

---

 論 文
 

---

大韓造船學會論文集  
 第29卷第2號 1992年 5月  
 Transactions of the Society of  
 Naval Architects of Korea  
 Vol. 29, No. 2, May 1992

## 컴퓨터 그래픽 표준에 대한 조사연구

한순홍\*, 서승완\*

## A Survey of the Computer Graphics Standards

by

Han, Soon-Hung\* and Suh, Seung Wan\*

## 요 약

컴퓨터 그래픽의 사용이 보편화 되어감에 따라 컴퓨터 그래픽에 대한 표준화가 필요하게 되었다. 국가적인 차원의 共用 시스템을 개발하려는 CSDP 사업을 위해서도 공통적으로 쓰일 그래픽 라이브러리의 선정이 필요하다. 이 글에서는 여러가지 그래픽 표준들에 대해 조사 및 정리하고, 전체적인 관점에서 그들의 상관관계를 살펴보았으며, 우리가 추진해 나가야할 방향에 대하여 검토해 보았다. 따라서 ISO 같은 국제적인 공공기관에서 채택된 표준들도 중요하지만, 산업계에서 널리 사용되므로서 표준이나 마찬가지로 그래픽 라이브러리들도 중요하다. *De facto standard*라고 부르는 이러한 그래픽 라이브러리들을 함께 다루어 보았다.

## Abstract

As more application programs have graphic functions, standards in the computer graphics are becoming important. To develop a computer aided design and production system for ships, which would be used by different shipyards, a graphic library should be chosen as a standard. Existing ISO graphic standards and *de facto* industrial standards are surveyed. They are, to name some of them, GKS, PHIGS, X-Window, PEX, PDES, and STEP. GKS and PHIGS concern about the application programmer's interface. Others handle the device interface. PDES and STEP are to standardize the graphical data transfer between heterogeneous systems. Their inter-relations are investigated. Some preliminary suggestions for the choice of graphic standards are included in this paper.

---

 발 표 : 1991년도 대한조선학회 춘계연구발표회('91. 4. 13.)

접수일자 : 1991년 4월 29일, 재접수일자 : 1991년 12월 4일

\* 정회원, 한국해사기술연구소

1. 서 언

1.1. 그래픽 표준의 필요성

빠른 컴퓨터의 발달에 힘입어 컴퓨터 그래픽의 사용이 보편화 되고 있다. 그것은 소프트웨어의 발전도 많았지만, 하드웨어의 가격이 낮아지며 동시에 그 기능이 향상되었기 때문이다. 7, 8년전만 하더라도 그래픽 기능이 있는 프로그램을 사용하기 위해서는 값비싼 그래픽 단말기와 플로터가 필요했다. 요즘에는 PC와 레이저 프린터가 그보다 더 나은 성능을 내고 있다. 물론, 그 뒤에는 점의 모임으로 화상을 나타내는 Raster Graphics라는 기술의 뒷받침이 있었다. IBM PC의 Microsoft Window나 Apple Macintosh의 Finder를 생각해 볼때, 상업용 프로그램은 이제 그래픽 기능이 없이는 판매가 어려워질 것으로 보이며, 일반 프로그래머들도 개발하는 프로그램에 그래픽 기능을 많이 추가하고 있다.

이렇게 컴퓨터 그래픽의 사용이 보편화되면서 새로운 문제가 발생하였는데, 그것은 호환성이 없는 그래픽 라이브러리(Library)의 양산이다. 현재 사용되고 있는 몇 가지 그래픽 라이브러리를 살펴보면, IBM PC의 CGA, EGA, VGA가 있고, Apple사의 Macintosh에서는 Toolbox라고도 불리는 Quickdraw가 있다.

Calcom사에서 제공한 PLOT10 라이브러리도 많이 쓰이고 있으며, Apollo 컴퓨터에서는 GPR과 GMR이라는 라이브러리를 제공한다. 이밖에 국내 조선소에서 사용중인 AUTOKON 시스템에서는 GPGS와 IGOR이라는 그래픽 라이브러리를 사용하고 있다.

