軍用컴퓨터의 강력한 計算能力所要는 4 MIPS 정도의 처리속도로는 만족할 수 없다. 새로운 기술, 즉 並列處理技法을 活用할 수 밖에 없다. 그래서 위의 표준軍用컴퓨터류의 카드 64 개를 조합하고 자료입출력을 위한 카드들과 電源供給 裝置 등을 포함하여 1 ft³의 크기로 60 lb의 무게 및 500 watt 의 電力消耗정도로 제작하게되면 計算能力密度(Computing Power Density)가 1 ft³ 당 200 MIPS 의 처리능력을 갖출 수 있게된다.

만약 현재의 칩(Chip) 크기의 集積回路 제작방법을 탈피하여 웨이퍼규모(Wafer Scale)의 集積回路 제작이 가능해지면 電力消耗의 감소는 물론이고 計算能力密度를 크게 높이는 效果를 본다.

또한 VHSIC 技術은 계속 발전하고, 컴퓨터內의 클럭속도(Clock Rate)가 크게 빨라져 500 MHz 까지 올라가며, VLSI 장치도 갈륨비소(Ga As; Gallium Arsenide)이용으로 技術的인 跳躍을 이룰 展望이다.

그러므로 2000年代 이전에 12M Words(16bits)의 RAM 을 포함하여 1 ft³ 당 3000 MIPS 의 計算能力密度의 目標達成이 예상된다.

이러한 計算能力은 現在技術로는 생각지도 못하는 여러가지 軍事的 應用을 가능하게한다. 예를 들면, 無人戰場로봇車輛(Autonomous Battlefield Robot Vehicle)제작에 필요한 高解像度의 立體映像處理를 實時間에 行하는 응용같은 것들이다.

處理能力的 飛躍的인 발전에 대한 이상의 예측은 現在 내다보이는 技術分野의 發展方向을 토대로 한것이다.

그러나 研究室에서 現在는 전혀 예상못한 새로운 技術이 출현하게 되면 世人을 놀라게 하는 또다른 획기적인 계산능력의 증대를 이루게될지 도 모르는 일이다.

앞으로 出現할지도 모르는 光學컴퓨터를 생각해 보자. 光學컴퓨터는 光纖維通信線과 트랜스페이서(Transphasor)들로 이루어지는 光回路에 레이저光이 흐르도록 설계되는 것이다.

트랜스페이서는 1978년에 발명된 것인데 트랜지스터와 동작특성은 완전히 同一하지만 動作速度가 트랜지스터보다 1000배 이상 빨라서 1

Picosecond 이내에 스위칭動作이 이루어진다. 이렇게 어마어마한 動作속도는 高速병렬처리컴퓨터나 과학기술계산용 및 五世代컴퓨터소프트웨어 구현에 크게 매력적이다. 그러나 전문가들의 見解로는 기술적인 장벽의 해소를 기다리다보면 今世紀內의 실현이 어려울 것으로 예상된다. [3]

4. 소프트웨어技術의 發展方向

컴퓨터 소프트웨어의 개발은 그 對象分野를 크게 두가지로 나누어 볼수 있다. 컴퓨터를 효율적으로 運用하기 위하여 존재하는 運營體制(Operating System)와 프로그래밍言語 및 關聯시스템유틸리티(System Utilities)들이 그 한 分野에 속하고, 컴퓨터를 실제로 活用코자 하는 대상의 應用소프트웨어(Application Software)가 또 다른 한 分野이다. 컴퓨터의 軍事的 應用 측면에서 보면, 그 소프트웨어가 다음의 특성들을 가지고 있다.

- 가. 實時間 處理(Real-Time Processing).
- 나. 分散處理(Distributed Processing).
- 다. 多量情報의 處理(Large Database).
- 라. 科學技術計算(Scientific Computing).

물론 하나의 프로그램이 이상의 네가지 特性을 모두 포함한다는 것은 아니다. 그러나 몇가지 特性을 共有하는 경우는 많다. 이를테면 空對空誘導彈의 發射統制소프트웨어는 그 役割이 實時間에 科學的인 알고리즘(Algorithm)을 동원한 高速計算이 그 生命이고 標的情報의 入手 및 活用, 誘導命令을 生成시키는 컴퓨터와의 通信 등을 위하여 資料通信(Data Communication)을 受發하는 分散處理 技術의 活用이 필요하기도 하다.

