

□ 특집 □

# Y2K 문제 해결을 위한 툴의 기술동향

박 상 서<sup>†</sup> 최 성<sup>††</sup>

◆ 목 차 ◆

- |                      |              |
|----------------------|--------------|
| 1 서 론                | 4 틀 선정시 참고사항 |
| 2 Y2K 해결 툴의 발전단계     | 5 결 론        |
| 3 국내외 Y2K 해결 툴의 기술동향 |              |

## 1. 서 론

Y2K문제가 국내에서도 많은 사람들의 관심을 끌고 정보처리종사자 이외의 일반인조차도 개략적이거나 Y2K의 문제를 알게된 것은 무엇보다도 다행스러운 일이다. 몇년전에 처음 Y2K의 문제에 대해서 인식하게 되었을 때와 비교하면 너무나 큰 차이에 혼란스럽기까지 하다. 당시만 해도 Y2K의 인식을 물론이며 아직도 4~5년 이상이나 남은 문제를 새삼스럽게 벌써부터 들먹일 필요가 있는냐는 식의 대답이 대부분이었으며 심지어는 97년 말까지도 대부분의 정보기술 관련자들조차 “Y2K를 업자들의 공공연한 사업거리”로 여겨져서 편찬 받기가 일수였다. 1996년 말에는 Y2K 문제의 심각성이 매우 클 것으로 판단되어서 세미나의 한 세션을 빌려서 약 2시간에 걸쳐서 Y2K의 전반적인 문제에 관하여 강의한 적이 있었는데 그때의 느낌에서도 “무슨 Y2K에 관한 강의냐?”는 것이었다. 물론 시기적으로는 약간이나마 이른감이 있기는 하였으나 그래도 Y2K에 관해서는 문제점 정도는 전문가로서는 인식하고는 있어야하지 않겠나?라는 것이 필자의 견해였으나 대부분

의 IT관련자들도 귀찮은 듯한 인식이었다. 이미 외국에서는 10여년 전부터 이에 관한 사업거리로 좋은 비즈니스기회로 인식하고 오래 전부터 준비해온 회사가 적지 않았다. 10여년에 걸쳐 오면서 많은 시행착오를 거치면서 Y2K 툴로서의 요건을 갖추면서 준비를 해왔다. 1998년 말부터는 많은 기업이 Y2K의 특수성을 감안하여 사업을 준비해왔으며 비전문적인 투자자가 공격적이며 투쟁적인 Y2K 사업을 벌리고 있는 것을 보며는 일반인에게도 Y2K의 시장이 매우 큰 것으로 보여진다. 그러나, IMF의 영향으로 툴의 사용보다는 정보부서 실무자들은 수작업인 방법으로 해결해 가는 추세여서 많은 Y2K사업자들이 사업을 하나씩 포기하고 있는 것이 실정이다. 늦은 감이 있기는 하지만 Y2K 해결을 위한 툴의 전반적인 것을 알아 보도록 한다.

## 2. Y2K 해결툴의 발전단계

초기의 Y2K Tool(이하 “툴”이라 함)은 상당히 원시적인 수준이었다. 세계시장을 보면 가장 많은 부분을 차지하는 것은 물론 PC나 UNIX가 단연 압도적으로 많을 것으로 보여지나 경제적인 측면에서 보면 대다수의 기업이 투자여력이 있는 기업을 보면 단연 Mainframe부문이며 언어측면에서

