

□ 특집 □

Y2K 해결 방법론(Non-IT 포함)

맹 칠 현[†]

◆ 목 차 ◆

- | | |
|-----------------------|--------------------------------------|
| 1. 서 론 | 4. Y2K WBS(Work Breakdown Structure) |
| 2. IT 분야 Y2K 해결방안 | 5. 결 론 |
| 3. Non-IT 분야 Y2K 해결방안 | |

1. 서 론

지금까지 컴퓨터상에서 프로그램을 작성할 경우에 날짜항목은 대부분 년도의 아래 두 자리로 정의해 왔다. 즉, 1998년의 경우는 98만 프로그램 상에 년도로서 표시한 것이다. 그 이유는 1900년대에 우리가 살고 있으므로 19를 빼고 98로만 표시해도 1998로 이해해 왔고, 19를 빼면 기억장치(memory), 저장장치(disk), 화면, 장표등에 보다 더 많은 정보를 저장, 기억, 처리할 수 있었기 때문이다. 인간의 편의성을 추구하는 본능이 이렇게 알뜰하게 관리를 해온 것이다.

그런데 문제는 2000년이 다가오면서 발생한 것이다. 2000년은 앞의 두 자리를 빼고 처리하면 20은 없어지고, 00만 남는 것이다. 이는 00이 1900년의 00인지 2000년의 00인지를 컴퓨터가 알 수가 없으며, 이 00이라는 정보를 가지고 연산을 하면 컴퓨터가 작동을 멈추거나, 엉뚱한 결과를 표시하기 때문에 문제가 되는 것이다. 즉, 컴퓨터가 정상처리를 못해서 피해를 주거나, 작동이 안되어서 피해를 주거나 하는 문제가 Y2K인 것이다.

오늘날은 정보시스템이 기업활동이나 사회활동에 깊숙이 관련되어 있으므로 Y2K로 인하여

컴퓨터가 제 기능을 못하면 사회나 기업, 개인에게 지대한 영향을 주기 때문에 이 문제는 어떤 문제 보다도 우선순위를 가지고 대응해야 된다.

여기서는 Y2K의 해결을 어떤 방안으로 하여야 되는지를 살펴 보기로 한다.

2. IT 분야 Y2K 해결방안

2.1 Y2K 대응 지침

Y2K의 대응을 위해서는 기업이나 기관이 보유하고 있는 프로그램, 데이터베이스, 파일등 모든 어플리케이션 자산에 대한 대응이 필요하다. 구체적으로 대응작업을 진행하는데 있어서의 포인트는 년도표시를 다음과 같은 방법에 의해 처리하는 방안이 있다.

- 자리 방식 : 년도를 4자리로 확장 처리하는 방식. 예, 2001
- 2자리 방식 : 일명 윈도우링(windowing) 방식이라고도 한다. 년도를 2자리로 그대로 표시하나, 기준년도(예:40)를 정해서, 기준년도 이상은 1900년대로, 그 이하는 2000년대로 간주하여 처리하는 방식으로 40을 기준으로 40이상은 1940년에서 1999년까지로 간주하고, 40미만은 2000년에서 2039년이라고 간주하여 처리

† 정회원 : SEEC Korea 지사장

어떤 방식을 취할 것인가는 기업이나 기관의 업무 특성, 인력, 잔여기간, 예산 등을 고려하여 결정할 문제인 것이다. 각 방법의 장단점은 (표 1)과 같다.

(표 1)

대응방법	장 점	단 점
4자리 방식	<ul style="list-style-type: none"> · 시스템의 수명이 길다 · 기본적으로 프로그램 로직은 변경하지 않아도 될 	<ul style="list-style-type: none"> · 데이터베이스 등의 변경이 필요 · 디스크 용량 증가 · 화면/장표의 양식 변경 필요 · 과거에 만들어 보관된 파일을 처리할 때는 주의가 필요
2자리 방식	<ul style="list-style-type: none"> · 데이터베이스, 파일 변경이 부필요 · 프로그램단위로 Y2K를 처리하여 가동가능 	<ul style="list-style-type: none"> · 시스템의 수명이 짧다 · 환경기준년 설정이 필요 처리하여 가동가능 · AP트직이 복잡 · 날짜비교, 연산등 내부적인 4자리화가 필요 · 분류, 범위검색등에도 특수처리 필요

그럼 어떤 기준에 의하여 위의 두 가지 방식 중 하나를 선택할 것인가? 다음의 경우엔 4가지 방식이 권장된다.