그런데 이러한 特性을 살리는 大規模 소프트웨어를 開發하는 프로젝트의 과거경험을 살펴보면 집중판구는 그 프로젝트가 成功的이지 못하였다. 대부분이 開發期間이 延長되거나(Schedule Slippery) 예산이 크게 증액되어 소모되었거나 開發된 소프트웨어에 큰 결함을 내포하고 있어 整備對策이 요구되거나, 또는 整備하기가 아주 어렵게 프로그램이 開發되어 있어 使用者들에게서 큰 不平을 듣게되곤 하였다.

이러한 프로젝트의 失敗들은 소프트웨어 開發

技術에 工學的인 技法들이 적절하게 活用되지 않았으며 電子工業의 발전과 더불어 컴퓨터 하드웨어 技術은 상대적으로 침체되어 있었기 때문이었다.

그래서 學者들은 이를 소프트웨어 危機(Software Crisis)라 부르고 소프트웨어 開發技法에 革新을 가져오고자 努力하는 소프트웨어 工學胎動의 契機가 되었다.

프로그램의 모듈화技法(Modular Programming), 段階的 細分化(Stepwise Refinement), 下向式 設計 및 下向式 具現(Top-down Design and top-down Implementation), 構造的 分析(Structured Analysis), 構造的 設計(Structured Design), 構造的 프로그래밍(Structured Programming), 그리고 소프트웨어 生命週期(Software Life-Cycle) 概念 등은 소프트웨어 開發技法 發展過程에서 나타난 技法 및 概念들이다.

이들은 소프트웨어의 效率性(Efficiency), 完全性(Completeness), 精密性(Accuracy), 正確性(Correctness), 信賴性(Reliability), 整備의 容易性(Maintainability) 및 人間工學的 측면에서의 有用性(Usability) 등을 감안하는 높은 品質의 프로그램을 生産해 내는 것이 목적이다.

이러한 技法들의 적용이 점차 확대되어감에 따라 自然히 소프트웨어 開發道具(Software Tool)들이 등장하여 開發過程을 부분적으로 自動化乃至는 標準化할 것을 시도하였다. 이러한 道具의 適用對象을 살펴보면 다음과 같다. [8]

- 가. 개발사업관리(Project Management)
- 나. 所要分析(Requirement Analysis)
- 다. 設計(Design)
- 라. 具現(Implementation)
- 마. 確認檢證(Validation and Verification)
- 바. 形狀管理(Configuration Management)

소프트웨어 開發道具들의 技術現況은 第四世代 言語들과 IPSE(Intgrated Project Support Environment)들이다. 이들은 대개 위에 열거한 道具의 適用對象들을 複合的으로 적용할 수 있도록 하여 소프트웨어 開發週期 全過程동안 동일한 道具를 사용케 하도록 시도하였다. 그러나 아직은 이러한 道具들의 應用範圍가 한정되어 있고 또 自動化도 完全하지 않다.

美國防省에서는 70年代 중반의 Ada 계획으로

부터 시작하여 다음의 세가지 계획을 추진하였다. [9]

가. Ada 계획

나. STARS(Software Technology for Adaptable Reliable Systems)계획

다. 소프트웨어공학연구원(Software Engineering Institute)

Ada는 美國 國防省 標準 프로그래밍 言語로 제정되어 현재 그普及이 활발하며 많은 사람들이 Ada의 장래를 밝게 점치고 있다. Ada가 成功하리라는 예측을 뒷받침하는 이유는, 첫째 프로그래밍言語의 標準化 측면이고, 둘째는 소프트웨어工學의 概念들이 Ada에 반영되었다는 점이다. [10]

言語의 標準化는 각기 다른 프로젝트의 수행에서도 經驗을 쉽게 活用할 수 있게되고 이미 개발된 프로그램 모듈들을 쉽게 再活用(Software reuse)하게 한다.

現在 Ada는 國防省 뿐만 아니라 ANSI(American National Standards Institute)의 標準으로 채택되었고 NATO의 標準으로 채택될 豫定이며 英國에서는 防衛體系에서 사용되는 2種의 標準言語中的 하나이다.

또한 ISO(The International Standard Organization)에서도 標準으로 채택될 전망이며 歐州 여러나라에서 Ada 컴파일러들이 개발되고 있어 가히 국제적인 標準言語로의 발전이 예상된다.

Ada를 活用한 事例報告는 더욱 희망적이다. McDonnell Douglas社와 Northrop社는 디지털飛行操縱을 위한 航空소프트웨어 開發에서 5倍의 生産性 向上을 달성하고 또 開發費를 크게 절감할 수 있었다고 한다. McDonnell Douglas는 空軍人力資源研究室의 教育支援시스템을 개발하는데도 Ada를 이용하여 크게 성공하였다 한다. [9]

STARS 계획은 Ada를 活用하여 防衛소프트웨어 開發의 生産성을 적어도 10倍이상 향상시키고 信賴度를 크게 改善하고자 하는 노력이다. STARS는 소프트웨어 개발계획수립, 테스트, 整備 및 再使用등 시스템 生命週期 全體過程을 통하여 品質 및 作業能率을 향상시키도록 계획하였다.

