



經營管理者를 위한 컴퓨터의 脈

金 麗 暉*

目 次

- | | |
|-----------|------------------|
| 1. 머리말 | 3. SYSTEM 開發 |
| 2. 컴퓨터 導入 | 4. SYSTEM 의 補修維持 |

1. 머리말

날이 갈수록 우리는 컴퓨터라는 尖端産業器機에 자신도 모르게 친숙해져 가고 있다. 銀行의 온·라인預金, 각종 公課金 告知書, 컴퓨터세탁기, 컴퓨터냉장고 등이 생활과 직결되어 있으며 證券시行情報, 不動産정보문의 그리고 올림픽행사까지 컴퓨터시스템의 작동상태에 따라 成功與否가 판가름되어질 정도로 컴퓨터는 우리생활에 깊숙히 適用되고 있다.

컴퓨터가 탄생된지 40여년이 지난 지금의 컴퓨터는 迅速·正確 그리고 大量處理가 가능한 電子두뇌로 진보하면서 사회 곳곳에 情報革命이라고 불리우는 혁신의 바람을 불어일으키며 工業사회로 부터 情報化사회로 접어들고 있다.

인류가 산업혁명으로 새로운 공업사회를 이룩해 인간의 육체적限界를 工業機械로 극복하였드시 情報革命은 인류에게 새로운 두뇌와 신경계통을 제공하여 인간의 정신적영역을 확장하고 時空間을 초월하는 힘을 부여하게 될 것이다.

이처럼 컴퓨터의 發展이 물고을 情報化사회로의 물결은 이제 더이상 거역할 수 없는 시대적인 조류로서 기업은 물론 개인의 가정생활도 새

로운 情報化器機(new media)에 친숙할 수 밖에 없는 상황에 다다른 것이다.

그런데 현재의 컴퓨터는 단순히 누구나 마음대로 사용할 수 있는 萬能기계가 아니다. 즉 컴퓨터의 價値는 어떤사람이 어디에 어떤 方法으로 사용하는가에 따라 그機能과 性能이 크게 差異가 난다는 것이다.

컴퓨터를 効果적으로 活用하기 위해서는 그에 따른 충분한 사전준비와 올바른 理解가 先行되지 않으면 안된다. 특히 기업경영의 운명을 짊어진 經營관리자는 더욱더 컴퓨터의 올바른 知識이 필요한 것이다.

컴퓨터活用方法을 알고 있는 經營관리자와 그렇지 못한자와는 마치 병사가 싸울 무기를 갖지 않고 맨몸으로 전쟁터에 나가서 무기를 가진자와 대적하는 경우와 같은 것이다. 이제 컴퓨터는 알기를 원하는 원하지 않은 간에 기업경영에 새로운 무기로써 받아들이지 않으면 도태되어 버릴 수 밖에 없음이 自明해지고 있다.

그런데 기업경영자가 컴퓨터에 관한 知識을 얻기 위해서 그에관한 서적이나 資料를 찾을 경우 대부분의 컴퓨터 서적이 컴퓨터 專門人을 위해 집필한 전문서적이어서 이러한 책자를 통해서 經營관리자가 필요한 資料가 쉽게 구해져

* 情報處理技術士(電子計算機組織應用).
韓國證券電算(株) 市場시스템部長

기 어려운 실정이다.

이에 필자는 경영관리자가 기업경영에 필요한 컴퓨터도입 절차나 사전준비상황 그리고 효과적인 소프트웨어 개발과정을 차례로 설명하고 운영업무의 대상과 시스템의 효과적인 補修維持를 설명함으로써 기업경영에 도움이 되고자 한다.

2. 컴퓨터 導入

컴퓨터를 도입하려면 전산화계획을 수립하기 전에 반드시 컴퓨터도입의 妥當性여부를 충분히 검토할 기회를 가져야 한다.

妥當性 調査는 豫備調査 또는 基礎調査라고도 하는데, 현재 처리되고 있는 업무의 문제점과 전산화 이후에 예측되는 문제점을 정확하게 分析하고 경영자의 경영정책과 사용자의 要求事項을 定義하고 각종 制約條件을 감안하여 현실적으로 전산화가 可能한가를 검토하고 투입되는 費用에 비해 어떤 効果가 있는지를 판단하여 전산화의 妥當性 여부를 판정해야 한다.

그리고 전산화의 妥當性이 인정되면 전산화 (computerize) 계획을 수립하게 되는데 이때 검토해야 할 事項이 별표 1 과 같다.