† 정 회 원 : 감우씨시스템 기술지원부 컨설턴트

†† 종신회원 : 남서울대학교 컴퓨터학과 교수

보면 아무래도 문제의 여지가 가장 많을 것으로 여겨지는(70 ~ 80년대에 가장 많이 쓰여진)언어는 COBOL이다. 따라서, 개발자들은 가장 시장성이 클 것으로 여겨지는 COBOL과 Mainframe을 대상으로 Y2K tool 개발을 시작했다. 국내에서는 과거 IBM의 정책에 따라 금융권의 대부분이 PL/I를 사용하여서 전세계적으로도 유례가 없는 PL/I 사용 국가로 알려져 있다. 미국내의 툴 개발자들이 한국에서 PL/I이 필요한가? 라고 물을 정도로 미국 내에서는 PL/I이 그다지 사용되고 있지는 않은 것으로 보여진다. 사용자의 입장에서 보면 대개의 기업이 한 개의 언어를 사용하는 경우는 아주 드물어진 것 같다. 한때 Client/server 환경의 급속한 시장 확장으로 인하여 일반 사용자용으로 많은 언어들로 개발되어 있어서 Y2K의 문제를 더욱 곤란하게 만들고 있다. IMF의 영향으로 한 개의 툴 구입마저도 어려운 시기에 여러 개의 툴을 구매한다는 것은 구매 결정자로서는 매우 심한 투동거리를 갖는 것이다. 그리고, 툴의 성능에도 그다지 만족하지도 못하는 것으로 알려져 있다.

### 2.1 Y2K 툴의 변천과정

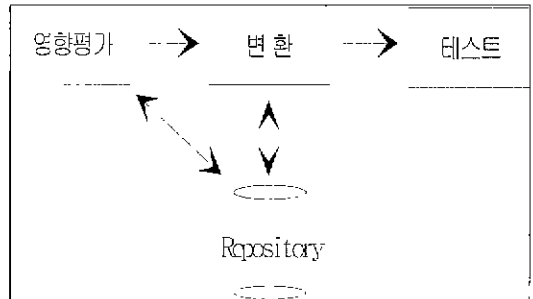
- (1) 초기 : 1990년 대 초반으로 추정할 수 있으며 국내에서도 많은 개발자들이 이 단계를 거쳐서 개발해 왔다. 이 단계의 특징은 대개 언어는 COBOL, PL/I, Assembler 등의 3세대 언어만을 지원하며 탐색 방식도 Scan 방식이 주류를 이루고 있다.
- (2) 중기 : 1995 ~ 1997년으로 추정되며, Client/server용의 언어들이 지원되기 시작했으며 탐색 방식도 Paging방식으로 한 단계 발전했다.
- (3) 말기 : 이 단계 툴들의 특징은 자원관리 기능을 갖추고 있으며 문자 위주의 탐색에서 내부적인 수화알고리즘을 사용하는 툴도 등장하고, Database의 날짜속성을 탐색하여

소스 프로그램과 중복하여 검색하여 정밀도를 더욱더 높이는 상태가 되었다.

### 2.2 툴의 일반적인 기능 분류

Y2K의 방법론에 의하면 5단계로 분류하는 것이 통상적이나 툴의 기능은 일반적으로 크게 3단계로 분류할 수 있다.

〈표1 : Y2K 해결 툴의 기능〉



#### 2.2.1 영향평가 단계용 툴

대부분의 툴들이 이 기능을 지원하고 있으며 성능 면에서 보편은 대동소이한 것으로 보인다. 소스프로그램(서브루틴포함), 카피복(Copy, Include 등의), 실행모듈, 작업실행 명령어등을 검색하여 날짜속성을 찾아낸다. 여기서의 문제점으로 대두되는 것은 바로 날짜 속성을 찾아내는 방법이다. 일반적으로 날짜 속성에 대한 표준화된 속성명(필드명)은 개개인의 명칭부여습성에 따라 다른 것이 일반적이다. 따라서 매우 세심하게 형상관리나 표준화 작업이 진행되었다하여도 일부 프로그래머의 오류로 인하여 Y2K 문제는 잔류하게 된다. 일부 툴에서는 이 날짜 속성명을 지정하는 방법이 상당히 뛰어난 것도 있지만 아무래도 해외에서 개발된 것들이 대부분이고 습성이 다르므로 외래어 표기와 같은 문제는 계속 남아있게 된다. 소위 “패턴파일, Seed File”이라고 불리는 이 날짜 속성명칭을 어떻게 정확히 100%에 가깝게 찾아낼 것인지는 매우 의심스럽다. 필자의 경험으로는