SEI(Software Engineering Institute)는 1984년 카네기멜런(Carnegi Mellon Univ.)大學에 설립하

여 소프트웨어工學技法, 開發道具 및 軍事的 應用에 대한 연구를 수행토록하여 現在 계속 확장 발전하고 있다.

Ada, STARS 및 SEI 계획이 잘 진행되어 결실을 보게되면 소프트웨어工學技法에 큰 발전을 이룩하여 컴퓨터 소프트웨어 開發이 점진적으로 自動化되며 軍用소프트웨어가 쉽게, 빨리, 적은 예산으로, 높은 品質과 信賴度를 갖추어 개발될 수 있게된다.

그러므로 DARPA 主導의 이러한 노력으로 高度精密誘導兵器의 개발은 물론, 재래식 兵器의 개량, 戰場管理시스템 개발, 그리고 대규모 戰場管理에 해당하는 SDI 등을 가능하게하는 原動力이 되는 것이다.

5. 結 論

未來戰爭은 방어에 主力을 두도록 政策設定이 될것이다. 이는 技術的인 가능성과 經濟性 및 信賴度를 바탕으로 하는 것인데 컴퓨터技術을 포함하는 尖端技術의 跳躍이 없이는 불가능하다.

精密誘導兵器나 指揮統制通信情報(C³I)體系 및 앞으로 개발될 方向성에너지武器 등의 개발과 實戰運營에 컴퓨터技術能力的 增進은 필수적이다.

半導體技術의 지속적인 발전에 힘입은 회로의 集積化 努力과 GaAs 등 新素材技術에 바탕을 둔 VLSI 기술 및 VHSIC 技術과 컴퓨터의 多重處理(Multiprocessing) 및 並列處理(Parallel Processing) 技術, 人工知能技法 및 소프트웨어開發技法의 발전등이 한데 어울어져 第五世代 컴퓨터의 출현이 전망된다.

소프트웨어 開發技術 또한 크게 개선되어 軍用소프트웨어에 Ada를 標準言語로 쓰고 效率的인 開發技法을 동원하는 道具(Tool)를 活用하여 프로그래밍의 自動化에 크게 進展을 보일 것이다.

이러한 컴퓨터 관련기술들이 총동원되는 未來戰에는 無人車輛이 戰鬪에 참여하고 컴퓨터의 지원을 받는 戰場管理(Battle Management)가 이루어지며 최근 점차적으로 뚜렷해지고 있는 戰略防禦計劃(SDI)에 의한 "별들의 戰爭"이 컴퓨터技術基盤위에 定着할 것이다.

2000年代가 되면 컴퓨터技術은 모든 분야에 보편화되어 컴퓨터가 요즘의 部品같이 취급되고 소프트웨어技術도 보편화하여 自動프로그래밍되는 단계로 발전할 것으로 기대되어 하루빨리 그러한 技術時代를 접하고 싶은 심정이다.

참고 문헌

1. Frank Barnaby, "How the Next War Will Be Fought," Technology Review, Oct. 1986, pp. 27—37.
2. Lloyd S. Takeshita, "The Strategic Defense Initiative and NATO," Military Review, Apr. 1986, pp. 31—37.
3. Stefan Geisenheyner, "Computers: The Fifth Generation," Military Technology, 1986, pp. 44—53.
4. N78—2249—4, 次世代 컴퓨터 구조 및 구성에 관한 개념적 연구, 과학기술처, 1985. 5. 20.
5. Strategic Computing-First Annual Report, DARPA, Feb. 1985.
6. Report of Defense Science Board Task Force on Military Application of New-Generation Computing Technologies, Defense Science Board, 24 Jan. 1985.
7. Edwin M. Drogin, "A Moving Target 10-Year Prediction For Military Computers," Defense Electronics, Aug. 1985, p. 33.
8. Gordon Kirk, "Design Criteria in Military Software", Military Technology, May. 1986, pp. 176—178.
9. John F. Judge, "Ada Progress Satisfies DOD," Defense Electronics, Jun. 1985, pp. 77—87.
10. J.G.P. Barnes, "Ada-The Standard Language for Software Engineering," Military Technology, May. 1986, pp. 68—72.