이렇게 다양한 그래픽 라이브러리를 사용하고 있기 때문에 이식성(Portability)문제가 발생한다. 그래픽 기능이 없는 일반 공학용 프로그램에서는, 그 프로그램 작성에 FORTRAN 77이나 ANSI C와 같은 표준 언어를 사용하면 이식성 문제가 생기지 않는다. 그러나 그래픽 기능이 있는 프로그램을 개발하기 위하여, 특정기종에서 제공하는 그래픽 라이브러리를 사용하면, 그 개발된 프로그램은 다른 기종에서 수행이 안된다. 이것은 그 프로그램이 특정 하드웨어의 기계적 특성에 의존하게 되기 때문이며, 이러한 것을 장치독립성(Device Independence)이 없다고 한다.

다른 분야의 표준화 작업에도 똑같이 적용될 수 있는 논리이지만, 프로그램의 이식성이 없으면 기종에 따라 프로그램을 고쳐야하며, 이에 따라 프로그래머의 재교육도 필요하다. 또한, 작성하는데 오랜 시간이 투입된 그래픽 데이터를 다른 기종으로 옮길 수가 없다. 이러한 문제는 결국 돈과 시간의 문제 즉, 생산성의 문제로 연

결된다. 한편, 컴퓨터 그래픽에 대한 용어와 개념의 표준화는 프로그래머의 교육을 위해서도 필요하다. 하드웨어 기능의 향상에 따라 소프트웨어의 생산성이 더욱 강조되고 있는 요즘에는 소프트웨어의 표준화가 소프트웨어 생산성 향상에 큰 역할을 한다.

근래에 관심의 대상이 되고 있는 통합화된 자동화 시스템(Computer Integrated Manufacturing : CIM)이 추구되기 위해서도 표준화가 의미하는 바는 크다. 그래픽 표준은, 국가적인 차원의 共用 시스템을 개발하려는 경우에도 주요 문제이다. 현재 진행중인 '선박 설계·생산 전산시스템 개발사업'(이하 CSDP라 부른다)을 위해서도 표준화가 필요하다. 조선소마다 다른 컴퓨터기종을 사용하고 있고, 다양한 소프트웨어 시스템을 도입해 사용하고 있는 국내에서는, CSDP사업을 통해 개발된 프로그램과 자료의 교환 문제가 발생할 것이다. 이러한 문제는 다양한 컴퓨터 시스템을 사용하고 있는 하나의 조선소 내부에서도 똑같이 나타나는 것이다.

1.2 표준화 작업현황

이러한 중요성을 일찍부터 느낀 선진국들은 그동안 표준화 작업을 추진해 왔다. 경제적인 통합의 길로 나아가고 있는 유럽은 표준화의 필요성을 가장 많이 느끼고 있으며, 특히, 컴퓨터 부문에서 미국에의 의존이 심하므로, 그 영향력에서 벗어나기 위해서도 표준화 작업에 노력하고 있다. 컴퓨터 부문을 주도해온 미국으로서는 그 위치를 고수하기 위하여 표준화 작업에 깊이 참여하고 있다. IBM과 같은 대형 메이커와는 달리 규모가 작은 소프트웨어 개발업자와 하드웨어 제작자에게는 그래픽 표준이 더 중요하다. 우리나라의 경우도 이러한 소규모 소비자의 입장에 있다고 할 수 있겠다.

그래픽 표준을 만드는 주요 조직으로는, 국제표준 제정기구인 ISO(International Standard Organization)와 IEC(International Electrotechnical Commission)의 공동 위원회(Joint Technical Committee)인 ISO/IEC JTC1과 미국 ANSI(American National Standards Institute)의 위임을 받은 X3 위원회가 있다. ISO 내에서 그래픽 표준에 대한 작업은, 분과위원회(Sub-Committee)인 JTC1/SC24에서 다루고 있다. 미국의 X3내에는 X3H3라는 분과위원회가 있어, 여기서 그래픽 표준을 다루고 있다. Table 1은 1989년 6월 현재 ISO에서 작업중인 그래픽 표준들을 보여준다[32]. 여기서 WD는 Working Draft, DP는 Draft Proposal, DIS는 Draft International Standard, IS는 International Standard의 약자이다.