[별표 1]

- 電算化(computerize)할때 검토해야 할 事項**
- I. 전산화 妥當性 調査(feasibility)
 - II. 전산화 계획수립시 考慮해야 할 對象
 - III. 어떤 컴퓨터를 導入해야 하나?
 1. 컴퓨터 도입計劃 및 節次는?
 2. 컴퓨터를 購買할 것인가? 賃借할 것인가?
 3. 機種 선정시 무엇을 검토해야 하나?
 4. 컴퓨터 賃借契約時 착안 사항은?
 5. 컴퓨터 導入時 經營者는 무엇을 알고 있어야 하나?
 6. 컴퓨터 設置時 事前에 준비할 사항은?
 - IV. 소프트웨어 (software)는 어떻게 開發해야하나?
 1. 直接 개발할 것인가? 外注 用役을 쓸 것인가?
 2. 시스템 定義와 개발 計劃(system definition & project plan)
 3. 소프트웨어 開發時 check point 는?
 4. 소프트웨어 生産管理는 어떻게 하나?
 5. 外注計劃時 무엇을 考慮해야 하나?
 6. 소프트웨어 開發費는?
 - V. 시스템 運用(operation)은 어떤 방식이 좋은가?

1. 直接 運用할 것인가?
2. 시스템을 효과적으로 運用管理하려면?
3. 電算器機 및 附帶施設의 區劃은?
4. 컴퓨터 관련 職種에 근무하는 사람들은?
5. 누가 얼마를 만나?
6. 시스템 運用費와 運用費 項目은?

그리고 전산화 計劃수립시 고려해야할 對象은 별표 2와 같이 制度改善, 組織개편, 要員확보, 資金계획, 하드웨어, 設備계획, 소프트웨어, 프로젝트 管理등이 綜合的으로 검토되어야 한다.

[별표 2]

- 전산화 計劃 수립시 考慮해야 할 對象**
1. 制度
 - 현황, 문제점, 개선안, 대체안, 최적인 선정 제도개선, 조직 및 규정변경
 2. 하드웨어(hardware)
 - 기능, 성능, 가격, 구성(configuration), 단말기, 기종선택, 도입시기, 설치준비
 3. 設備
 - 전원설비, 공조설비, 통풍시설바닥(floor-access) 통신시설, 보안대책 등
 4. 소프트웨어(software)
 - 목적, 대상, 적용 범위, 필요성, 효과, 자료 처리량, 처리期限, 제약조건, 신뢰성, 보수용이성
 5. 人力
 - 요원, 지식, 기술, 경험, 직무권한, 요원배치, 휴가·병가, 건강관리
 6. 移行·運用
 - 신규전표, 서식, 원장등록, 오퍼레이션(operation), 보수유지(maintenance)
 7. 經費
 - 자료조사비, 개발비, 외주용역비, 이행비, 교육훈련비, 운용비등 最適經費산출
 8. 教育·訓練
 - 개발요원, 외주요원, 운용요원, 현업담당자, 관리자 교육
 9. 電算委員會
 - 제도개선, 조직 및 규정변경, 하드웨어, 소프트웨어, 경비, 가동시기등 重要사항 意思決定
 10. 프로젝트(project)管理
 - 예산, 일정계획, 요원계획, 진척관리, 비용관리 외주관리, 설비계획, 이행계획, 운용계획, 보수유지계획, 확장성 검토

컴퓨터 導入節次는 별표 3과 같이 妥當性調

査로부터 전산實務班의 시스템도입계획이 작성되고 전산委員會의 承認을 거쳐 컴퓨터 도입계약이 체결되며 컴퓨터를 導入設置하게 된다.

[별표 3]

컴퓨터 導入計劃 및 節次	
1. 전산화 妥當性 조사(feasibility)	
2. 시스템 도입 計劃 작성—(전산 實務班)	
a. 업무개발계획	f. 설비계획
b. 일정계획	g. 이행·운영계획
c. 요원확보계획	h. 교육훈련계획
d. 조직계획	i. 확장성 검토
e. 예산(비용견적)	j. 의주계획
3. 전산 委員會 承認	
—전산실무반이 작성한 각종 계획을 修正補完 후 승인	
4. 컴퓨터 도입 提案 요청	
—제안서 설명회 개최	
—제안서 접수 마감	
5. 機種 선정 및 평가	
—기종 평가 방법 결정	
a. POED(중요항목을 정량화하여 평가)	
b. GIBSON MIX(자주 쓰이는 명령의 실행시간 비교)	
c. AUERBACH(표준문제를 해결하는데 소요되는 시간 비교)	
d. BENCH MARK(사용자가 만든 표준문제에 대해서 maker 측의 실행 결과 비교)	
e. CHECK LIST(항목별로 채점하여 상대적으로 비교)	
6. 전산 委員會 및 理事會에서 기종 選定 承認	
7. 컴퓨터 도입 契約 체결	
8. 政府에 컴퓨터 도입 승인 신청……해당기관	
9. 附帶設備 및 컴퓨터 설치 準備	
10. 컴퓨터 導入 設置	

이때 경영자가 알고 있어야 할 事項은 별표 4와 같다.