약 96% 가량의 정밀도가 거의 최선이었다. 나머지는 다른 단계에서 발견되는 것이 일반적이다. 이 문제를 보완하려고 수학자들의 아이디어로 만들어진 툴이 얼마 전에 국내에도 소개가 되었으나 구매자의 흥미와 이해를 충족시켜주지 못해 외면 당하고 있는 실정이며 이와 유사한 툴들이 계속 개발되고 있는 것으로 알려져 있다. 국내의 특이한 완벽주의와 비실용주의에 비위를 맞추지 못하는 툴들이 대부분이어서 과연 툴이 국내에 얼마나 많이 보급될까하는 의문이 생긴다. 이 단계에서의 특징은 Library관리 기능이 있다는 것이 또 다른 특징으로 볼 수 있다. 이것은 바람직한 기능으로 2000년 이후에도 계속 사용할 수 있다는 장점이 있어서 구매력을 향상시켜주고 있으며 이 단계에서 수집된 각종 정보는 다음 단계인 변환단계에서 사용되기 위하여 Repository database를 사용하고 있다. 영향평가 단계용 툴의 기능을 요약하면 다음과 같다.

#### (1) Library관리 기능

Copy, Subroutine, 작업실행 명령어, Source program, 실행모듈을 검색하여 관리해주는 기능으로 2000년 이후에도 사용 가능하다. 이중으로 존재하거나 아예 소스코드 자체가 없는 것, Copybook에 존재하지 않거나 중복된 경우, 실행모듈이 복수 개가 존재하거나 없는 등의 일관성이 결여된 경우에 그 정보를 제공하고 Subroutine, Copybook을 사용하는 프로그램을 알 수 있어서 후에 이 Copybook이나 Subroutine을 수정, 폐기등 변경이 있을 경우 관리할 수 있는 정보를 제공한다.

#### (2) 검색방법

날짜속성을 갖는 변수를 검색하기 위한 방법이다.

- 패턴매칭(Seed file) : 일반적으로 많이 사용되는 방법으로 다양한 표현을 위해 Wildcard를 사용하여 표현하는 방법이 많으나 검색에서 누락율이 높은 것이 단점이나 최근에는 많은 프로젝트 경험으로 거의 무시될 정도로 향상되었다. Wildcard

의 사용에 따라서 검색되는 필드의 수가 폭발적으로 증가할 수가 있어서 영향평가 단계에서 변경 유무 판정시간이 매우 길어지고 동시에 복잡해 질 수 있으므로 주의하여야 한다. 패턴매칭 방식에서는 문제가 있어서 툴이 판정하지 못하는 경우(예를 들면, 날짜 format이 정의에서는 YMD 형식이었으나 MDY형식의 필드와 비교, 계산을 행하는 경우)는 모두 확인하게 만드는 경향("False positive"라고 부른다)이 있어서 생각하지 않게 작업량이 늘어날 수가 있다.

- 수학적 비교에 의한 검색 : 외국에서 특허를 받고 단시간에 변환이나 검색을 할 수 있다고 하였으나 검색하는 mechanism을 이용자에게 설명하지 않는 점으로 인해 외면되고 있다. Y2K를 발생시키는 요인이 단순한 날짜 속성필드의 이동이나 동등한 비교는 문제가 되지 않는다고 판단하고 순수하게 오류를 일으킬 수 있는 비교, 계산, 순위 등에 대해 적용하여 Y2K 해결 시간을 획기적으로 단축시킬 것으로 보고되고 있으나 국내에서는 변경에서 Windowing방식의 지원이라는 면에서 그다지 관심을 보이지 않고 있다. 이 방식에서는 문제가 있어서 툴이 판정하지 못하는 경우는 모두 무시하여 작업량을 줄인다.("False negative"라고 부른다)

#### (3) 보고서 작성 기능

보고서 생성은 매우 다양하여 대다수의 툴이 사용자의 요구를 만족하고 있으나 보고서 형태가 국내에 적용되기는 일부 보완해야 할 필요가 있으며 꼭 필요한 보고서는 3종류(소요시간 및 비용에 관한 보고서, 프로그램별 날짜 속성명세서, 날짜 속성명세서)정도이며 나머지는 오히려 프로젝트 시간을 늦어지게 할 수가 있다. 보고서에는 다음과 같은 내용이 있다.