이 글은 CSDP사업에 공통적으로 쓰일 그래픽 라이

브러리의 선정을 위해 기초적인 작업을 한 결과이다. 따라서 ISO 같은 국제적인 공공기관에서 채택된 표준들도 중요하지만, 산업계에서 널리 사용되므로서 표준이나 마찬가지로 라이브러리들도 중요하므로 *De facto Standard*라고 부르는 이러한 라이브러리들을 함께 다루어 보았다. 이러한 목적을 위해 여러가지 그래픽 표준들에 대해 조사 및 정리하고, 전체적인 관점에서 그들의 상관관계를 살펴보았으며, 우리가 추진해 나가야할 방향에 대하여 검토해 보았다.

2절에서는 응용프로그램머들에게 필요한 응용 프로그램 水準의 표준을 살펴보았으며, 3절에서는 그래픽 하드웨어 제작자나 시스템 프로그래머들에게 필요한 하드웨어 수준의 표준들을, 4절에서는 서로 다른 하드웨어나 소프트웨어들 간에 그래픽 정보를 교환 하는데 필요한 표준에 대해 살펴보았다. 5절에서는 새롭게 출현하고 있는 몇가지 중요한 시스템들에 대하여 조사하였으며, 6절에서는 국내의 현황과 함께, 선박의 설계 및 생산 전산화에의 활용방안에 대하여 살펴보았다. 그 전체적인 윤곽은 Fig. 1과 같으며[31], 5절에서 상세히 설명될 것이다.

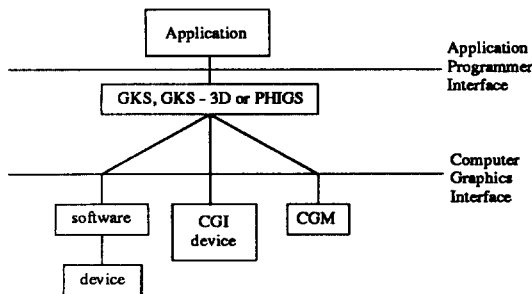


Fig. 1 Relationship among ISO graphics standards [31]

2. 응용프로그램 수준의 그래픽 표준

응용프로그램 수준의 표준(Application Programmer's Interface : API)은, 그래픽 응용 프로그래머의 입장에서 필요로 하는 기능들을 모두 나열하고, 이들을 설계하여 표준화한 것이다. 컴퓨터 그래픽 교과서에 나타나는 기능들을 표준화한 것으로 볼수 있으며, 실제 그 기능이 어느 특정한 그래픽 장치에서 구현될수 있는가는 염두에 두지 않는다.

현재까지 ISO 표준으로 제정된 것으로는 1985년에 채택된 GKS(Graphical Kernel System)와 1988년에 채

택된 GKS-3D, 그리고 1989년에 채택된 PHIGS(Programmer's Hierarchical Interactive Graphics System)가 있다. GKS는 2차원 그래픽 만을 그 대상으로 하고 있으며, 처음에는 독일의 국가표준(National Standard)이었다. 이것을 3차원 그래픽으로 확장한 것이 GKS-3D이며, 따라서 GKS와 GKS-3D는 호환성이 있다. PHIGS는 처음부터 3차원 그래픽을 대상으로 미국에서 설계된 것이며, GKS와는 호환성이 없다. PHIGS PLUS는 PHIGS의 기능을 확장한 것인데, 현재 ISO에서 표준화 작업중에 있다.