[별표 4]

컴퓨터 導入時經營자가 알고 있어야 할 事項	
1. 프로젝트 팀(Project team) 任命	
2. 電算化 妥當性 調査 結果	
3. 新規시스템 定義와 開發計劃(Project plan)	
4. 新規 시스템의 設計 內容	
a. 입력 설계	e. file 설계
b. 출력 설계	f. 단말 설계
c. code 설계	g. 통신망 설계
d. process 설계	h. recovery 설계
5. 컴퓨터 및 소프트웨어 選定과 導入契約	
—컴퓨터 專門家에게 諮問요청	
6. 適用 소프트웨어(Application software)開發 및 移行日程	
7. 關聯規程·制度 및 各種書式 변경	
8. 適用 소프트웨어 開發 및 테스트 確認	
9. 컴퓨터 設置環境 造成	
a. 컴퓨터실 floor 공사	d. 내장시설
b. 전원설비	e. 통신설비
c. 공조 설비	f. 기타 시설
10. 컴퓨터 設置 및 시스템 테스트	
11. 移行 準備	
a. 조작해설서 및 업무지침서(manual)	
b. 오퍼레이터, 이용자 교육	
c. 신규 전표 및 서식 인쇄	
d. 전산소모품, 비품 등 구입	
12. 並行稼動	

컴퓨터를 導入하는 경우에 컴퓨터를 購買하거나 賃借하게 되는데 컴퓨터가 小型인 경우는 보통 구매를 하게 되나 大型인 경우에는 割賦賃借 또는 條件附割賦購買를 하게 되는 것이 通常의 이며 이들을 細部 항목으로 분류하여 比較하여 보면 별표 5와 같이 비교해 볼 수 있다.

컴퓨터 導入시 가장 중요한 검토사항은 機種 선정이다. 따라서 기종선정시 검토해야 할 重要的 사항은 별표 6과 같이 ① 컴퓨터 도입목적

[별표 5]

컴퓨터의 購買 또는 賃借의 比較

비 교 항 목	購 買 方 式	割 賦 賃 借	條 件 附 割 賦 購 買
	PURCHASE	RENTAL	LEASE
소 유 권	있 음	없 음	계약기간이 경과한 후 잔존가액으로 소유 가능

컴퓨터비용	一時拂支給	月임차료가 購買가격의 $\frac{1}{40\sim50}$ 정도	사용기간(3년, 4년, 5년)을 정한 金利 契約
도입부대비용	일시불	일시불 또는 분할지급	일시불 또는 분할지급
보험료	자기부담	통상적으로 賃借料에 포함되어 있음	자기부담
취득세	구매시 징수	없음	계약기간경과후 구매시 징수
도입자신용여부	상관없음	賃貸會社가 임의로 판단	Lease 보증금: 5~10% -신용보증기관을 요구하는 경우도 있음
보수정비	自己 부담	賃貸會社에서 實費로 제공	自己 부담
기타	-자금이 풍부하거나 변경, 개조를 원하는 경우는 구매방식이 좋음 -대형 컴퓨터의 경우는 구매를 피하는 경향이 있음	-기종의 변경이 빠른 기일 내에 예상되는 경우에는 賃借가 유리함 -Maker의 충분한 After Service가 예상됨	-Lease 기간이 완료되면低價로 購買 可能함 -해약시 상당한 違約金이 필요함.

檢討 ② 經濟性 검토 ③ 類似시스템 및 技術動向 조사 ④ 컴퓨터 規模 선정 ⑤ 機種評價 방법 결정 ⑥ 컴퓨터 製品 선정등이다.

[별표 6]

<p>機種 선정시 檢討할 事項</p> <p>1. 컴퓨터 導入目的</p> <p>-도입 목적의 妥當性 재검토</p> <p>-기술적 및 現實의 可能性</p> <p>-장기계획과의 一貫性 검토</p> <p>2. 經濟性 검토</p> <p>-費用대 效果 분석</p> <p>-컴퓨터 관련 費用負擔 능력</p> <p>3. 類似시스템 및 技術動向 조사</p> <p>-類似시스템과의 對比</p> <p>-最新 技術 動向 조사</p> <p>4. 컴퓨터 規模 선정</p> <p>-전산화 대상업무 範圍 결정</p> <p>-전산처리량 規模 추정</p> <p>-CPU 및 입출력 장치 처리 速度</p> <p>-단말기 및 통신 Network 구성</p> <p>-자료 처리 方式 선택</p> <p>• Batch, real time, TSS</p> <p>5. 機種評價 방법 결정</p> <p>-POED</p> <p>-GIBSON MIX</p> <p>-AUER BACH</p> <p>-BENCH MARK</p>