- (1) 근본적인 치유를 원할 경우
- (2) 수정할 시간이 충분할 경우
- (3) 프로그램 로직을 손대지 않을 경우
- (4) 데이터베이스나 파일이 비교적 작거나 단순한 경우
- (5) 디스크를 확장할 예산이 있는 경우
- (6) 장표나 화면등에 확장할 자리가 충분할 경우
- (7) 향후 5년이내나 10년정도까지 시스템을 변경할 계획이 없는 경우

그러나 어플리케이션마다 그 특성을 고려하여, 4자리 방식과 2자리 방식을 적절히 혼합해서 사용할 수도 있다. 즉, 데이터의 입출력은 2자리 방식을 그대로 쓰고, 데이터베이스상의 날짜 항목과 프로그램 내부처리는 4자리방식으로 한다든가 하는 혼합형도 고려할 수 있다.

이와같이 시스템을 잘 아시는 여러분이 지혜를 모아 결정하여야 할 것이다.

또한 전체 정보 시스템에 대한 방침을 결정하기 전에, 모델 시스템을 가지고 시작(pilot)시스템을 해보아서 그 결과를 보고, 대응방침, 절차, 소요인력, 예산등을 확인해 본 후 최종 방침을 선정하여야 한다.

2.1 Y2K 대응 작업 진행 방안

미국의 가트너그룹의 추정에 의하면, Y2K의 대응공정의 비율은 다음과 같다.

사전조사	6%
조사분석	20%
개수작업	20%
테스트	45%
본 가동	9%

이 이외에도 전 공정의 25%에 해당하는 프로젝트 매니지먼트(project management) 노력이 필요하다고 한다. Y2K 해결에 있어서 프로젝트 매니지먼트능력은 성공의 중요한 요소가 된다. 특히 Y2K의 대응은 작업공정 모델이나 방법론에 따라서 수행하는 것이 시행 착오를 방지하며 시간낭비를 줄일 수 있다.

Y2K를 효율적으로 대응하기 위하여 다음과 같은 작업공정 모델을 제시한다.

작업 공정 모델 ;

- (1) 인식(Awareness) 단계

- (2) 영향평가 단계
- (3) 변환 단계
- (4) 검증 단계
- (5) 시험운영 단계

(1) 인식(Awareness) 단계

인식단계에서는 Y2K를 정의하고, 최고경영층의 지원을 얻는다. 또한, 본격적인 Y2K 해결 작업을 시행하기 전에 Y2K 대응 검토 프로젝트를 조직하고, 전체 작업의 마일스톤(milestone)을 입안하는 것을 주 목적으로 한다.

그 구체적인 활동은 다음과 같다.

- a) Y2K의 심각성을 최고경영층을 포함한 사내 관련 부서에 제공한다.
- b) 전체 작업량은 어느 정도인지를 개략 산출한다.
- c) 그 작업량에 따라서 실현 가능한 스케줄과 추진체제를 입안한다.
- d) 시스템 계획을 포함해서 기존 계획과 일치하는지를 점검하고, 우선순위를 부여한다.
- e) Y2K 대응에 나타날 각종 제약조건(예, 요원 문제, 설비문제, 투자비용 등을 정리한다.

인식단계에서의 산출물은 다음과 같다.

- 문제영향 정의서
- 종합계획서 및 지침서
- 캠페인 계획서 및 실시완료 보고서

(2) 영향평가(Assessment) 단계

이 단계에서는 Y2K가 기업이나 기관에 미칠 수 있는 영향을 평가한다.

이 단계에서는 모든 어플리케이션 자산의 조사와 분석을 하는 어플리케이션 자산조사와 어플리케이션 영향도 조사를 실시한다. 동시에 사용하고 있는 시스템 소프트웨어, 미들웨어 등의 사용 현황과 그 소프트웨어의 레벨(level)을

조사한다.

a) 어플리케이션 자산조사

현재 보유하고 있는 어플리케이션 자산을 전부 조사하고, 현재 실제로 사용중인 자산을 선별하여 확정한다.

b) 어플리케이션 영향도 조사

현재 실제 사용중인 자산이 어느 정도인가를 확정된 후 각각의 어플리케이션의 날짜가 어떤 형태로 사용되어 있고, 그 영향은 2000년에 어떻게 미칠 것 같은가를 조사, 분석한다

실제는 현재 사용중인 어플리케이션 자산으로부터 날짜 정보를 찾아내는 작업이지만, 프로그램, JCL/WFL 및 데이터를 저장하고 있는 D/B나 파일도 해당된다.

동시에 Y2K가 기업에 어떤 영향을 주며, 정보 시스템에 영향을 주는가를 조사해야 한다.

c) 어플리케이션 대응 방침 수립

어떻게 어플리케이션을 수정할 것인가의 대응 방침을 수립한다.

가) 방침 입안

우선, 2000년을 4자리로 할까, 2자리로 할까 또 다른 방법을 사용할까를 처리 모듈(module) 별로 결정한다. 여기서 처리 모듈이라 함은, 데이터 파일이나 데이터베이스 상의 날짜정보, 프로그램상의 날짜정보의 처리 로직(logic), 화면이나 장표상의 날짜정보 항목에 대한 입력, 디스플레이(display) 등을 말한다.

