

# 국방 소프트웨어산업 활성화 방안 (2)



朴允鎬  
국방품질관리연구소 소프트웨어 실장  
공학 박사

## 방산분야의 실태

현대 무기체계는 과학기술의 발전으로 인해 고도화·첨단화 되어 가고 있는 추세이며, 특히 최근의 전쟁양상을 보면 첨단고도 정밀무기, 전자전무기 등과 같은 소프트웨어 집약형 무기체계를 구성하는 핵심인 소프트웨어의 중요성이 날로 증대되고 있다.

또한 과거에는 무기체계의 하드웨어가 전체 체계에서 차지하는 비중이 높았으나, 이제는 소프트웨어로 인해 소요되는 수명주기 비용과 중요도가 하드웨어를 이미 능가하고 있다.

### \* 체계적인 개발활동 미흡

체계가 고도화, 지능화 될수록 소프트웨어의 비중이 커지고, 소프트웨어에 의해 체계의 성능이 좌우됨을 인식하고는 있으나 이를 어떻게 체계적으로 관리하고 획득할 것인가에는 큰 관심을 두지 않아 인력이나, 시간, 비용투자가 뒤따르지 못하고 있는 실정이다.

국내 개발업체가 기존의 소프트웨어를 체계적으로 관리하기 위한 시도가 실패하는 이유를 장기적

인 관점에서 지속적인 노력을 투입하기 보다는 일과성의 활동으로 그치고, 주로 컨설팅에 의존하여 외부의 노력으로 내부의 문제를 해결하려고 하여 문제의 본질을 파악하는데 실패한 때문으로 분석하고, 소프트웨어 역량을 강화하고 품질을 개선하기 위해 기존의 제품중심의 품질개선 운동과는 다른 프로세스 개선중심의 방법을 추진하고 있다는 점은 국방분야의 소프트웨어 품질활동에 시사하는 바가 매우 크다.

즉, 소프트웨어 품질개선은 제품중심의 개선보다는 제품을 개발하고 생산하는 수명주기 프로세스 중심의 개선된 접근방법이 시도되어야 함을 의미한다.

1998년 말부터 국내 일부 방산업체에서는 소프트웨어 전문개발 그룹의 발의에 의해 CMM을 중심으로 소프트웨어 수명주기 프로세스 개선활동을 추진하고 있으며, 미국의 SEI에서 실시하는 심사원 양성 교육에 직원을 파견하는 등 소프트웨어 관련 전문 인력을 확보하기 위해 상당한 투자를 하고 있다.

소프트웨어 수명주기 프로세스 개선을 위해 美국방성에서 적용중인 CMM 모델을 적용한 국내 개발업체의 소프트웨어 프로세스 개선활동 실태를 살펴보면, 레이더, 유도무기, 무전기 등의 군사용시스템을 전문적으로 개발/생산하는 국내 방산업체는 과거 소프트웨어에 대해서는 다소 전문성이 부족한 형편이었으나, 1998년부터 일부 대기업을 중심으로 국제적으로 변화하는 추세에 발맞추어 경쟁력을 확보하기 위해 하드웨어 중심에서 소프트웨어 중심으로 개발역량을 재편하기 위하여 많은 노력을 기울이고 있다.

#### \*기술자료의 미작성

##### ○ 무기체계 유지보수의 제한사항으로 작용

무기체계가 좋은 성능을 갖도록 개발되는 것도 중요하지만, 약전에 배치된 이후에도 해당 무기체계가 그 성능을 유지하고 필요할 경우에는 효과적

으로, 경제적으로 성능을 향상시킬 수 있도록 보장되는 것도 이에 못지 않게 중요한 사항이다.

특히 소프트웨어의 경우에는, 개발 당시에 발견되지 못했던 결함을 전력화 이후에 발견하는 경우가 상당수 있으며, 전장 환경의 변화에 따라 부분적인 기능의 변경, 추가 등이 수시로 발생하는 것이 사실이다.

그러나 우리의 소프트웨어 개발과정은 앞서 지적했듯이 체계적인 개발을 거치는 경우가 거의 없으므로, 전력화 이후 소프트웨어의 유지보수에 필수적으로 요구되는 기술자료가 생성되지 않기 때문에 유지보수 활동에 근본적인 제한 사항으로 작용하고 있다.

소프트웨어의 경우 하드웨어와 달리 개발경험과 노하우를 보유한 개발 담당인원이 퇴사하거나 지식 관리가 안 될 경우 양산단계 이후의 사업 주관부서인 품관소에서는 사후관리를 진행하는데 상당한 곤란을 겪고 있는 것이 사실이다.

소프트웨어는 하드웨어와 달리 개발자의 특성이 반영되어 있기 때문에 다른 사람이 분석한다는 것은 매우 어렵고 설계개념을 모르는 상태에서는 분석이 불가능한 경우도 발생할 수 있으며 대부분의 프로그래머들은 다른 사람이 작성한 소프트웨어에 대해 심한 거부감을 가지고 있는 것으로 분석되고 있다.