- 영향평가 보고서 : 이 보고서에는 다시 몇 개의 보고서로 구성되어 있다. 변환 방법별 개인별 특성 적용시 소요시간 및 비용 보고서의 상세 및

총괄 보고서, 서브 시스템(프로젝트)별, Library별, Data요소별로 상세 및 총괄 보고서 등이 있다.

- 날짜 속성명 보고서 : Library별, 필드명별, 서브 시스템(프로젝트)별로 보고서가 작성되며 필드명, Line 번호, 길이, 초기값, 사용된 형태(비교, 이동, 계산, 분류, File key, 표기...), 연관된 필드명, 사용언어가 표시된다. 이 보고서가 분석되어서 다음 단계인 변환 단계에서 아주 유용하게 사용될 수 있어야 한다. 한편으로는 Batch형식이 아닌 Online형식으로 동시에 지원이 되어서 변환에 대한 시간을 줄일 수가 있다.

2.2.2 변환 단계용 툴

변환 단계용 툴은 대체로 가격이 고가이며 일부 언어만을 지원하며(대개 COBOL이다) 사용법도 매우 복잡하다. PL/1이나 어셈블러등도 지원된다고는 하지만 기능면에서 보아 부분적으로 지원하며 다양한 경우를 완벽하게 지원한다고 보기는 어렵다. 변환 방법은 다음과 같으며 (표 2)에 비교를 하였다.

- 확장방법(Extended method) : 이 확장방법은 완전성으로 인해 대다수가 선호하나 시간과 비용이 많이 드는 단점이 있다. Database/File에 대해 적용하는 경우 이를 사용하는 응용시스템과의 개발완료/테스트 시기를 맞추어 가며 조절해야하는 문제와 필요에 따라 브리지 프로그램도 작성해야 하는 번거로움이 있다.

- 윈도우창방법(Windowing method) : 서구에서 일반적으로 시간의 단축이라는 점에서 많이 선호되고 있으나 영향평가 단계에서 100년이 경과하는 경우에 대해서는 적용할 수 없고(특히 우리나라에서는 분석에 소요되는 시간을 충분히 할당하지 못하는 문제나 환경의 제약) 어플리케이션 프로그램을 수정해야하는 단점으로 기피하는 경향이 있으나 일반적으로 툴이 이 방법만을 지원하는 경우가 대다수이다.

2.2.3 테스트용 툴

Y2K에서 가장 어려운 문제로 꼽히는 부분으로 대다수의 툴이 지원하지 않으며 일부가 사용자

(표 2) Y2K 변환방법의 장단점 비교

접근 방식	장 점	단 점	선택기준
확장방법 (데이터 기초방식)	<ul style="list-style-type: none"> <li>o 영구적이다.</li> <li>o 알고리즘변경 거의 없음</li> <li>o 향후 유지보수비용 절감</li> <li>o 프로그램 일관성유지</li> </ul>	<ul style="list-style-type: none"> <li>o 데이터 변환은 필수</li> <li>o 부가적인 저장소가 필요할 수 있음</li> <li>o 스크린, 보고서가 수용하기 어려울 수 있음</li> <li>o I/O 증가로 추가적 튜닝이 필요할 수 있음</li> </ul>	<ul style="list-style-type: none"> <li>o 대부분의 경우 선택(시간과 비용에서 가능할 경우)</li> </ul>
윈도우창방법 (프로세스기초방식)	<ul style="list-style-type: none"> <li>o 데이터 변환이 필요없음</li> <li>o 변환 시간이 비교적 짧음</li> <li>o 비교적 적은 비용투입</li> </ul>	<ul style="list-style-type: none"> <li>o 프로그램 로직변경 불가피</li> <li>o 에러발생 가능성높음</li> <li>o Sort기능 수행 불가</li> <li>o 최대100년 범위내 가능</li> <li>o 향후 유지보수 비용 높음</li> <li>o 프로세스의 일관성상실 우려가 있음</li> <li>o 철저한 시험 필요</li> </ul>	<ul style="list-style-type: none"> <li>o 시간/비용 부족시</li> <li>o 업그레이드 계획이 있고,</li> <li>o 가장 오래된 데이터가 2000년 기준100년이 되지 않고 100년안에 업그레이드계획이 있을 경우(가령 가장 오래된 데이터가 1910년일 경우 2010년까지 업그레이드 되어야함.)</li> </ul>