다음으로는 개수작업의 절차를 확실하게 정한다. 그리고, 개수작업에 소요될 인력을 산출하기 위한 방법과 기준안을 작성해서 실행한다.

나) 모델 시행

대응 방침이 잘 되었는가, 작업 절차가 효율적인가, 소요 인력 산출은 타당한가를 실제 모델을 검증해 본다.

다) 계획 수립

대응하기 위한 계획을 수립한다. 실시계획서를

작성한다. 실지로 작업을 할 경우에는 한꺼번에 어플리케이션 전체를 개수하는 것이 아니고, 업무시스템(어플리케이션 시스템을 구성하고 있는 서브시스템) 단위로 대응해 나가는 것이 일반적이고, 이 방법이 추천되어 진다. 실제로 외국에서 해 본 결과 서브시스템 단위로 하는 것이 훨씬 빠르고, 성공적임이 밝혀 졌다. 따라서, 업무 시스템 단위로 실시계획을 수립하는 것이 필요하다. 이 때 우선순위를 설정하여 추진하는 것이 중요하다.

영향평가 단계의 산출물은 다음과 같다.

- 적합성 기준 정의
- 주요업무 및 프로세스 확인서
- 현 보유 자산 일람표
- 사용중인 자산 일람표
- 디스크 용량 조사표
- 날짜를 가진 파일 일람표(D/B, 파일, 화면, 장표)
- Vendor S/W 조사 일람표
- 영향도 범위 조사표
- 인터페이스 조사표
- 위험분석과 시스템 Contingency(복구) 계획

(3) 변환(Renovation) 단계

대상 플랫폼, 어플리케이션, 데이터베이스 및 유틸리티를 변환, 대체, 또는 폐기하고, 인터페이스를 수정한다.

a) 시스템 개수준비

작업실시전의 준비작업으로서, 대응 매뉴얼(manual)이나 개별 룰(rule), 작업에 필요한 시스템 도큐먼트(document)등을 작성한다.

b) 시스템 개수

소요 인력이 많이 드는 테스트 공정을 효율적으로 하기 위해서는 테스트를 실행 하고, 검증할 수 있는 환경을 준비하여 프로그램의 개수와

테스트를 해야한다

이 실시공정에서는 통상 운영시에 해오든 시스템 개수작업과 같은 흐름을 따라한다. 단, 개수작업이 장기간에 걸쳐 이루어지기 때문에 정상적인 테스트 환경을 준비해야 한다.

변환단계의 산출물은 다음과 같다.

- 시스템 변환 결과 문서
- 변경관리 문서
- 평가 보고서
- 시험 계획서
- 새로 작성한 스펙
- 대응 완료 업무프로그램표

(4) 검증(Validation) 단계

변환 또는 대체된 플랫폼, 어플리케이션, 데이터베이스 및 유틸리티를 검증한다. 운영환경에서 이들 플랫폼, 어플리케이션, 데이터베이스 및 유틸리티의 성능, 기능 및 통합성을 테스트한다.

- a) 테스트 계획을 수립해서, 테스트 방법이나 검증방법, 스케줄등을 결정한다.
- b) 테스트를 실시하고, 본 가동으로의 전환 작업을 수행한다

데이터 파일이나 데이터베이스상의 항목을 변경할 경우는 특히 주의해서 전환 작업을 하여야 한다.

이 단계의 산출물은 다음과 같다.

- 테스트 계획 대비 결과 보고서
- 테스트 조직도
- 테스트데이터 및 시나리오

(5) 시험운영(Implementation) 단계

변환 또는 대체된 플랫폼, 어플리케이션, 데이터베이스, 유틸리티 및 인터페이스 등을 실행(시험운영)한다. 필요에 따라 만약의 경

우를 대비한 데이터 교환에 관한 계획을 이행한다.

이 단계에서의 산출물은 다음과 같다.

- 시험운영 계획서
- 전환환경 및 절차서
- 문제발생 대비계획 및 복구계획
- 인수시험서

3. Non-IT 분야 Y2K 해결방안

Non-IT분야라 함은 자동화된 생산설비, 운송설비 등 Embedded System을 내장한 장비들에 대한 IT분야와 동일한 유형의 Y2K를 말한다. Embedded System은 자동화된 장비, 시설, 설비 등의 제어, 감시, 운영지원을 목적으로 구성된 각종 H/W, F/W, S/W의 총칭이다. 이는 마이크로칩, Firmware, 시스템 S/W, 범용 package, 응용 S/W등이 혼합되어 구성되어 있는데 세계적으로 약 250억개의 마이크로칩 탑재장비중 5000만개 이상이 문제를 발생시킬 것으로 예상된다.

자동화 설비의 종류와 구성요소를 보면 다음과 같다.