이러한 문제점을 자연스럽게 해결할 수 있는 것이 개발 당시에 생성된 소프트웨어 기술자료인데, 이러한 기술자료가 제대로 작성되지 않고 있는 것이 국내 방산 분야의 현실이다.

1991년에 정부주도로 개발된 OOO무전기의 경우, 전력화 이후 상당 시간이 경과한 후에 항공기용 무전기와 중계기능, 전문처리기 및 보안장비와의 연동기능을 추가해 달라는 군의 추가 요구사항이 제기 되었다.

이 요구사항을 만족시키기 위해서는 OOO무전

기애 내장된 소프트웨어를 변경해야만 했으나, 당시 시제 개발을 담당했던 업체는 도산한 상태이고, 소프트웨어 관련 기술자료도 작성되어 있지 않은 상태였다. 결국 당시 소프트웨어를 개발하였던 업체의 직원을 수소문하여 개별적인 위탁계약에 의해 어렵게 소프트웨어를 개선했던 사례가 있었다.

이 작업을 위해 상당한 시간과 비용이 소모되었음은 물론이다. 만일 이때 해당 소프트웨어에 대한 기술자료가 충실히 작성되어 유지관리가 되고 있었다면 훨씬 경제적으로 개선활동이 수행되었을 것이다.

결국 전력화된 무기체계의 소프트웨어를 유지보수하기 위해서는 개발 당시와 벼금가는 시간, 비용, 자원 등이 소요되어야 원하는 목적을 달성하는 문제가 발생하고 있으며, 이는 정부의 입장에서도 무기체계의 운영유지 비용을 상승시키게 되는 비경제적인 결과를 낳고 있다.

#### ○ 업체의 신규 개발사업지연 초래

소프트웨어의 기술자료가 작성되지 않음으로 인하여 무기체계의 유지보수가 비경제적으로 이루어 진다는 사실은 앞에서 설명하였으나, 기술자료의 미작성 또는 유지관리가 미흡함으로 인하여 발생하는 또 다른 문제는 다른 개발 사업에도 심각한 영향을 미치고 있다는 사실이다.

이에 대한 사례로서, 군관리 업체주도로 개발된 육군의 전술통신체계를 들 수 있다. 이 전술통신체계는 군단 및 사단급 통신망을 구성하는 육군의 기간통신망이다.

따라서 기존에 운용되던 모든 단말 통신장비 및 향후에 개발되는 단말 통신장비는 모두 이 체계에 연동되도록 운용개념이 설정되어 있다. 이미 개발되어 운용 중인 이 체계에 다른 업체에서 최근에 개발된 OOO교환기를 연동해야 하는 상황이 발생하였는데, 이를 해결하기 위해 전술통신체계의 인터페이스 관련 자료를 교환기 개발업체에서 요구하게

되었다.

그러나 모든 인터페이스 특성이 소프트웨어로 구현되어있는 이 체계의 소프트웨어 기술자료는 작성이 되어 있지 않았고, 결국 인터페이스 특성에 관한 자료를 필요한 시기에 교환기 개발업체에 제공할 수 없는 상황이 발생하였다.

결국 이로 인하여 교환기의 사업 자체가 상당히 지연되고, 군의 입장에서도 원하는 장비의 전력화 계획에 차질을 가져 오게 된 사례이다.

한 가지 무기체계의 소프트웨어를 개발하면서 기술자료를 작성하지 않는 것은 언젠가는 해당 개발업체가 또는 다른 국내 업체가 새로운 개발사업을 추진할 때 상당한 부담과 지연요소로 작용하게 됨으로써 그 피해 당사자는 결국 업체 자신이 된다는 점을 간과하지 말아야 한다.