환경을 만들어 주기는 하지만 기능이 매우 약하다. 여러 단말기에서 하나의 응용 프로그램을 동시에 다른 날짜로 테스트하여야 할 경우가 대부분이나 테스트용 툴은 전체(Batch나 Online전체)가 한 날짜에만 국한되는 경우가 대부분이다. 그나마 이 툴이 없다면 응용 프로그램을 시스템날자를 획득하는 부분을 comment문장으로 만들어 놓은 뒤에 초기 값을 주는 방식으로 테스트하지 않으면 안 되는 번거로움이 있다. 한편으로 Y2K 전문툴보다는 오히려 Database/File Utility성의 상용툴들이 더 많이 팔리는 기현상을 보였다. 일반적으로 응용 프로그램의 변경은 수작업으로 행하여졌으나 Database 시스템의 효율적인 내용확인을 위해서는 이런 툴들이 선호되어진 것이다. IBM mainframe에서의 응용프로그램의 테스트는 대개 SVC10(Supervisor call 10번)이 많이 쓰여지는 관계로 이 기능은 대부분이 지원하나 시스템프로그램인 IMS/DC나 CICS등의 online system이나 DB2등의 database에서는 STCK(Sotre clock)을 사용하고 있는데 이 기능은 일반적으로 지원하지 못하는 것으로 보여진다. 기술적인 어려움과 명령어군의 확장으로 인해 응답속도가 늦어진다.

### 3. 국내외 Y2K 해결 툴의 기술 동향

#### 3.1 해외공급사(툴 및 해결방안 공급 회사)

- 변환서비스제공 : Alydaar Software Corp.  
Peritus Software Services Inc.  
Cogni-CASE.
- DATE ROUTINES : Trans Century Data System.
- 분석 툴 : Revolve(Micro Focus)  
COBOL Analyst(SEEC Inc.)  
System Vision  
Year2000(Adpac Corp)  
ESW(Viasoft)  
Discover(Software Emancipation

Technology Inc)

- 프로그램 변환 툴 : COBOL Workbench(Micro Focus)
- 데이터 변화 툴 : File-Aid(Compuware Corp)  
Century Conversion Software  
(Quintic Systems Inc)
- 테스트 툴 : VIA/Smarttest & Validata(Viasoft)  
Tictor(Isogon)  
Xpediter & Xchange(Compuware Corp)
- 컨설팅 제공 : Micro Focus, EDS Andersen Consulting

#### 3.2 국내 공급사

1998년 초반에 국내에서는 툴을 영업적인 목적으로 판매나 용역을 하려는 업체의 수가 약 30여사 이내였으나 1998년 말에는 80여사 이상인 것으로 업계에서는 알려졌다. 한국전산원에 등록된 업체 이외에도 미등록된 상태로 영업활동을 벌이는 기업이 상당수가 있는 것으로 알려지고 있다. 그러나 IMF의 영향으로 툴구입의 기피로 말미암아 1999년 초 현재 많은 초기의 선두주자들이 이 Y2K 사업을 포기하고 있다. 대부분이 IBM위주이며 고가이고 IMF로 인한 환율상승으로 국내에서는 더 이상의 영업적인 의미가 상실되었기 때문이며 일부 PC에서 가동되는 툴만이 겨우 명맥을 유지하고 있는 실정이다. Host로부터 응용 프로그램을 PC로 Down받아서 작업을 마치고 다시 Up해야 하는 불편함에도 불구하고 가격이 저가라는 이유로 선호되고 있다. 가격 면에서도 1997년에 비해 1998년에는 100% 이상 인상되고 1999년에는 200% 이상 인상될 것이라는 경고에도 불구하고 많은 국내의 기업을 이것을 받아들이지 않고 있으며 해외 툴 개발사에서는 “대한민국은 2000년 1월 1일부터 장사가 시작될 것(?)”이라는 비아냥거림도 감수해야했다. 다행히 국내에서도 성능이 좋다고 평가되는 툴들이 속속 개발되어 중소기업을 운영

하는 기업가에게 한 가닥 희소식이 되고 있다.

