

2000년 문제의 본질 및 중대성

문제해결을 위해서는 제일 중요한 것이 사항의 심각성을 빨리 인식해야 한다는 것과 시간 관리를 철저히 해야 한다는 것이다. 심각성을 인식하는 순간부터 2000년 이전까지의 어떤 방법으로 해결할것인가, 어떤 틀을 이용해서 해결할거라는 생각을 버리고 우리 조직이 이런 문제를 해결하기 위해서 문제를 해결하기 위한 과정에 대한 계획을 세우고, 과정을 진행하면서도 피드백을 통해서 철저히하게 관리를 해야 한다는 것이다.

조 증 성 동서대학교 산업공학과 교수

‘2000년 표기문제’란 60년대, 70년대 처음 컴퓨터시스템이 도입될 때 디스크 공간과 칩용량의 한계로 인해서 연도표기를 2자리로 하자, 또는 다른 이유도 있겠지만, 우리가 연도 입력을 할 때 2자리로 하면 키스트록을 줄일 수 있지 않는

보면 01/01/00으로 표기하게 된다. 다음은 2000년 표기문제의 두자리 표기로 인한 연산왜곡처리의 한예로서 날짜항목의 대소비교가 왜곡된 예로 한부동산에 관한 경과 데이터를 날짜순으로 소팅하여 프린트한 경우이다.

서기번호 4자리로 정확히 처리		
날짜	데이터	
19000104	구	입
19000106	소유권	이전등기
19911201	저당권	설정등기
20000105	저당권	말소등기

서기번호 2자리로 잘못 처리		
날짜	데이터	
000104	구	입
000105	저당권	말소등기
000106	소유권	이전등기
991201	저당권	설정등기

가하는 생각에서 서구에서 쓰는 컴퓨터 연도표기 방식이 도입이 되어서 사용되면서 비롯된 것이다.

아주 사소한데서 출발한 문제가 엄청난 부담을 주고 있다는 것이다. 연도표기를 사용하다 보면 문제에 봉착하는 시점이 2000년 1월 1일이라는 것이다. 2000년 1월 1일 기존처럼 표기하다

이런 기본적인 컴퓨터의 연산의 왜곡으로 인하여 관련된 모든 정보들이 왜곡이 될 수 있다는 것이 심각하게 대두될 수 있다는 것이다.

다음은 뺄셈으로 인하여 나타나는 연산왜곡처리의 예이다. 1990년 4월 1일 입사한 사원의 2001년 4월 1일까지의 근속년수를 계산한 경우이다.

서기번호 4자리 처리를 하였을 경우(200010401-19900401)/10000 = 11로서 11년인데 이것을 서기번호 2자리로 계산하였을 경우에는 (010401-900401)/10000 = -89로서 -89년이라는 우스운 결과가 나오게 된다.

마찬가지로 이문제만 보게되면 간단히 해결될 수 있지만, 이와같은 왜곡처리된 데이터가 들어 있는 잘못된 정보들이 엄청나게 많이 들어가 있을 수 있다는 것이다. 문제의 본질은 첫째는 영향범위가 굉장히 광범위하다는 것이다.

어떠한 형태의 모든 컴퓨터, 시스템들이 영향을 받을 가능성을 내포하고 있다. 날짜데이터를 기본으로 쓰는 시스템은 모두 영향을 받을 수 있다는 것이다. 둘째는 연장이 불가능한 2000년이라는 기한이 존재한다는 것이다.

이는 문제가 시스템의 틀 자체 근본에서 기인함으로 이 문제의 대응기회를 놓칠 수 없다는 것이다. 세번째로는 전세계적으로 동시다발적으로 발생하는 문제라는 것이다.

영향범위가 넓기 때문에 나 혼자만이 해결하였다고 그 문제가 해결된다고는 할 수 없다. 네번째로는 기업간 또는 이용자간, 나아가서는 국가간의 시스템의 날짜항목에 대한 대응방식과 시기조정이 필요할 것이다. 다음은 다양한 사용자들의 다양한 요구를 만족시킬 수 있는 획기적인 해결책을 기대하기 어려울 것이다.

마지막으로는 책임소재의 애매성이다. 시스템의 디스크 용량을 절약한다든지 입력하는 데이터 키스트록을 줄이기위해서 2자리로 입력을 해온 이 문제해결을 위해 이제와서 누가 책임을 질 것인가이다.

문제의 영역을 다섯가지로 나뉘보면 HW적인 문제와 시스템 SW, 어플리케이션, 데이터, 주변장치 등으로 유형분석이 되고있다. 국내의 시스

템의 사용환경을 보면 주로 시스템SW는 외국에서 도입해서 쓰는 입장이기 때문에 시스템 SW는 어느정도 시스템 SW를 개발하는 사람들에 의해서 어느정도 해결책이 나오리라고 생각된다.