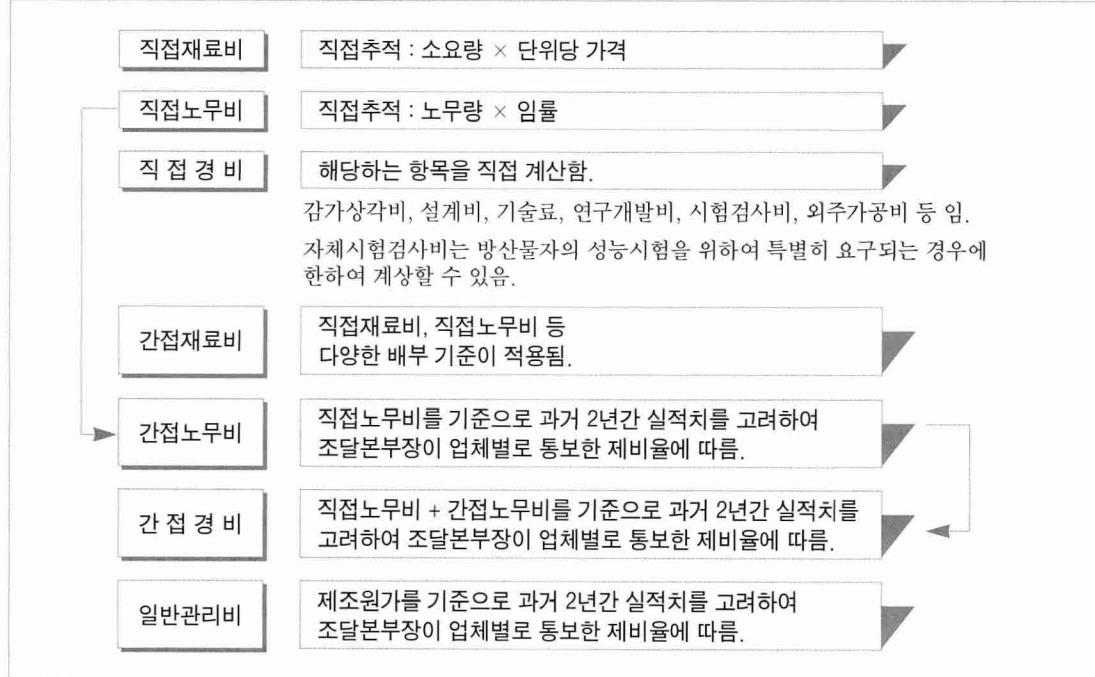
소프트웨어 기술자료가 작성되어 있지 않음으로 인하여 개발업체 입장에서 겪게 되는 또 다른 문제점은 소프트웨어 개발에 소요된 비용, 자원의 투입에 대한 객관적인 증빙자료를 제시할 수 없다는 점이다. 이로 인하여 정부가 소프트웨어 개발에 소요되는 비용을 사후에 인정해 주고자 할 경우에도 여러 가지 곤란한 점이 발생하게 되는 것이다.

따라서 소프트웨어의 기술자료를 작성하고 유지 관리하는 활동은 무기체계의 유지보수에도 필수적으로 요청되는 일임과 동시에 업체의 입장에서도 개발에 소요된 비용을 객관적으로 증빙할 수 있는 자료라는 점에서도 대단히 중요한 사항이다. 이에 대한 사항은 뒤에서 상세히 논의하도록 한다.

#### \* 소프트웨어 개발비용 보상체계 미흡

요구사항의 분석, 설계, 구현, 시험, 단계별 기술자료의 작성 등 소프트웨어를 체계적으로 개발하기 위해서는 상당한 비용을 필요로 하는 것이 사실이다. 개발업체는 개발에 투자하고, 소요된 비용을 정부로부터 보전 받을 수 있어야 앞서 말한 체계적인 소프트웨어 개발활동을 수행할 수 있다.

### 방산물자의 원가계산



따라서 관련기관은 무기체계의 최초 소요기획 단계에서부터 필요 예산을 반영하고 계약시 이를 인정해 주는 등 소프트웨어의 개발에 소요되는 비용을 보상해 주는 개념을 가져야 할 것으로 생각한다.

현재 무기체계의 개발비용은 방산물자의 원가계산에 관한 규칙(국방부령 제513호 : 2000. 2. 10)과 방산물자의 원가계산에 관한 규칙의 시행세칙(국방부계회 45113 - 58 : 2000. 2. 11)과 그 하부의 지침 4종(구분회계지침, 공시보고지침, 이윤산정 및 제비율 적용지침, 방산계약사무처리규칙 시행지침)을 적용하도록 되어 있다.

상기 규칙과 시행세칙에 따르면 원가는 해당 개발사업의 사업관리자와의 조정과정을 통해 우선 원가에 산입하지 아니하는 비원가항목과 원가에 산입하는 원가항목으로 구분된다. 조정과정을 거쳐 원가에 산입하는 원가항목으로 인정받은 각 원가요소는 직접재료비, 간접재료비, 직접노무비, 간접노무

비, 직접경비, 간접경비, 일반관리비로 구분된다.

원가요소의 각 항목은 위의 그림과 같이 계산되는데 원가계산에 있어 기본원칙은 직접비는 추적비에 의하여 계산하고, 간접비는 객관성, 계속성, 인과 또는 효익관계가 인정되는 업체의 배부기준에 따라 배분된 과거 2년간의 간접비 실적치를 기준으로 조달본부에 의하여 산정된 제비율을 통하여 계산하는 것이다.

다음에서는 정부에서 개발비를 지불하는 정부주도 및 업체주도 연구개발과 업체에서 개발비를 지불하는 업체자체 연구개발의 경우로 나누어서 실제 원가 보상 실태를 분석한다.

#### ○ 정부주도 및 업체주도 연구개발

정부주도 및 업체주도 연구개발의 경우 예정가격의 작성시기는 국과연이 체계시제규격서(개발규격서) 및 관련 기술자료를 작성한 이후에 시제업체와 시제품 계약을 체결하는 단계에서 작성하고 있다.