(표 3)에 국내에 공급되는 Y2K 툴을 비교하였다.

(표 3) Y2K 툴 비교

업체명	툴명	운영환경	지원언어	지원언어			
				당장 평가	변환	검증	시험 유형
CNV시스템(주)	• Platform2000 Century Source Conversion SW	IBM/MVS, VSE/AS400, Tru6400	COBOL, RPG 외 모든언어	✓			
	• Platform2000 Century File Conversion SW	IBM/MVS, VSE/AS400, Tru6400	모든 DB		✓		
	• Platform2000 Quince Plus	IBM/MVS, VSE/AS400	COBOL, RPG		✓		
김우시스템	• SURVEY2000	MVS, OS/390	COBOL, PL/I, ASM 외 모든언어	✓			
	• TRAASLATE2000	MVS, OS/390	COBOL, 카타인어		✓		
	• SIMULATE2000	MVS, OS/390	모든언어			✓	✓
기린기술(주)	• Nice Dates	제한없음	COBOL				
에스도인포텍	• B12K	PC, HOST	COBOL	✓	✓	✓	✓
발성컴퓨터 소프트웨어	• SAFE2000/CBL-TRANS	제한없음	ANSI-COBOL		✓		
	• SAFE2000/CBL-VARIFY	제한없음	ANSI-COBOL			✓	
	• SAFE2000/CBL-ANALYSIS	제한없음	ANSI-COBOL	✓			
상원시스템	• 신원칼라2000 (COBOL)	PC, LINUX, 메인프레임 Windows NT	COBOL	✓	✓		
	• 신원칼라2000(CPF)	PC, LINUX, 메인프레임 Windows NT	CPF	✓	✓		
메이스	• YES2000	Windows 95/NT	COBOL	/	✓	✓	
디커크리아 (리콜의 이집트, 유니 차원인기별로 다른 포함)	• UNRAVEL2000	Windows 95/NT	C++, VB,	/	✓		
	• UNRAVEL2000	Windows 95/NT	Power Builder, Shell Script	/			
	• UNRAVEL2000	Windows 95/NT	ORACLE	/			
	• UNRAVEL2000	Windows 95/NT	COBOL	/	/		
	• UNRAVEL2000	Windows 95/NT	SQL	/	✓		
• UNRAVEL2000	Windows 95/NT	FoxPro, JAM, Perl	/	/			
플루리다리아	• Smart Editor	MVS/ESA, CICS, DB2, Windows NT, NOVEL	ASM, COBOL, PL/I, SHS, FOCUS, ADS/O	/			
	• Calendar&Dodge Routines	MVS/ESA, CICS, DB2, Windows NT, NOVEL	ASM, COBOL, PL/I, SHS, FOCUS, ADS/O		/		
	• Calendar Date Routines	MVS/ESA, CICS, DB2, Windows NT, NOVEL	ASM, COBOL, PL/I, SHS, FOCUS, ADS/O		/		
	• Comperator	MVS/ESA, CICS, DB2, Windows NT, NOVEL	ASM, COBOL, PL/I, SHS, FOCUS, ADS/O			/	