- 전장기기 ; PLC, DCS, SCADA, RTU, CNC, 전용Controller(Weld, Motion,.)등
- 계측기기 ; 분석기, 기록기, 센서(Transmitter, Transducer, Convertor,.)등
- 통신기기 ; 필드통신기(DH+, ModBus+, NodeBus), 네트워크통신기(HUB, Switch.)등
- 전산기기 ; PC, Workstation, Micro-computer, I/O장치(Storage, Terminal..)등
- Package 응용 S/W ; 공정관리, 생산관리, 설비관리, BAS, CAD, CAE, SPC, POP등
- System S/W ; Firmware, O/S, DBMS, Middleware, NMS등

특히 Non-IT분야 Y2K의 특징은 다음과 같다.

- 난해성 ; 날짜사용여부의 파악 및 해결방법론 도입 애로
- 위험성 ; 테스트할 때 완전히 Down가능성 상존
- 치명성 ; 문제 발생시 손해 또는 재해가 치명적
- 의존성 ; 문제 해결은 제작업체나 제작자에 의존
- 방대성 ; 점검 대상, 소요기간, 소요인력이 과다
- 전문성 ; 제조업체 미지원시 자체 전문가가 필요

3.1 Non-IT 분야 대응 단계

Non-IT분야의 대응은 접근방법은 IT와 유사한 다음 5단계가 있다.

- 1 단계 ; Non-IT분야 자산조사
- 2 단계 ; 확인 조사/분석
- 3 단계 ; 해결 계획 수립
- 4 단계 ; 수정, 교체
- 5 단계 ; 테스트

(1) Non-IT 분야 자산조사 단계

이 단계에서는 관련 제품(물품)목록에 관한 정보를 수집조사하고, 고객의 데이터를 수집한다. 수집된 데이터는 Site담당자와 Technical analyst가 분석하여 D/B(repository)를 구축한다. 이를 위해서는 Client Kickoff Meeting이 권장되어 진다. 이 Meeting을 통해 다음 사항을 확인한다.

- 프로젝트의 boundary 설정
- 장소와 위치(sites/locations) 확인
- 장소와 위치의 우선순위 설정
- 사전에 agenda를 배포
- 프로젝트 SOW(Statement of Work)를 점검
- 프로젝트 범위에 대한 client의 기대를 토의하여 상호 의견조정과 합의
- 해당 site의 담당자 확정

- Business영향도에 따라 어느 곳부터 할 것인가 우선순위 선정

이 단계는 확인된 Inventory와 영향분석 조사서를 client에게 전달하고, 다음 단계로 진행할 것인가에 대한 동의를 받는 Phase End Review를 끝으로 종료한다.

(2) 확인 조사/분석 단계

이 단계에서는 조사된 자산(제품, 물품)을 분석하여, 해당 제품이 Y2K를 준수하였는지를 점검한다. 특히 여기서는 제품 제작사, 납품회사에게 일일히 문서를 발송하여 확인을 받아야 하며, 가급적 실명확인을 받도록 한다. 문제가 되는 제품의 계약사항을 점검하고, 업무에 미치는 영향을 분석한다. 이 단계의 활동은 다음과 같다.

- Inventory 단계에서 수집된 데이터 분석
- 보고서 작성
- 조직내에 어디에 mission critical한 것이 있는지를 보여 줌
- Repository(D/B)내의 데이터를 검사하여 일관성이 있는지를 확인
- 보고서를 조사한 사이트들에 다시 돌려 보내어, 더 보완이 필요한지를 확인 받음
- 데이터는 다음과 같이 분류한다 ; No Information, Questionable, Non-Compliant, Vendor Claims Compliance, Client Accepts Compliance
- Inventory단계에서 발견된 non-compliant나 questionable한 항목이 business에 얼마나 risk를 주는가 확인
- 영향의 심각성에 따라 우선순위를 작성
- Risk가 많은 vendor나 제조업체에 대응상태를 문의하는 문서발송
- 인터넷이나 research partner를 통해 vendor조사

(3) 해결계획 수립 단계

문제가 있는 제품에 대한 수정전략을 강구한다. 수정전략으로서는 Repair, Replacement, Retirement가 있다. 수정전략은 합동 Workshop을 통해 토론을 거친 후에 결정한다. 전략을 수립한 후에는 프로젝트 계획을 만들어야 한다. 이 때는 Vendor와 확인이 필요하고, 소요자원(인력, 시간, 예산)에 대한 견적이 필요하다. 특히, Vendor의 대응 상태에 따라 주문 해야 할 부품이나 시스템이 원하는 시점에 도착할 것인가, 만약 도착이 안될 경우는 어떻게 할 것인가, 다시 주문해도 시간적으로 충분인가를 고려하여 상세한 작업 계획을 작성한다.

(4) 실행 단계

프로젝트 작업팀을 운영하여 확인된 장비를 수리하거나, 대체하거나, 폐기하거나 하고 반드시 이에 대한 정보를 D/B에 경신하여 보존한다.