그러나 어플리케이션이나 어플리케이션에 의해서 생성되고 처리되는 데이터와 같은 우리 국내에 상당히 많이 개발되고 저장되어 있는 상태이다.

어플리케이션과 데이터의 비용과 난이도를 보면 어플리케이션이 중간정도 되지만 비용이 엄청나다는 것이다.

이것은 미국에서 대략적으로 예측을 했을때 조직에 있는 어플리케이션의 2000년문제를 해결하기 위해서 드는 비용이 한프로그램 라인당 35센트에서 40센트정도 소요된다는 추산이 나왔다.

자신이 속한 조직에서 어플리케이션의 프로그램 라인이 몇 개인지를 생각해 보면 비용을 산정할 수 있을 것이다.

데이터의 비용은 중간정도되지만, 데이터 부분은 처리하기가 상당히 까다롭다고 되어있다. 국내의 환경을 보면 어플리케이션과 데이터부분에 2000년 표기문제는 국가차원에서 심각하게 대처를 해야 할 것이다.

과급효과와 난이도는 독립된 시스템의 경우는 낮지만, 기업에서 국가로 나아가 부피가 커질수록 크고 복잡해 진다.

문제의 심각성을 더하는 것은 인식이 부족하다는 것이다. 누군가가 해결할 것이며, 나의 책임이 아니다라고 생각하는 것이다. 그리고 2000년 1월 1일까지 이 문제를 해결해야 하는 시간의 제약이 있다는 것이다. 그리고 많은 비용이 든다는 것이다.

미국에서 추산할 때 포춘지에 들어가는 50대의 기업들이 약 300~400억달러의 비용이 든다

는 것이다. 그리고 인력자원의 문제이다.

응용프로그램의 예를 든다면 그 응용프로그램을 일일이 분석해서 낱짜표기문제를 수정할 때 엄청난 인력이 소요된다는 것이다.

이에 자체내에서 이런 문제들을 해결할 수 있는 인력을 갖추지 못하면 시간이 지날수록 비용도 점점 상승하고, 인력스카우트도 점점 심각해질 거라는 예측도 할 수 있다. 2000년에 하루하루 다가갈수록 이런 문제에 봉착하게 될 것이다. 미연방회의 '컴퓨터 2000년 위기문제' 특별공청회에서는 이 "인력스카우트의 문제는 비용이나 기술로 계산할 수 없는 특수한 난제가 있다."라고 발표되었을 만큼 아주 심각하다.

2000년 표기문제해결의 경험과 예측을 살펴보면 다음과 같다.

회 사	추진내용
가트너 그룹	- 전세계적으로 3000~6000억 달러 소요 - 비용증가로 20% 정도의 회사가 문닫음
NY 증권 거래소	- 1987년에 시작 - 100명이상의 프로그래머 투입 - 약 2,900만 달러 소요
Owaha Bank	- 1993년 약 25명으로 시작 - 1,000만 달러 비용이 들었는데 25% 진척
미국 텍사스주	- 영향을 받을 코드가 154백만 라인으로 예상

문제해결을 위해서는 제일 중요한 것이 사항의 심각성을 빨리 인식해야 한다는 것과 시간관리를 철저히 해야 한다는 것이다. 심각성을 인식하는 순간부터 2000년 이전까지의 어떤 방법으로 해결할 것인가, 어떤 툴을 이용해서 해결할 거라는

생각을 버리고 우리 조직이 이런 문제를 해결하기 위해서 문제를 해결하기 위한 과정에 대한 계획을 세우고, 과정을 진행하면서도 피드백을 통해서 철저하게 관리를 해야 한다는 것이다.

그리고 최고 경영층의 이해와 지원을 촉구해야 한다. 실무담당자가 최고 경영층에 있는 분들에게 문제의 심각성을 알리고 이해와 지원을 받아야 한다. 국가에서도 적극적으로 대응을 해야 하고 필요하면 예산 지원도 해야 한다.

결론은 심각한 문제의 본질이 있기 때문에 하나의 솔루션이나 툴을 가지고 해결하기 보다는 문제를 해결하기 위한 과정을 잘 세우고, 수행하는데에도 철저하게 수행된 결과에서 나오는 것들을 피드백 받아서 프로세서를 진행하는데 관리를 해야 한다.

그리고 문제의 심각성을 인식하는 그 순간에 바로 시작을 해야 한다는 것이다. 개인이든 조직이든 이 문제의 대처방안을 세워야 한다.

경험이 있는 분이나 해결책을 생각하는 사람들의 도움을 받아야 한다는 것이다. 이 문제의 심각성을 인식하고 있는 사람들끼리 서로 정보를 공유하는 일도 중요한 일이다. ●