업체명	툴명	운영환경	지원언어	지원언어			
				당장 평가	변환	검증	시험 유형
송우정보(주)	• SoftFactory2000 - Resolve2000 SmartFind - SmartFix	Windows 95/NT	COBOL, PL/I, ASM	/	✓		
	• WorkBench2000	Windows 95/NT	COBOL, PL/I, AS4, CICS, IMS, DB2		/	✓	
	• PL/I WorkDench	Windows 95/NT	COBOL, PL/I, VSM, CICS, IMS, DB2		/	/	
	• Mainframe Express	Windows 95/NT	COBOL, PL/I, AS4, CICS, IMS, DB2		/	/	
• TestSuite	Windows 95/NT					/	
유니가이브 시스템	• Change Man	MVS/AA, ESA, OS/390	ASM, BMS, COBOL, COLBOL2, CAP, APS, PL/I	✓	/	/	
	• CDF(Concurrent Development Facility)	MVS/AA, ESA, OS/390, FACOM OS IV/F4 MSP	ASM, BMS, COBOL, COLBOL2, CAP, APS, PL/I	/	/		
	• StarTool	MVS/AA, ESA, OS/390, FACOM OS IV/F4 MSP	ASM, BMS, COBOL, COLBOL2, CAP, PL/I	✓	✓		/
• COMPARTX	MVS/AA, ESA, OS/390, FACOM OS IV/F4 MSP	MVS/AA에서 사용가능한 모든 언어				✓	
캐니시스템	• Change Man	Windows 95/NT, UNIX	C, C++, COBOL, VB FORTRAN, ADA81	✓			✓

#### 4. 툴 선정시 참고사항

##### 4.1 예상소요비용 산정방법

- 산출기준은 가트너 그룹에서는 라인당 \$05~\$1.0이상(프로그램당 \$450~\$600)으로 계산하고 있으며, 앤더슨 컨설팅에서는 100만 라인당 2~4인/년 이상으로 산정하고 있다.
- 단계별 소요비용은 문제의 인식에 15%, 분석에 10%, 수정에 30%, 단위 테스트에 25%, 통합 테스트에 20%로 배분한다.

##### 4.2 툴 선정시 유의사항

분석 및 변환 작업이 이루어지는 위치에 따라 메인프레임용 툴과 PC용 툴로 구분할 수 있는데,

일반적으로 호스트 컴퓨터의 부하 증가를 방지하고, 응용 프로그램의 DOWN-TIME을 줄이며, 나아가 GUI 등 다양한 기능들을 활용하기 위해서는 PC 로 다운로드하여 처리하는 방법이 보다 유리하다고 할 수 있다. 툴은 프로그래머의 생산성 향상은 물론, 나아가 변화작업 전체의 성패까지 좌우할 수도 있다. 따라서 선정시에는 다각적인 검토가 필요한데, 몇가지 유의사항을 살펴보면 다음과 같다.

- 툴의 종류
- 가격 및 기능 범위
- 사용의 편의성 및 친숙도
- 추후 활용성
- 시스템 부하 증가량
- PL/I의 지원 여부

### 4.3 검토 및 실행 단계의 착안사항

- 수정작업을 수행할 인력자원은 충분한가?
- CPU의 부하량은 추가적인 수정작업을 감당할 수 있을 만큼 충분한가?
- 대용량의 파일이나 데이터베이스 처리를 위하여 디스크 용량은 충분한가?
- 응용프로그램의 많은 부분이 변경될 경우 테스트 작업은 어떻게 수행한 것인가?
- SAS, FOCUS와 같은 외부 툴로부터의 입력은 어떻게 처리하고 테스트할 것인가?
- PC에 있는 사용자의 데이터의 및 프로그램 수정은 누가 지원할 것인가?
- 버퍼 크기와 맞지 않는 화정된 레코드로 인하여 발생할 수 있는 실행속도 문제는 어떻게 대처할 것인가?