- CHECK LIST 등
- 6. 컴퓨터 製品 選定
- 하드웨어(H/W) 性能
- 소프트웨어(S/W) 適合性
- 시스템 擴張性
- 메이커의 支援사항
- 補修整備 및 障害복구체계
- 價格 및 賃貸조건
- 他회사 納品실적
- 메이커의 知名度
- 同一메이커 제품의 고객 評判
- 컴퓨터 전문가에게 諮問요청

3. SYSTEM 開發

가. 시스템의 개발요구가 있으면 무슨 업무를 왜 전산화해야 하는가를 검토하는 妥當性調査를 실시하여 문제점과 개발목적 그리고 업무범위를 정하게 된다. 전산화의 타당성이 인정되는 경우에는 사용자의 要求事項을 정확히 정의한다. 要求分析단계에서는 사용자의 要求사항에 관하여 사무처리의 흐름, 전표류의 통제합, 標準化등 전산화하기 위한 現象조사와 업무분석을 면밀히 실시하고 制約條件을 감안하면서 그 要求사항에 타당성이 있는가를 다각적으로 검토하게 된다. 이때 要求分析의 대상은 별표 7과 같다.

[별표 7]

- 要求 分析 대상**
1. 電算化 目的
 - 경제적, 기술적, 현실적 타당성 분석
 2. 電算化 要求事項
 - 배경 분석
 - 효과 분석
 - 기능 분석
 - 규모 분석
 - 현실성 분석
 - 긴급성 분석
 - 적용시기 분석
 3. 適用範圍
 - 대상업무 범위의 분석
 - 규정 및 제도의 변경 검토
 - 처리 방식 변경의 범위
 - 처리기한의 타당성
 - 사람과 컴퓨터간의 관계
 - 기능 확장 분석
 - 신규 업무 추가의 범위
 - 관련 부문의 영향도 파악
 - 정상적 처리와 예외처리의 구분
 4. 關聯情報分析
 - 조직과 기능
 - 사무처리도
 - 입출력 전표
 - 장부 및 관리대장
 5. 制約條件分析
 - 시간적 조건
 - 경제적 조건
 - 기술적 조건
 - 조직적 조건
 - 장래의 환경변화 조건

나. 사용자의 요구사항을 분석한 결과 시스템 개발의 必要性이 인정되면 시스템分析을 실시하게 된다.

시스템 分析단계에서는 요구분석단계에서 분석한 조사결과를 기초로 해서 그에 대응한 類似 시스템과 최신의 技術動向을 조사하여 사용자가 요구한 사항에 代替案을 작성하고 사용자의 환경을 충분히 감안하여 비교 검토하면서 최적 시스템案의 개요를 작성하게 된다. 시스템分析對象의 상세한 내역은 별표 8과 같다.

[별표 8]

- 시스템 分析 대상**
1. 類似시스템 및 技術動向 조사
 - 유사시스템 조사
 - 개발자, 명칭, 대상범위, 기능, 효과
 - 구성, 규모, 특징, 비용, 개발기간,
 - 적용예, 문제점 평가
 - 기술 동향 조사
 - 自社의 타부분, 민간기업, 컴퓨터베이커
 - 소프트웨어하우스, 대학, 연구소 등
 - 최신 자료 참고
 - 하드웨어의 기능, 성능, 가격조사 시스템개발 운용의 방법, 기법 등 선진적 user의 사례 각종 문헌, 해외 자료
 2. 環境條件
 - 하드웨어(H/W)
 - 시스템의 기기구성, 업무의 규모, 처리비용에 적합한 기기의 선택
 - 소프트웨어(S/W)
 - 프로그래밍 언어, 서비스 제공 프로그램의 적합성
 - 제약 조건
 - 시간적, 경제적, 기술적, 조직적 장래의 환경 변화
 3. 代替案 作成 및 비교검토
 - 기능·성능
 - 대상업무범위, 처리효과, 대기시간, 관련 부문과의 관계
 - 경제성
 - 기능·효과와 비용, 환경조건과 비용
 - 효과
 - 관리, 비용, 업무, 연구 면에서 비교 검토 평가
 - 기술분석
 - H/W, S/W의 기술 동향
 - 유사시스템과의 대비
 - 상위점, 이점, 불리점
 - 기능, 범위, 경제성, 적용기법
 - 體制·工期
 - 업무범위, 개발, 운용의 工數, 체제, 工期
 - 개발난이도
 - 작업내용, 작업량, 기술적 대응, 일정표 작성
 4. 시스템 案
 - 목적과 효과
 - 대상업무범위
 - 시스템 概要 設計
 - 입출력 정보, 기본기능의 구성과 처리개요, 시스템의 규모·기능·비용·工期, 工數

다. 사용자의 요구사항을 충족시키고 이해하기 쉬우며 維持補修가 용이한 代替案들을 비교, 검토, 평가하여 주어진 期間과 費用내에서 機能과 性能이 종합적으로 가장 우수한 시스템을 선택하여 별표 9 와 같은 시스템定義書를 作成하게 된다.