(5) 테스트 단계

테스트 절차에 의해 신 장비나 수리된 장비를 테스트하고 이를 문서화 한다. 또한 테스트 결과를 점검하고, D/B에 정보를 경신한다. 테스트 계획은 성공의 key이기 때문에 테스트의 목적과 범위를 명확히 해야 한다. 상세 테스트 계획에는 테스트 항목의 확인, 테스트 해야 할 기능의 확인, 테스트 할 방안의 확인, 합격/불합격 기준, 테스트할 것과 결과물, 테스트 계획, Risk와 Contingency의 확인, 테스트 절차 승인을 포함하여야 한다.

모든 테스트가 완료된 후에는 최종 보고서를 작성하여 승인을 받는다.

4. Y2K WBS(Work Breakdown Structure)

Step	Task/Step Name	Notes
1	기업의 Y2K 분석 계획 작성	
1-1	Enterprise Analysis Planning 작업계획을 개발한다.	
1-2	재개발 관련 정보시스템의 goals를 설정한다.	goals을 설정하기 위해 전략계획을 활용
1-2-1	goals을 설정하기 위해 document outline을 산출한다.	
1-2-2	경영층과 review한다.	
1-3	Enterprise Assessment Objectives를 설정한다.	개략적인 information, goals, objective를 전략계획에 기재한다.
1-4	Enterprise-Wide 분석범위를 설정한다.	business에 관련된 시스템을 확인
1-4-1	모든 application system의 list를 확보	
1-4-2	모든 business area의 list를 확보	
1-4-3	각 business area에 application system의 관련성 조사	
1-4-4	타 business areas나 application systems과 어떤 interface와 relationship을 가지고 있나를 확인	
1-4-5	business areas나 units이 scope내인지를 확인	
1-4-6	조사할 mainframe sites를 확인	
1-4-7	조사할 distributed sites를 확인	
1-4-8	Non-IT(Information Technology) business areas를 어떻게access할 것인가를 결정	
1-4-9	source program수와 lines of code를 개략적으로 견적	
1-5	Enterprise Assessment Step List를 작성	
1-6	Assessment Roles과 Deliverables를 검토	input이 무엇이고, 어떤 resources가 필요하고, 무슨 deliverables가 필요?
1-7	Enterprise Assessment 견적	Activities 2/3/4를 하는데 얼마나 걸리는가를 견적
1-8	Enterprise Analysis 작업계획을 확정	경영층과 계획을 검토하고, software assessment tool은 이용가능한지, 필요한 skill level은 있는지, 경영층이 무엇을 commitment해야 하는지 조정

Step	Task/Step Name	Notes
2	Enterprise-Wide Y2K 분석 수행	이 작업에 의해 system inventory가 작성된다.
2-1	System Inventory를 categorize한다.	
2-2	Technological Attributes를 요약한다.	각 시스템에 대하여 HW Type, OS, major languages types, DBMS등을 조사
2-2-1	Mainframes에 관한 고객조사	
2-2-2	User systems에 관한 고객조사	
2-3	Enterprise Physical Systems의 Inventory	Executable modules의 수, Sources programs의 수, Lines Of Code, Systems future등을 상세히 획득
2-3-1	조사할 각 시스템의 확인	
2-3-2	각 시스템 조사에 사용되어질 tools의 확인	
2-3-3	Executable modules수의 결정	
2-3-4	Sources programs수의 결정	
2-3-5	Sources programs의 Lines of Code수를 결정	
2-3-6	Subsystems(application)이 Y2K 대응이 되었는지, 어떻게 되었는지 결정	
2-3-7	Subsystems에 대한 예상 failure date를 결정	
2-3-8	Systems/subsystems을 replace할 것인가, rewritten 할 것인가, delete를 결정	
2-3-9	2-3-9 2-3-8에 대한 프로젝트 완료 target date를 결정	
2-3-10	Systems/subsystems이 vendor supplied된 것인지, System SW인지, customer written인지 확인/결정	
2-3-11	Vendor supplied software가 customer에 의해 maintenance되는지, vendor에 의해 되는지 확인/결정	
2-3-12	Vendor supplied products의 source code가 available 한지를 확인/결정	
2-4	Technical Summary Legacy Model을 update	Repository tool을 사용
2-5	현재의 Technical Architecture를 요약	component type, 수 등을 조사하는데 얼마나 어려운지, 시스템간에는 어떤 일 관성이 있는지 등을 strategy를 update
2-6	Physical Data Architecture를 분석	Business areas와 subsystems에 관련되고, store하는 모든 data를 capture
2-6-1	JCL, control tables과 Database tables을 확인	
2-6-2	외부 agents나 의존도를 알 수 있도록 architectural model을 작성	
2-7	상세한 Enterprise-wide Physical Data 사용법	
2-7-1	CRUD matrix를 작성	
2-7-2	Systems과 subsystems사이에 공유되어진 저장된 데이터를 확정하기위해 matrix를 분석	
2-8	Open repository에 데이터 사용법을 load	Repository(logical meta-model)을 갱신
2-8-1	Repository갱신 후 중복이 있나를 확인	