## ○○○체계의 소프트웨어 용역원가

비 목	금 액(원)	비 고
직접노무비	69,901,081	6,621.26 M/HR × 10,557원
간접노무비	35,761,393	35,751,393 × 51.16%
직접경비	-	
간접경비	62,694,279	69,901,081 × 89.69%
소계	168,356,754	
일반관리비	8,417,837	168,356,754 × 5%
이 윤	17,677,459	176,774,592 × 10%
총 원 가	194,452,051	

이때, 원가계산에 적용되는 기준은 방산물자의 원가계산에 관한 규칙의 제5장 용역원가계산 부분을 적용한다. 이런 방식으로 산정된 ○○○체계의 실제 소프트웨어 용역원가를 위의 표에 나타내었다.

이러한 방식으로 소프트웨어 원가를 산정하는 데에는 정부 내에서도 현재 이견이 존재하고 있는 실정이다. 첫번째 의견은 방산물자의 원가계산에 관한 규칙의 용역 원가계산 기준을 적용함으로써 시제품 생산업체의 실발생 비용 위주로 인정하자는 의견이다.

즉, 체계개발 단계까지의 업무를 정부가 수행하며, 시제품 계약체결 내용이 제조 생산과 용역 활동을 함께 필요로 하므로 방산물자의 원가계산에 관한 규칙의 시행세칙 제4조(원가계산 기준의 구분 적용)를 적용하되, 하드웨어는 제조원가계산 기준을, 소프트웨어는 용역 원가계산 기준을 적용하자는 내용이다.

두번째는 방산물자의 원가계산에 관한 규칙 제33조에 따라 적용하자는 의견이 있다. 즉, 소프트웨어 개발 용역에 대해 다른 법령에서 별도 규정한 소프트웨어 사업대가 기준을 적용토록 방산 규정에서 정하고 있고, 시제품 총액 계약명세서상에 하드웨어와 소프트웨어를 별도 구분 명시하는 것으로 하여 정통부고시를 적용하는 것이 타당하다는 내용이다.

현재의 소프트웨어 원가 보상방식은 정부의 적정 예산편성 기준 부재로 효율적 국방예산 관리에 애로가 있으며, 예산 부족시 사업 지연 우려와 함께 적정비용을 업체에 보상하지 못함에 따라 Know-how 관련 기술자료 획득에도 애로가 있으며 불공정계약 관련 민원제기의 소지도 내재하고 있다.

결국, 개발업체는 개발시 보상받지 못했던 비용을 양산시에 추가로 보상해 달라는 요구를 하게 되고 있는 실정이다.

## ○ 업체자체 연구개발

업체자체 연구개발은 개발비를 업체가 먼저 부담하고 이후 양산단계에서 비용을 보전 받는 사업형태이다. 이 경우 예정가격 작성 시기는 연구개발 및 시험평가가 완료된 이후에 방산업체와 초도 생산 또는 양산품 계약 체결 단계에 수행하고 있다.

원가계산의 적용기준은 방산물자의 원가계산에 관한 규칙 제2장 제조에 관한 원가계산 기준의 직접 경비(연구개발비) 항목을 적용하고 있다. 이때 직접비는 사업별, 월별 발생비용을 구분 집계하고, 간접비는 분기 또는 연간 발생된 공통 비용을 사업별 직접인건비를 배부 기준으로 배부 계상하고 있다.

또한 동일사업 발생 비용 중에서 하드웨어와 소프트웨어 해당 비용을 구분하되, 기업 또는 정부 원가 회계처리 기준상 특별히 이를 별도 관리하라는 요구사항이 없기 때문에 대부분 업체 사업비 총액

## ○○○조정기 연도별 계정 과목별 발생비용 현황

(단위 : 천원)

		'97년			'98년			'99년		
계정과목	계	H/W	S/W	계	H/W	S/W	계	H/W	S/W	
재료비	251,737	251,737	-	113,888	113,888	-	70,497	70,497	-	
노무비 (%)	143,047 100%	108,316 75.72%	34,731 24.28%	110,006 100%	69,156 62.8%	40,850 37.13%	204,549 100%	138,468 67.69%	66,081 32.31%	
경비	복리후생비	17,164	8,582	8,582	11,646	7,321	4,325	14,108	9,550	4,558
	여비교통비	22,086	11,043	11,043	14,813	9,312	5,501	36,319	24,586	11,733
	세금과공과	3,872	1,936	1,936	3,407	2,142	1,265	9,082	6,148	2,934
	감가상각비	60,304	30,152	30,152	69,601	43,754	25,847	80,572	54,542	26,030
	수선비	628	314	314	445	280	165	1,934	1,309	625
	보험료	1,100	550	550	423	266	157	1,761	1,192	569
	소모품비	2,340	1,170	1,170	2,283	1,435	848	1,621	1,097	524
	지급수수료	23,126	11,563	11,563	3,904	2,454	1,450	21,816	14,768	7,048
	교육훈련비	2,544	1,272	1,272	536	337	199	839	568	271
	기타	9,412	4,706	4,706	4,469	2,809	1,660	6,064	4,106	1,958
당기 총 연구개발비	142,576	71,288	71,288	111,527	70,110	41,417	174,116	117,866	56,250	
	537,361	431,342	106,019	335,422	253,155	82,267	449,162	326,831	122,331	

으로 관리하며, 만일 이를 하드웨어와 순수 소프트웨어, 체계 통합시험 등 복합적인 연구관련 수행업무를 실적치로 명확히 구분 관리할 경우에는 추가적인 관리조직이 필요한 실정이다.