[별표 9]

시스템 定義書(System Definition)	
1. 문제점 기술	
2. 시스템의 타당성	
3. 시스템의 목적과 업무범위	
4. 환경 및 제약조건	
5. 시스템의 기능 및 성능	
6. 사용자의 특징(요구사항 등)	
7. 개발, 운용, 보수유지 환경	
8. 시스템의 목표	
9. 시스템 기능의 우선순위	
10. 시스템의 접수기준	
11. 참고 자료	
12. 용어의 해설	

[별표 10]

프로젝트 계획(Project plan)

단계	요구분석 및 계획	설계	코딩	테스트	시스템 인수	실치 / 운용	유지보수
개발내용	문제분석	시스템 개발계획					
		기본설계					
		상세설계	프로그램명 및 테스트				
		공통	시스템테스트준비, 정산시스템 인수, 인수테스트준비, 인계시스템 인수, 인계교과서, 시스템설계 및 테스트, 운영교과서, 운영준비				
시스템내용자문	시스템내용자문	시스템내용자문	시스템내용자문	시스템내용자문	시스템내용자문	시스템내용자문	
文書(산의 주요인물)	시스템 개발계획서	시스템 기본설계서	시스템 상세설계서	시스템 인수준비요령서	시스템 인수준비요령서	시스템 인수준비요령서	시스템 인수준비요령서
	시스템의 인수준비	시스템의 인수준비	시스템의 인수준비	시스템의 인수준비	시스템의 인수준비	시스템의 인수준비	시스템의 인수준비

라. 개발하여야 할 시스템이 定義되면 여기서 개발목적과 업무범위가 명백히 정해지게 되므로 이에 따라 시스템 개발의 전반적인 計劃을 수립하게 된다. 시스템개발에는 여러가지 週期(life cycle)모델이 있는데 一例로서 별표 10 과 같은

프로젝트 계획(project plan)을 세워야 하며 별표 11 과 같은 여러가지 計劃書를 作成해야 한다.

마. 시스템 設計단계에서는 사용자의 요구사항을 충족시키기 위해서 컴퓨터가 어떻게 기능해야 하는가를 정의하는데 이 設計(design) 단계에는 별표 12 와 같이 入力, 出力, 處理, file, 통신망(network), 장애대책 등의 설계가 수행되어야 하며 별표 13 과 같이 각각의 배치도(layout)와 발간해야 할 책자가 設計되어야 한다.

[별표 11]

計劃의 종류	
1. 豫算(費用 見積)	
2. 日程계획	
3. 外注계획	
4. 品質管理계획	
5. 要員계획	
6. 設備계획	
7. 移行계획	
8. 運用계획	
9. 補修維持계획	
10. 擴張계획	

[별표 12]

설계(Design) I	
1. 입력(Input)	
—전표	
—단말 키보드(keyboard)	
—입력형태 분류(transaction)	
2. 출력(Output)	
—조회화면(display)	
—보고서	
—출력 단말	
3. 처리(Process)	
—기능	
—처리능력	
—대기시간	
—장애회복(recovery)	
4. 코우드(Code)	
5. 화일(File)	
6. 데이터 베이스(Data Base)	
—개념적 설계	
—물리적 설계	
7. 통신망(Network)	
—회선망	

- 통신 장비 구성
- 8. 장애 대책
 - 예비(back-up)
 - 복구(recovery)
- 9. 移行
- 10. 運用

[별표 13]

- 설계(Design) II
(A. 배치도)
1. 하드웨어구성(H/W configuration)
 2. 소프트웨어 구성(S/W configuration)
 3. 전자제산실 배치도(Layout)
 4. 공조시설 배치도(Layout)
 5. 자료 소산 배치도(Layout)
 6. 보안 유지 시설 배치도(Layout)
 7. 각종 부속실 배치도(Layout)
- (B. 발간책자)
1. 시스템 개요 설명서
 2. 시스템 기본 설계서
 3. 프로그램 상세 설계서
 4. 시스템 운용 책자(manual)
 5. 기기 조작 해설서(manual)
 6. 장애 대책 지침서(manual)
 7. 장애 관련 리스트(error message)
 8. 프로그램 관리 리스트

바. 개발되어야 할 소프트웨어(software)는 시스템 정의서(system definition)를 기초로 하여 소프트웨어 요구사항 명세서로 작성된다. 별첨 14와 같은 소프트웨어 요구사항 명세서는 소프트웨어 제품이 갖는 특성을 보다 상세하게 설명하기 위해서 記述하는 것으로 소프트웨어의 必須要件을 정의하고 요구되는 機能性能, 例外條件, 引受基準등을 상세하게 기술하게 된다.