Step	Task/Step Name	Notes
3 3-1	Century Date변경 Upgrade 단위 설정 현 비즈니스 구조(architecture)를 분석	이 task는 Upgrade단위를 만들고, 현 IS프로젝트의 efforts를 확인
3-1-1	주된 비즈니스 영역을 문서로 작성	비즈니스 영역내의 시스템에 관련된 기능을 결정하고 사고일(failure dates)을 확인. 위험확인 보고서 작성
3-1-2	비즈니스 영역과 관련있는 주된 비즈니스	
3-1-3	제품과 서비스를 확인	
3-1-4	Y2K로 비즈니스에 영향을 주는 잠재	
3-1-5	사고일(failure date)을 확인	
3-1-6	언제부터 시작해야 되는지를 확인	
3-1-7	저장된 공유데이터 사용법을 통해 연결	
3-2	시스템 확인	
3-2-1	기록되어 있지 않은 사용자(user)시스템을	
3-2-2	확인	
3-2-3	위험확인(Risk assessment) 보고서 작성	
3-2-4	현 IS프로젝트 활동 현황 확인	현 MIS프로젝트 활동 현황을 작성하고, 무슨MIS프로젝트가 Y2K영향을 받는지?
3-2-5	현재 진행중이거나 계획중인 프로젝트 현	
3-2-6	황을 요약하여 중요문서로 확보	
3-2-7	사용자 관점에서 계획중이거나 진행중인	
3-2-8	프로젝트 확인	
3-2-9	그 프로젝트가 데이터 전환인지 시스템	
3-2-10	교체인지를 확인	
3-2-11	교체시기와 계획일까지 가능한가 분석	
3-2-12	위험확인 보고서 경신	
3-3	작업분할 전략 수립	비즈니스 영역, 기능, 연계성, 구성요소 수
3-3-1	공유 데이터 저장소와 출력데이터 저장소	에 기초하여 upgrade단위 설정
3-3-2	확인	
3-3-3	시스템의 주된 형태 확인	
3-3-4	주 공통항목 확인	
3-3-5	필요한 규모 확인	
3-3-6	핵심/비핵심 시스템 확인	
3-3-7	교체시기, 계획 점검	
3-3-8	시스템의 사고일(failure date) 점검	
3-3-9	각 upgrade에 대비한 브릿지(bridge) 작성	
3-3-10	Upgrade우선순위 설정	

Step	Task/Step Name	Notes
4	기업 2000년 전략 확정	산출물; 최종 작업 계획
4-1	기업의 IS기반구조(infrastructure) 확인	기반구조가 Y2K에 어떤 영향을 미치나?
4-1-1	경영층의 인지도, 협력구조 적합성 확인	
4-1-2	시스템/applications 표준화 적합성 확인	
4-1-3	시스템 형상과 분류 절차 적합성 확인	변경관리, 이행절차, 일관성
4-1-4	각종 절차의 적합성 확인	유지/보수, 프로젝트 관리, 품질관리, 감사 등
4-1-5	테스트와 검증과정 적합성 확인	
4-1-6	Y2K를 위한 기술기반(technology infrastructure)	작업환경 및 도구 확인
4-1-7	시스템 소프트웨어, 하드웨어 적정성 확인	
4-1-8	조직의 안정성 확인	
4-1-9	제반 적합성 여부 확인/점검	
4-2	프로젝트 권장안(recommendation) 작성	상호 기능적 프로젝트 대체안과 권장안 작성
4-3	재개발(redevelopment) 시나리오 권장안 작성	
4-3-1	날자 필드 전환 프로젝트 검증	
4-3-2	교체(replacement) 프로젝트 검증	
4-3-3	발견된 사실에 기초해서 각 비즈니스 영역에 대하여 하나 또는 복수의 시나리오 작성 보고	
4-4	최종 Y2K 전략 확정	최종 계획을 전략보고서로 작성 문서화
4-4-1	개요, 부록 등을 작성	
4-4-2	우선순위도, 정보시스템 기반구조, 필요사항, 위험, 비용 등을 요약	외부조직에 대한 지침서 작성
4-4-3	우선순위 관리, 훈련, 이행계획, 고려사항 등을 토의	
4-4-4	적합성확인 전략을 검토	
4-4-5	소요예산을 요약	
4-4-6	보고서 검토, 변경 후 작성완료	