실제 사례로서 ○○○조정기에 대한 3년간의 계정과목별 발생비용 현황을 위의 표에 나타내었다.

결국, 업체자체 개발에 있어서 소프트웨어 가치를 보상할 경우에는 초도생산 또는 양산품 조달 시점이 개발활동이 완료된 상태이므로 소프트웨어 개발용역 관련 발주 및 수주간 계약 체결시 견적가 산정을 위한 정통부고시 “소프트웨어 사업대가 기준”을 적용하는 경우와 상이할 수 밖에 없고, 정상적인 기업 활동결과 발생된 실비를 보상하는 방산원가 관련 규정과 기업회계 처리 결과를 반영하여 연구개발비로 가치를 판단, 보상하는 수밖에 없는 실정이다.

또한 소프트웨어에 관한 세부규격(사양) 등 기술자료가 마련되어 있지 못하고, 조달 요구에 따른 조달판단 및 계약 체결시 소프트웨어를 별도의 계약

구성항목으로 설정하지 않기 때문에 현행 조달관련 법규상 정통부고시 소프트웨어 사업대가기준을 적용하여 원가를 계산하기가 법적, 제도적으로 곤란한 점이 있다.

또 한 가지 문제로서, 프로젝트별 또는 소프트웨어 및 하드웨어 발생비용 관리 정도에 따라 증빙자료의 신뢰도가 크게 영향을 받기 때문에 업체 자체의 발생 비용 및 자료의 신뢰성에도 문제가 있는 것도 사실이다.

현재의 원가계산 방식은 눈에 보이는 하드웨어를 기준으로 볼 때에는 적용의 타당성이 있으나, 소프트웨어의 개발비용을 산정하기에는 미흡한 방식이라고 할 수 있다.

소프트웨어는 개발당시에 소요되는 인건비를 보전해 주는 것도 중요하지만, 구현된 소프트웨어에 담겨진 기술적인 가치에 대한 보상이 없다는 것도 개발자에겐 불망의 요소로 작용하고 있는 것이 사실이다.

결국 업체가 체계적인 소프트웨어를 개발하려고

해도 소요되는 비용을 정부로부터 보상받을 수 없는 원가계산 방식이 있는 한 아무리 정부가 소프트웨어의 체계적인 관리를 요구한다고 하더라도 현실성이 부족할 수밖에 없는 실정이다.

물론 이를 개발 당시에 업체에게 보상하기 위해서는 예산획득 등 관련되는 문제가 없는 것은 아니지만, 무기체계의 발전추세 및 수명주기를 고려한다면 초기투자비와 향후 소프트웨어의 운영유지비의 상관관계를 염밀히 분석한 무기체계 계약관리가 요구된다고 할 수 있다.

다시 말해, 무기체계 개발이후 생산 및 운영배치 중 시스템 개조 및 성능개량시 필요한 소프트웨어의 개조, 변경, 유지보수 등에 필요한 자체개발 및 변경기술을 습득하기 위해 개발환경 및 인프라를 계약단계에서 계약금액에 반영할 것인지 아닌지에 대한 판단은, 한번 개발된 무기체계의 수명 및 사용기간이 통상 20년 이상인 만큼 거시적인 안목으로 국방예산의 장기적 절감과 비용 대 효과 분석을 실시하여 수명주기 총 비용(total cost)개념을 기반으로 소프트웨어 개발비용의 보상 문제를 판단하여야 할 필요가 있다.

### 미국의 무기체계 소프트웨어 획득관리

美 국방성은 무기체계 소프트웨어를 그 규모가 방대하고, 개발 후 운용기간이 장기간 소요되며, 소요군의 요구사항이 지속적으로 변화함에 따라 소프트웨어에 대한 변경이 수시로 요구되고, 운용되는 하드웨어 속도와 용량면에서 제약조건이 매우 크며, 높은 신뢰도를 요구하는 것으로 분석하고 있다.

여기에서는 미국에서 적용중인 무기체계 소프트웨어의 획득 및 관리개념을 분석하기 위해 소프트웨어 수명주기 관리 개선 및 획득관리를 위해 적용중인 소프트웨어 공학 관련 중점 고려요소를 분석하였다.