[별표 14]

- 소프트웨어 要求사항 명세서 형식
1. 소프트웨어의 개요
 2. 개발·운용·보수유지 환경
 3. 외부 인터페이스와 자료 흐름
 - 화면 및 보고서 형식
 - 사용자 명령어 요약
 - 자료 흐름도

- 자료 공급처·제출처
- 자료저장소
- 자료 사전
- 4. 기능 요구
- 5. 성능 요구
- 6. 예외 처리
- 7. 구현 우선 순위
- 8. 변경 및 개선 예정 사항
- 9. 인수 기준
 - 기능 및 성능 테스트
 - 문서화 표준
- 10. 설계시 힌트와 제약 조건
- 11. 참고 자료
- 12. 용어의 해설

사. 사용자가 원하는 소프트웨어 요구사항 명세서와 설계단계에서 만들어진 각종設計書를 원시코드(source code)로 변환시키는 작업을 프로그래밍이라고 한다. 이때 원시코드를 記載하는 행위를 코딩이라고 하는데 프로그래밍과 코딩을 혼용하기도 한다. 프로그래밍과 테스트 과정을 圖解하여보면 별표 15와 같이 그려볼 수 있겠다.

아. 프로그래밍 단계에서는 설계 단계에서 만들어진 각종 設計書대로 각종 프로그램을 프로그래밍言語로 코딩하여 개별적으로 單位테스트를 실시한다. 이때 誤謬(bug)가 있으면 디버깅(debugging)* 작업을 실시하여 설계명세서와 원시코드(source code)가 일치하도록 檢證 및 修正작업을 계속하게 된다.

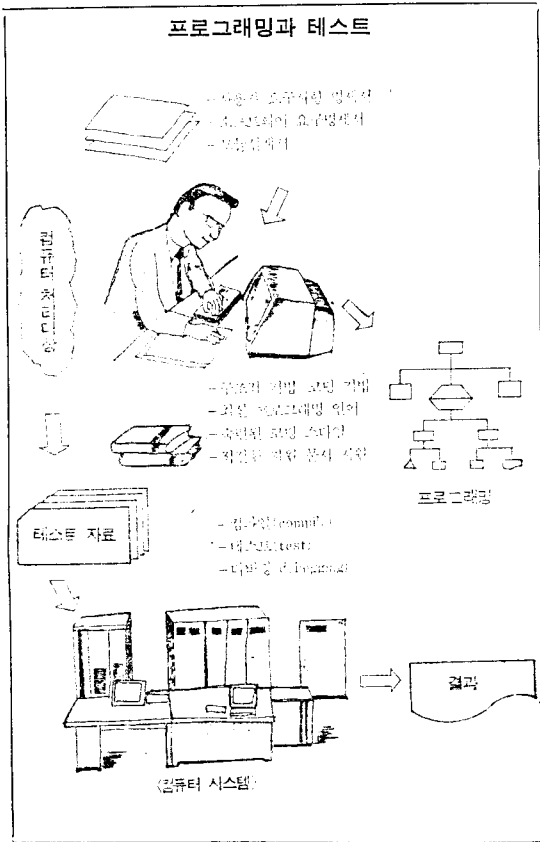
자. 單位테스트가 완료되면 모든 프로그램을 연결해서 프로그램간의 상호 接續기능이 잘 되어 있는지를 확인하는 結合테스트를 실시하며 실제 사용자의 운용 環境을 준비하여 놓고 시험자료(test data)를 이용하여 引受테스트(acceptance test)도 실시하게 된다.

전체 시스템의 개발이 완료되고 無故障성이 입증된 후에라도 충분한 여유를 갖고 新舊시스템을 일정한 기간동안 同時に 並行처리하여 보는 것이 바람직하다. 지금까지 설명한 테스트과정은 별표 16과 같이 圖示해 볼 수 있겠다.

차. 한편 개발된 시스템을 운영하고 維持하기 위한 각종 文書와 指針書등을 작성하고 또 이시

* 디버깅(dubugging) ; 프로그램을 작성할 때 잘못된 부분(bug)을 修正하는 행위를 디버깅이라고 한다.