Step	Task/Step Name	Notes
5	영향평가(Assessment) 계획 확정	
5-1	영향평가 태스크 리스트(task list) 작성	각 단위(upgrade unit)에 대한 영향평가 계획 작성
5-1-1	기술적 환경(technical environment)에 대한 작업 태스크 구축	
5-1-2	교체할 구성요소(component)에 대한 작업 태스크 구축	
5-1-3	지원 팀 인터뷰 작성	
5-2	환경분석작업 계획 확정	물리적 시스템, 외부시스템 자산에 대한 확인,분류, 작업량 산출. 분석 결과 점검
5-2-1	환경분석 태스크에 대한 시간 견적	
5-2-2	환경분석 결과에 대한 점검	
5-2-3	경영층의 승인 획득	
5-3	데이터정의(Data Definition) 분석작업 견적 확정	데이터 정의 분석, 필드사이즈 등의 분석 작업견적
5-3-1	작업 태스크 항목 설정	
5-3-2	각 단계 작업 견적 할당	
5-4	프로세스 흐름(process flow) 분석 작업 견적 확정	
5-5	프레젠테이션 층(presentation layer) 분석 계획 확정	배치(batch) 출력 프레젠테이션 매체, 배치 입력소스, 온라인 프레젠테이션 매체의 확인과 분류
5-6	데이터 호출(access) 층(layer) 영향평가 계획	데이터베이스와 데이터파일 자산조사에 대한 작업 견적
5-6-1	데이터베이스 자산조사 견적	
5-6-2	데이터 무결성수준(integrity level) 조사 견적	
5-6-3	분석 요약 작업 견적	
5-7	자산조사/분석 작업 계획 작성	최종 조절(tuning)에 의한 작업 계획 작성

Step	Task/Step Name	Notes
6	상세설계	
6-1	물리 시스템 구성요소(physical system components)	JCL, 파일, 데이터베이스, 확인 분류 소스 프로그램, copy books, 실행모듈(executable modules) 화면 정의, 데이터 정의, 통신처리(teleprocess), control tables 등
6-1-1	물리 시스템 구성요소 분류 확인	시스템 명/서브시스템 명 취합
6-1-2	구성요소에 대하여 라이브러리(library) 정보 기록	
6-2	외부시스템(external system)구성요소 확인/분류	모든 문서, 시스템 모델, dictionaries, JCL 등 확인
6-2-1	기능적/기술적(functional/technical) 문서 분류	위치, 타입, 모델 포함
6-2-2	외부시스템 인터페이스 정보 분류	
6-2-3	현 가동정보(production information) 분류	실행(execution), job scheduling
6-2-4	이 단계에서 수집된 모든 항목의 cross	시스템, 서브시스템별 reference 작성
6-3	System-wide 데이터 정의(definition) 분석	모든 레코드(record) 정의
6-3-1	서브시스템별로 모든 소스(source)를 확인/분류	
6-3-2	레코드 레이아웃(record layout) 확인	
6-4	필드 사이즈 확장(field size expansion)에 따른 분석	필드사이즈 확장에 따른 영향을 확인
6-4-1	소스코드(source code)내의 데이터 명 확인	
6-4-2	레코드 레벨 확장(record level expansion) 필요성 확인	레코드(record) scan
6-4-3	필드확장(field expansion) 영향 결정	전 프로그램, 물리파일, JCL, 화면정의, DBMS정의, Tables, 보고서들을 조사/기록
6-4-4	날자 표준화 요건 결정	
6-5	물리 데이터 분석 수행	
6-5-1	물리 데이터 분석 범위 결정	
6-5-2	정확한 데이터 정의, 룰, 제약조건을 확인	
6-6	데이터 정의 분석 결과의 재검토	
6-6-1	전문가의 함께 검토	
6-6-2	데이터 정의 분석을 조정과 재검토	
6-7	관련 시스템 인터페이스를 문서화	
6-7-1	상위 시스템 인터페이스 Data Flow Diagram 작성	
6-7-2	Diagram 조정, 재검토	
6-8	맷치(batch) 출력 프레젠테이션 매체 확인	
6-8-1	각 시스템/서브시스템 맷치 출력 보고서 분석	