## 5. 결 론

초기의 툴들은 영향평가단계, 변환단계, 검증단계(테스트)단계에서 지원되었으나 근래에 와서는

Database/file내의 데이터의 확인 및 변환처리, 테스트용 일자나 data의 생성도 지원하는 툴들이 등장하고 있다. 이러한 툴들은 대개 자체의 방법론도 보유하고 있는 것이 대부분이나 가격이 대체로 비싼 편이다. 대부분의 툴들이 mainframe이나 수요자가 많은 중간급 시스템을 목표로 개발되었기 때문에 특별한 용도의 언어나 하드웨어적 환경을 갖는 기업은 이러한 툴제품이 거의 없거나 기능이 매우 미약하므로 기업 스스로 해결책을 강구해야하는 문제점이 있다. 국내에서의 툴 적용은 99년도 7월 이후에 중소기업들을 중심으로 한 차례의 변환을 대상으로 하는 툴구입이 예상되며 검증을 위한 툴 이용은 4월 이후에나 본격적으로 시작될 것으로 보인다. 인증문제에 있어서는 검증 자체가 형식적인 인증서를 위한 것으로 여겨지는 경향이 있어서 많은 기업으로부터 외면 당할 것 같은 예감이 들며 오늘도 Y2K사업이 매우 큰 수익이 될 것으로 여기고 있는 많은 비전문적 투자자들이 생겼다가 없어지는 반복을 되풀이하는 옷지 못할 일은 그만 중지되기를 바라며, 다행히 성능이 우수한 국산용 툴이 보급되어 2000년이 되어서도 Y2K의 피해를 당하는 기업이 하나라도 줄고 경제적인 뒷받침으로 툴을 사용하여 피해를 최소화되었으면 한다.

### 참 고 문 헌

- [1] 한국전산원, 컴퓨터2000년 문제해결을 위한 지침(I), 1998년 8월
- [2] 목전의 '서기200년' 혼란과 대책, 정보화 사회 특집, 정보통신진흥협회, 1997.1
- [3] '컴퓨터 2000년 연도표기 문제 현황과 대처 방안', 정보화사회로가는 길, 정보문화센터, 1998.5
- [4] 김숙자 역, "2000년 컴퓨팅의 해결책", 성안당, 1996.9.

- [5] 한국전산원, “'95년도 국가기간전산망 추진실적 평가”, 1996.8.
- [6] 한국전산원, 한국정보통신진흥협회주최 세미나 자료, “제1차 2000년 연도표기문제 세미나”, 1997.5.
- [7] Brian Kahin, Janet Abbate, “standards policy for Information Infrastructure”, 1995.
- [8] GAO(United States General Accounting Office), “Year2000 Computing Crisis:An Assessment Guide”, 1997.2.
- [9] “ISM Year 2000 Draft”, 1997. 2.
- [10] William M.Ulrich, Ian S. Hayes, “The Year 2000 Software Crisis”, Yourdon Press Computing Series, 1997.
- [11] <http://www.itpolicy.gsa.gov>의 2000년 관련 자료
- [12] <http://infosphere.safb.af.mil/~jwid/fadl/world/fedguide.html> (SSA자료)
- [13] 정보처리전문가협회, ‘관리자세미나자료 SDS 발표’, 1998.7
- [14] 정보처리전문가협회, ‘관리자세미나자료 UNISYS 발표’, 맹철현, 1998.3
- [15] 정보처리학회지, 최 성, ‘Y2K(밀레니엄버그)’, 98년도 9월호.



**박 상 서**

1982년 단국대학교 화학공학과  
 1982년 홍국생명 전산실  
 1987년 현대전자 통합운영부  
 1989년 지구어시스트 기술지원부  
 1995년 정보처리기술사  
 1996년 잡우씨시스템 기술지원부

현) (사)정보통신 기술사협회 정보보호위원회 부위원장  
 관심분야 : ITS(지능형교통시스템), 소프트웨어 품질 평가, 컴퓨터구조



**최 성**

1983년 연세대학교 산업대학원 전자계산 (공학석사)  
 1980년 고려대학교 경영대학원 경영정보 (경영학석사)  
 1999년 강원대학교 대학원 전자계산전공 박사 수료

전) 제주은행 전산실장, 한국생산성본부 OA추진사무국장 역임  
 현) 남서울대학교 컴퓨터학과 교수, 한국정보처리전문가협회 총무이사  
 관심분야 : 영상게임, 전자상거래, 소프트웨어엔지니어링, MIS/OA, 기술경영