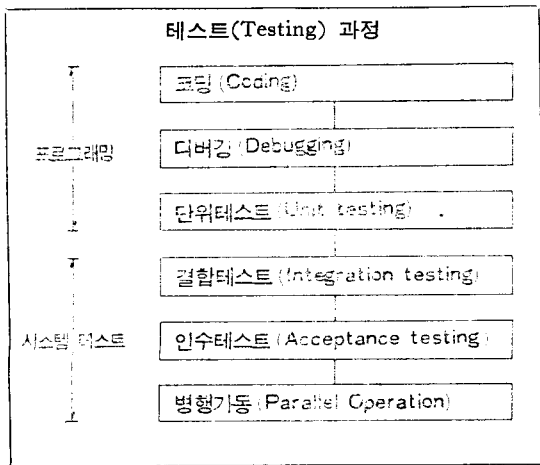
[별표 15]



시스템을 이용해야 할 사람들을 위해 教育訓練도 실시해야 한다.

카. 일반적으로 시스템을 開發한 사람과 실제

[별표 16]



시스템을 사용할 사람들은 다른 그룹(group)에 속한다. 따라서 별표 17 및 별표 18과 같은 點을 유의하여 檢數를 實施하게 된다.

[별표 17]

檢數時 유의사항

1. 檢數를 위한 Check List 작성
2. 檢數成績書 확인
3. 실제 가동 환경에서의 性能 테스트
4. 사용자 운용요령 등 각종 文書 검사
5. 補修維持(maintenanc) 체제 확인

[별표 18]

소프트웨어의 檢數 Point

1. 機能의 適合性(Usability)
 - 사용자 요구사항 충족도
 - 제한사항의 타당성
 - 잘못된 입력 자료에도 불구하고 기능유지 정도
2. 信賴性(Reliability)
3. 補修維持의 容易性(maintainability)
 - 障害 조사의 便宜性
 - 보수시 修正 속도의 신속성
4. 性能(Performance) 評價
 - 他 시스템과의 처리 속도 비교
 - 단위시간당 처리량(throughput)
 - 처리시간(turn-around time)
 - 응답 대기 시간(response time)
5. 使用者 運用要領 등 각종 文書의 記載事項

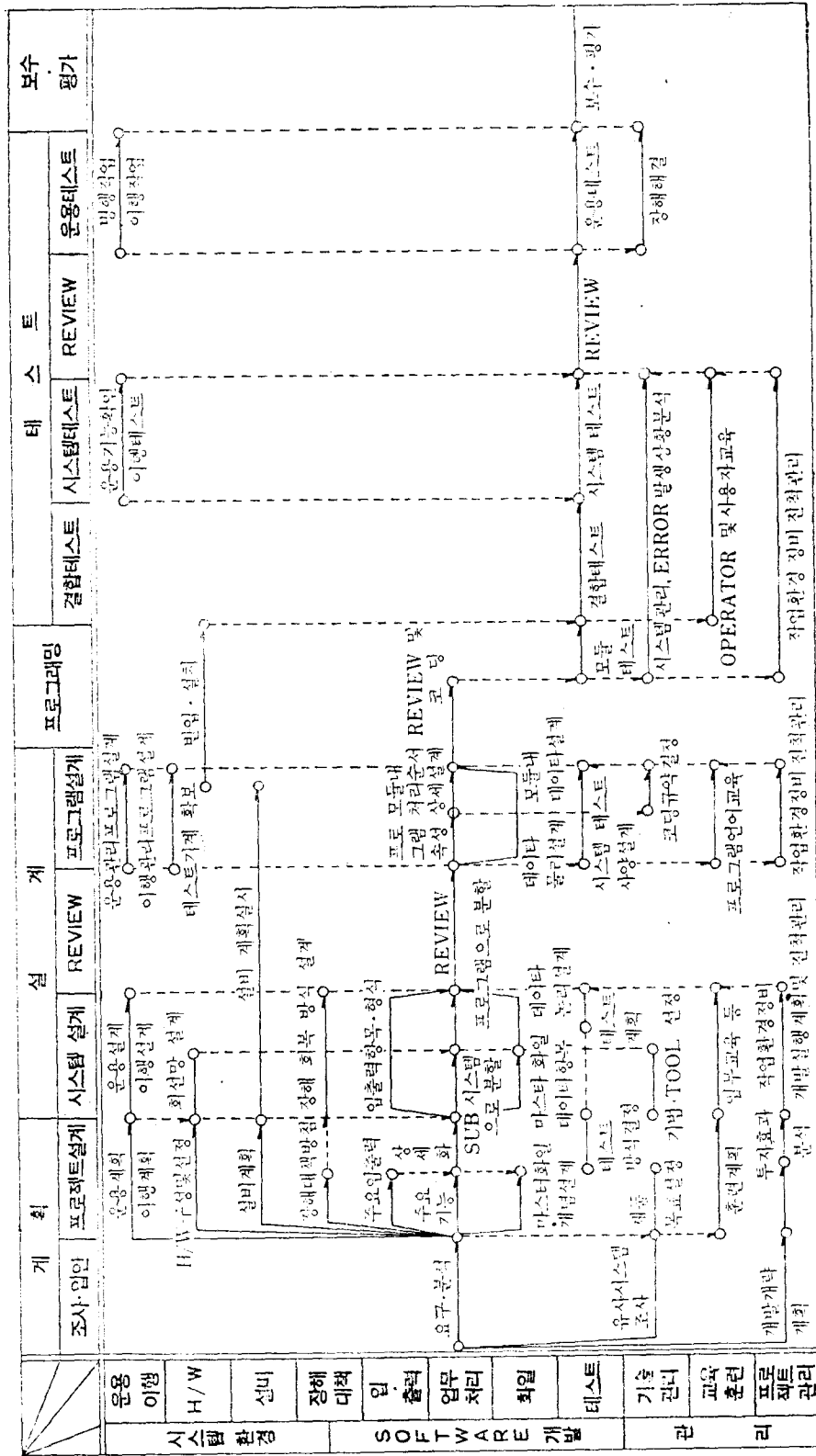
다. 컴퓨터시스템이 컴퓨터製作會社(marker)로부터 사용자에게 인도되면 사용자(user)는 生産化 要求(need)에 맞추어 컴퓨터 시스템을 構成(configuration)하고 적용프로그램(application program)을 개발하여 정보처리를 실행하게 된다. 이렇게 사용자의 특정목적(예를들면 은행은·라인 시스템)에 맞추어 설치한 컴퓨터 시스템이 사용자의 운용환경에서 최상의 性能으로 운용되도록 시스템을 조정하게 되는데 이러한 調整과정을 시스템 튜닝(tuning)이라고 부른다.