Step	Task/Step Name	Notes
6-8-2 6-9	보고서를 출력하여 중복 확인 batch 입력 소스를 분류, 확인	보고서 정보간에 중복 여부 확인 입력부분이 날짜 확장에 의해 영향 받는지 확인
6-9-1	각 시스템/서브시스템 batch 입력 레코드를 분석, 전문가와 함께 정확성 검토	
6-9-2	batch 입력간에 중복 여부 확인	
6-10	온라인 프레젠테이션 매체 분류, 확인	화면에 가는 모든 출력 레코드 확인, 확장
6-10-1	각 시스템/서브시스템 화면 분석	계획 결정
6-10-2	화면간의 중복여부 확인	
6-11	데이터베이스/데이터파일 자산조사 마무리	
6-11-1	물리데이터베이스와 데이터파일 영향평가 자산조사	batch, 온라인 DFD활용 마무리
6-11-2	데이터 정의어(definition language) 소스 자산조사 마무리	
6-12	데이터 인터페이스 전략 개발	얼마나 많은 인터페이스, 무슨 타입이 필요한가
6-12-1	재개발을 위해 확장이 필요한 데이터 결정	이 경우는 날짜필드 확장시에만 필요
6-12-2	어떤 데이터 파일이 입력이고, 어디에 출력 되는지 확인	
6-12-3	데이터 인터페이스 전략 요약	
7	대응계획 마무리	
7-1	어플리케이션 스테이징(staging) 계획 마무리	구성요소를 확인, 작업 라이브러리(working library), 라이브러리 표준 등
7-1-1	작업 지정	
7-1-2	견적 조정	레코드, 프로그램, JCL, 화면, DBMS정의
7-2	필드/레코드 사이즈 확장 계획 마무리	
7-2-1	확장 요구사항 확인	
7-2-2	필드나 레코드 사이즈 확장에 소요되는 작업 시간 산정	
7-2-3	비용 견적	
7-2-4	어플리케이션 전문가와 재검토하여, 최종 작업계획	
7-3	물리 데이터 업그레이드 계획 마무리	파일 점검, 변경, JCL변경, 스페이스(space) 할당 등
7-3-1	작업 TASK 항목 설정과 작업 배당	
7-4	적합성 확인(validation)기준과 계획 설정	
7-4-1	전반적인 적합성 확인 범위 조사	
7-4-2	작업 계획 마무리, 승인 획득	

Step	Task/Step Name	Notes
8	구성요소 확장(component expansion) 업그레이드 실행	데이터 파일/데이터 베이스를 제외한 모든 구성요소의 업그레이드 수행
8-1	필드/레코드 확장 라이브러리 셋업(setup)	
8-2	날자 처리 루틴(routines) 표준화	
8-3	확장되어질 레코드 그룹 검토	
8-4	레코드 그룹 확장	copy내의 레코드 확장, 보고서 형식 변경, 저장소(repository) 경신
8-5	프로그램 level의 필드 확장	각 프로그램 경신
8-6	관련 어플리케이션 객체(objects) 조정	데이터베이스를 제외한 모든 영향요소를 경신
9	어플리케이션 데이터 파일/데이터베이스 경신	
9-1	물리 데이터 무결성 검증	
9-2	물리 데이터 확장/변환	모든 날자 필드와 데이터 수정
9-3	인터페이스 확인	변환되었거나 안된 프로그램의 인터페이스 작성
10	Y2K 변경에 대한 적합성 확인	
10-1	품질 평가 실시	확장한 프로그램 필드, 레코드 레이아웃(layout)의 재검토
10-2	파일, 데이터베이스의 적합성 확인	
10-3	수정한 프로그램의 적합성 확인	
10-4	테스트 실시/완료	
10-5	적합성 확인에 대한 승인 획득	

5. 결 론

Y2K는 단지 정보시스템의 문제만이 아닌 사회 기반시설의 문제이기 때문에, 이에 대한 철저한 조사와 대응책 마련만이 다가올 2000년의 재앙을 예방할 수 있다. 이를 위해서는 체계적인 프로젝트 관리와 방법론, 그리고 도구를 적절히 활용하여야 한다고 생각한다.

Y2K는 시행착오를 허용하지 않는다. 한번 실수하거나 실패하면 그것을 recovery할 시간이 없기 때문이다.

참고문헌

- [1] Unisys, Template 2000 방법론, Sep.,1996
- [2] Unisys, Year2000 Non-IT Infrastructure Methodology, Mar., 1998

- [3] 한국전산원, 컴퓨터 2000년문제 해결을 위한 지침(I), Sep.,1998
- [4] 한국전산원, 컴퓨터 2000년문제 해결을 위한 지침(II), Dec.,1998
- [5] 산업자원부, Non-IT분야 Y2K문제해결을 위한 지침, Dec.,1998
- [6] 한국정보통신진흥협회, 광운대학교, 숭실대학교, 정보기술원, 2000년 문제의 개요와 대응, 1998
- [7] 한국정보통신진흥협회, 국내외 2000년문제 대응, 해결 사례집, Nov.1997



맹철현

1969년 연세대학교 이과대학 지구
시스템과 (이학사)
1983년 고려대학교 경영대학원 MIS
전공(경영학석사)
1981년 정보처리기술사(전자계산
조직용용)

1998년 정보시스템 감리인(한국전산원)
1972년-1998년 한국유니시스 (주) 상무이사
1998년-현재 SEEC Korea 지사장
관심분야 : 소프트웨어공학, ERP, 데이터웨어하우스, MIS

ICAT '99(차세대 네트워크 기술)

- 1. 일 시 : 1999년 5월 18일(화) ~ 5월 19일(수)
- 2. 장 소 : 한국과학기술회관 대강당(강남역)
- 3. 행사내용 : 등록, 초청강연, 논문발표, 튜토리얼
- 4. 행사문의 : 문영성 교수(숭실대)

(02)820-0676