이를 위해 美 공군성의 소프트웨어 기술지원센터에서 발행한 “소프트웨어 집약형 시스템의 성공적인 획득 및 관리를 위한 지침서(Guidelines for Successful Acquisition and Management of Software-Intensive System)”를 중심으로 미국의 무기체계 소프트웨어 수명주기 관리 및 획득개념, 개발개념 등을 분석하였다.

#### ● 무기체계 소프트웨어의 획득관리 측면

##### \* 소프트웨어 관리제도 및 적용규정

美 국방성의 획득관리규정은 무기체계와 자동화 정보체계로 분리되어 운영되어 왔으나, 1996년에 범국가적인 획득개혁이라는 차원에서 DoD 5000 시리즈와 DoD 8000시리즈를 DoD 5000.1호, “국방획득” 및 DoD 5000.2-R, “주요국방 획득 프로그램 및 주요 자동화 정보체계의 획득사업을 위한 강제절차(MDAPs & MAIS : Mandatory Procedure for Major Defense Acquisition Program and Major Automated Information System Acquisition System)”로 통합하였다.

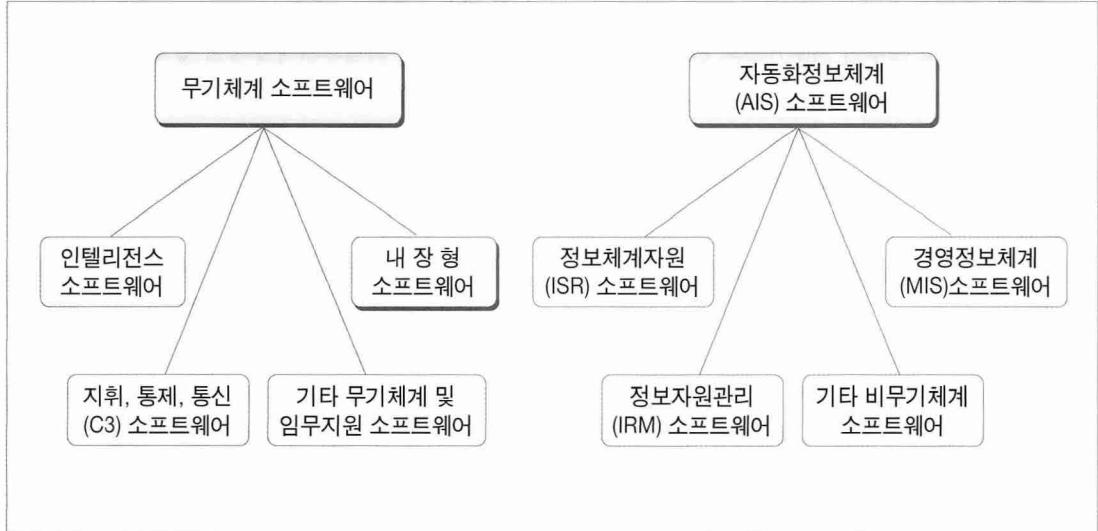
역사적으로 美 국방성은 이들 두 프로그램 부류를 정책과 절차상의 관점에서 다르게 취급했었다.

美 국방성의 획득관리 규정상의 변화와 더불어 규정상에 새롭게 명시된 소프트웨어공학적 주요 적용개념을 살펴보면, DoD 5000.2-R에서는 모든 소프트웨어 개발은 상업적 최고의 프로세스를 이용해 관리 및 설계되어야 하고, 비용, 일정, 그리고 실행위험을 감소시키도록 실행되어야 할 것을 요구하고 있다. 이러한 주요 원리는 다음과 같이 요약할 수 있다.

- 아키텍처 : 소프트웨어 시스템 아키텍처는 개방형 시스템 개념을 지원할 수 있도록 개발되어야 한다. 이는 컴퓨터 제품의 상용품을 활용하거나 또는 모듈성, 재사용성, 확장성 소프트웨어에 기반을 둔 점진적인 개선을 제공하는 것이다.

- 재사용 : 새로운 소프트웨어의 개발 착수전에

### 美 국방성의 소프트웨어 획득범주



소프트웨어 재사용 기회가 정의되고 활용되어야 한다.

– 프로그래밍 언어 : 프로그래밍 언어는 전체 수명주기 비용, 위험, 상호운용성에 대한 잠재성에 영향을 미치는 시스템과 소프트웨어의 문맥속에서 선별된다.(C3I, Ada를 사용)

– 표준 데이터 : 국방성 표준 데이터가 이용된다.

– 계약자 : 계약자는 유사 소프트웨어 시스템에 관한 개발경험과 더불어 성공적인 과거의 수행경력이 필요하며, 명백한 소프트웨어 개발능력과 성숙된 프로세스를 가져야 한다.

– 측정 : 계약자는 기획 및 소프트웨어 프로그램 추적평가, 그리고 소프트웨어 개발 프로세스와 관련된 소프트웨어 제품의 평가와 개선에 있어서 측정 가능한 프로세스를 사용해야 한다.