시스템 튜닝 작업은 여러가지 高度의 技術的 검토와 確率的 性能평가가 병행적으로 이루어져야 하는 복잡한 작업이므로 情報處理 專門家에게 맡겨야 한다.

지금까지 설명한 開發過程을 ① 운용·移行 ② H/W ③ 設備 ④ 장애대책 ⑤ 入出力 ⑥ 업

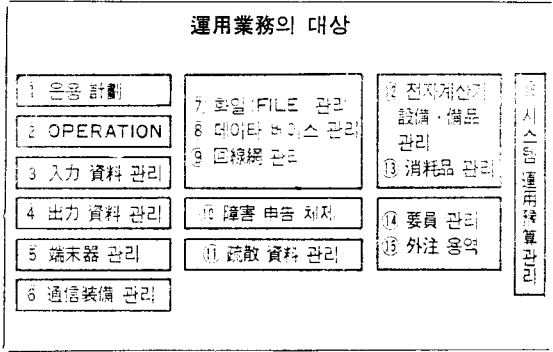
電算化를 위한 開發過程 “實例”

[별표 19]



무처리 ⑦ FILE ⑧ TEST ⑨ 技術管理 ⑩ 教育訓練 ⑪ 프로젝트管理로 구분하여 별표 19와 같은 電算化를 위한 開發過程 “實例 로서 圖解하였다.

[별표 20]



4. 시스템의 保修維持

가. 開發이 완료되어 정상적인 稼動에 들어가면 별표 20과 같은 業務가 運用의 對象이 된다 이때 運用管理를 效果的으로 實行하려면 별표 20과 같은 事項이 충분히 검토되어야 한다.

나. 시스템 開發이 완료되어 정상적으로 운용중이던 프로그램도 사용자의 환경이 변화하면 이에 대응하기 위해 지속적인 시스템의 補修維持가 要求된다.

[별표 21]

運用 管理時 考慮事項

1. 容易性: 運用(operation)이 쉬워야 한다.
2. 適應性: 환경 變化에 잘 적응해야 한다.
3. 經濟性: 費用이 저렴해야 한다.
4. 障害對策: 장애 復舊가 신속해야 한다.
5. 安全對策: 事故에 備한 안전대책을 강구해야 한다.
6. 機密保護: 資料(data)의 保安을 유지해야 한다.
7. 資料疏散: 돌발적 事態 및 事故에 대비해야 한다.
8. 災害對策: 天災地變 등 재해시 對策을 미리 강구한다.

이때 설계과정에서 고려된 확장범위 以外的 범위까지 대폭적으로 시스템의 변경이 要求될때는 經濟性和 效率性을 면밀히 검토하여 운용중인 시스템을 補修할 것인가? 또는 전혀 새로운 시스템을 개발할 것인가를 결정해야 한다.

시스템의 補修유지와 變更관리를 效果的으로 실행하기 위하여 다음과 같은 管理가 必要하다.

- 프로그램별 보수유지 이력카드 사용
- 소프트웨어 변경관리 위원회 운영
- 소프트웨어 변경요청 일람 작성
- 소프트웨어 품질보증 활동
- 보수·유지 프로그래머 조직의 구성

新刊 컴퓨터의 脈中에서

—一部 발췌—