– 위험관리 : 정보 운영상의 위험을 보장해야 한다.

#### \* 무기체계 소프트웨어의 획득범주

美 국방성은 무기체계(weapon systems)와 자동화정보체계(AIS : Automated Information System) 소프트웨어의 상이한 운영조건에도 불구하고 이들

두 영역을 입력, 기록, 처리, 저장, 통신, 복구, 그리고 컴퓨터에 입력한 정보를 표현함에 있어 동일한 기능을 수행한다고 정의하고 자동화정보체계 및 무기체계의 소프트웨어 집약형 시스템(software intensive system)의 획득관리에 적용할 수 있는 공통의 규정을 사용하고 있다. 이러한 美 국방성 소프트웨어 획득범주는 위의 그림과 같다.

美 국방성에서는 지금까지 소프트웨어 획득의 실패원인을 관리적인 책임의 영역으로 보고 있으며, 이러한 관리적 요소로는 기술-수행 솔루션(technology-driven solution), 불안정한 요구조건, 소프트웨어 자체의 복잡성, 자동화기술이 빈약한 설계활동을 보충해 주리라는 막연한 기대감, 규모/일정/비용에 대한 미흡한 산정, 부적절한 소프트웨어 관리요원, 이러한 요소들의 복합적인 조합에 의한 도미노 효과(domino effect)라고 분석하고 있다.

한편, 무기체계 소프트웨어는 전체 체계에 통합된 구성요소로서 설계되거나 제공되어져야 하며, 무기체계의 통합 부분으로서의 소프트웨어 기능은 설계 및 실행 요구조건을 만족시킬 수 있어야 함을 강조하고 있다.

### \* 소프트웨어 획득관리를 위한

#### 주요 고려요소

미국의 산업 및 정부의 전문가들은 신뢰성 있고, 경제적인 소프트웨어의 획득을 하지 못한 것은 궁극적으로 美 국방성의 부적절한 관리실체에 있으며, 이러한 요소에는 다음과 같은 것이 있다고 분석하였다.

- 기술-수행 솔루션 (Technology-Driven Solution)

- 불안정한 요구조건
  - 소프트웨어 자체의 복잡성
  - 자동화기술이 부족한 설계실행을 보충하리라는 기대감
  - 규모, 일정, 비용에 관한 미흡한 산정
  - 부적절한 소프트웨어 관리요원
  - 이러한 요소들의 조합에 의한 도미노 효과
- 美 국방성이 이러한 실패요소를 관리하기 위해 소프트웨어 획득에 적용하고 있는 주요 개념은 다음과 같다.

#### ○ 소프트웨어 획득시 위험관리 개념 적용

美 국방성에서는 소프트웨어 획득시 통합된 위험 관리의 개념을 적용하고 있는데, 위험관리란 구매자(정부)와 공급자(소프트웨어 개발자 및 유지보수자)라는 2가지 측면에서 고려되어야 하고, 성공적인 구매자가 되기 위해서는 구매자 고유의 위험은 물론 공급자의 위험도 고려해야 한다고 보고 있다. 특히, 효과적인 획득 위험관리자가 되는 것은 구매자의 위험 관리 프로그램과 통합제품팀(IPT : Integrated Product Team) 프로세스를 계약자 위험 관리 프로세스에 통합하는 데 달려 있는 것으로 분석하고 있다.

DoD 5000.2-R에서는 사업관리자(PM : Program Manager)로 하여금 성능, 비용, 그리고 일정을 정의하기 위해 각각의 획득에 대한 위험관리프로그램을 만들도록 요구하고 있다. 이 소프트웨어 획득 위험

관리 프로그램은 다음과 같은 사항을 만족시켜야 할 것을 요구하고 있다.

- 위험 및 위험추진체의 정의 및 추적
- 위험 회피계획의 정의
- 어떻게 위험이 변경되었는지를 결정할 수 있도록 각각의 획득양상에 동안에 지속적인 위험평가의 제공

그러나 이러한 선별된 획득 전략에도 불구하고 美 국방성에서 특별히 관심을 가지고 있는 획득 위험관리의 핵심요소는 다음과 같다.

- 소프트웨어 집약형 시스템 : 많은 소프트웨어 구성품으로 구성된 시스템은 종종 납품이 지연되고, 예산을 초과하며, 사용자의 기대를 만족시키지 못한다.

- 하드웨어/소프트웨어 기술의 급진적인 변경 : 급속한 기술 발달로 인해 전통적인 획득 모델은 종종 구형기술의 시스템획득이라는 결과를 야기하였다. 그러한 시스템은 종종 상당한 비용이 소요되는 계속적인 시스템의 최신화를 요구하였다.

- 통합된 구성요소로서의 인간(HIL : Human In the Loop) : 인간이 통합적 구성품인 시스템의 경우 기술과 사용자의 상호작용 효과는 예측하기가 매우 힘들며, 이러한 사실은 시스템 요구조건을 정의하기 어렵게 만든다. 게다가 시스템 결함은 운용시험 전까지 종종 발견되지 않는다. 또는 가장 최악의 사례로 그러한 결점은 확장된 운영시험에서도 발견되지 않는다. 이러한 현상은 소프트웨어의 후기 개발 단계에 막대한 비용을 초래한다.

- 다수의 사용자들로 구성된 시스템 : 특히 인간과 컴퓨터 인터페이스(HCI : Human & Computer Interface) 설계영역에서 사용자들의 경험 및 경쟁의 변화는 사용자 요구조건을 정의하기 어렵게 만든다. 이러한 문제는 많은 수의 사용자들이 있는 경우 더욱 악화된다.

- 전례가 없는 시스템 : 사용자들은 전례가 없거나

나 경험이 없는 경우 요구조건을 만족시키기 위한 시스템 요구조건을 정의하는데 어려움에 빠진다.

이러한 소프트웨어 획득 위험요소에 대비하기 위해 적용하고 있는 주요 대응방안은 다음 3가지로 요약될 수 있다.

### ■ 위험기반 공급업체 선택

美 육군 통신 및 전자자회 소프트웨어 엔지니어링 이사회(Andy Mills)에서는 공급자들이 어떠한 방식으로 소프트웨어 위험관리를 효과적으로 언급하는가에 기반을 둔 공급자의 기술적 제안서가 평가 방식에 있어 위험기반 공급업체 선정 및 접근방식을 규정하고 있다.

또한 제안요구서(RFP : Request for Proposal)에서 공급자는 계약 전반을 통해 어떻게 위협이 관리되어질 것인가에 관한 평가가 수행되어져야 할 것을 제안하고 있다.

평가절차 동안에 정부는 정부 고유의 위험관리 프로그램을 실행해야 하며, 공급업체 선택에 대한 표준화된 위험관리 접근방법을 적용함으로써 획득 흐름을 효과적으로 증진시키는 완전한 평가가 이루어질 수 있는 것으로 분석하고 있다.

### ■ 위험기반 제안요구서 적용

美 국방성에서는 성공적인 소프트웨어 집약형 시스템의 획득은 위험을 관리하는 계획된 획득경로의 수립에 달려 있다고 분석하고, 위험기반 접근방법을 요구함에 있어 공급자의 제안서는 미래의 기술적인 도전과 위험에 대한 현실적인 평가에 기반을 둔 소프트웨어 활동을 어떻게 계획하고 있는지 설명할 것을 요구하도록 하고 있다.

또한 공급자의 위험관리 계획은 적합한 소프트웨어 아키텍처, 재사용 전략, 요구조건 관리 프로세스, 매트릭스, 개발방법, 툴, 그리고 기술을 통해 소프트웨어 위험에 대해 어떻게 대처할 것인지에 대한 계획을 기술해야 한다고 보고 있다. 사실상 공급자의 제안서는 최초의 소프트웨어 개발계획(SDP :

Software Development Plan)이라고 볼 수 있다.

위험관리에 기반을 둔 제안서를 요구하는 것이 의미하는 사실을 이해하기 위해서는 모든 소프트웨어 개발활동은 하나의 위험 및 또 다른 위험을 완화시키기 위해 추적 가능해야 한다는 사실을 가정해야 한다.

즉, 제안요구서는 공급자의 접근방법이 식별된 소프트웨어 개발위험에 따라 조직화되고, 공급자가 계약실행 전반을 통해서 어떻게 위험완화를 촉진할 것인지를 설명되어야 한다.

### ■ 위험기반 제안서 평가

제안서 평가는 전통적으로 기종 평가 및 선택절차에 참여하는 인증된 기술적 팀에 의존하고 있다. 소프트웨어 위험평가 방법론을 사용하는 것은 위험 식별을 촉진하고, 공급자의 가능성과 장점에 관한 주의를 집중함으로써 평가인력의 능력을 확장시키는 개념이다.

기종선택은 수락 가능한 위험한계 내에서 사용자에게 최고의 가치를 제공하는 팀 능력, 일정, 그리고 비용 목표의 조화에 기반을 두어야 한다.

그러므로 기종 선택팀은 고유의 소프트웨어 개발 위험을 언급 및 조종하는 동안 제품과 기술적 절차, 일정, 그리고 비용 요구조건에 대응할 수 있는 각 공급자의 능력을 평가해야 한다. 평가팀은 다음의 사항을 기반으로 공급자를 식별해야 한다.

- 최고의 실행 기준을 가지고 비교함으로써 결정된 제품, 절차, 접근방법, 관련된 위험
- 프로그램 고유의 중요한 위험요소에 중심을 둔 실행능력
- 모든 강제적인 법적 항목과 관련된 요구조건에 충실하는 것
- 평가된 제안 프로그램에 대해 규모, 복잡도, 그리고 소프트웨어 영역에 있어서 유사한 소프트웨어 개발노력에 관한 과거의 능력

(다음호에 계속)