2.1 2차원 그래픽 표준 GKS

GKS는 ISO에서 제정한 최초의 그래픽 표준인데, 전체가 약 200개의 프로시저(Procedure)들로 구성되어진 그래픽 라이브러리이다. Segment라는 개념을 도입해서 그래픽 작업의 내용들을 저장하며, 서로 다른 그래픽 장비의 용량에 맞추어 구현(Implementation) 할 수 있도록 몇개의 단계(Level)로 나누어져 있다. 2차원 그래픽 만을 다룰 수 있으며, 이미 정의되어 있는 물체를 화면에 보이는 (Viewing)기능에 초점이 맞춰져서 개발되었다.

Fig. 2에는 GKS의 구성요소들을 보였다[35]. 언어결합(Language Binding)은 응용프로그램에서 쓰이는 언어, 즉 Fortran, Pascal, C 등의 언어에 따라 GKS 라이브러리의 접속이 다르게 되는 것을 나타낸다. Device driver는 각 그래픽 장비마다 필요하며, 각 장비의 특성에 따라 정의된 그래픽 기능을 어떻게 구현(Implementation)하는가를 정의한다. 이때 하드웨어가 직접 제공하지 못하는 기능은 소프트웨어로서 모방(Emulation)되기도 한다.

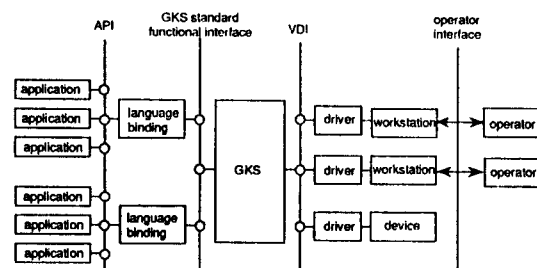


Fig. 2 GKS interfaces [35]

2.2 3차원 그래픽 표준 PHIGS

GKS는 이미 정의되어 있는 물체를 화면에 보이는 (Viewing)기능에 초점이 맞춰져서 개발되었지만, PHIGS는 여기에 물체를 자유롭게 정의하고 변경시키는데 필요한 모델링의 개념이 추가된 것이다. Fig. 3은

Table 1 Current status of ISO/IEC JTC1/SC24 projects(as at June 1989) [32]

Project	Document Reference	Availability of Text for :			
		WD	DP	DIS	IS
GKS	ISO 7942				IS (1985)
GKS Addendum 1	ISO 7942/ADD.1			10/88	10/89
GKS Language Bindings					
Fortran	ISO 8651-1				IS (1988)
Pascal	ISO 8651-2				IS (1988)
Ada	ISO 8651-3				IS (1988)
C	SC24/N180	7/88			
CKS-3D	ISO 8805				IS (1988)
CKS-3D Language Bindings					
Fortran	DIS 8806-1				4/89
Pascal	SC24/N190	7/88			
Ada	SC24/N189	7/88	2/89	6/90	
C	SC24/N181	7/88	3/89	12/89	
PHIGS					
Functional description	ISO 9592-1				IS (1989)
Archive file format	ISO 9592-2				IS (1989)
Archive file clear text encoding	ISO 9592-3				IS (1989)
PHIGS PLUS	ISO 9592-4	6/89	10/89		
PHIGS Language Bindings					
Fortran	ISO 9593-1				12/88
Extended Pascal	DP 9593-2		5/89	12/89	
Ada	DIS 9593-3		9/87	1/89	8/89
C	DP 9593-4		5/89	12/89	
CGM					
Functional description	ISO 8632-1				IS (1987)
Character encoding	ISO 8632-2				IS (1987)
Binary encoding	ISO 8632-3				IS (1987)
Clear text encoding	ISO 8632-4				IS (1987)
CGM Addendum 1					
Functional description	ISO 8632-1/ADD.1			10/88	10/89
Character encoding	ISO 8632-2/ADD.1			10/88	10/89
Binary encoding	ISO 8632-3/ADD.1			10/88	10/89
Clear text encoding	ISO 8632-4/ADD.1			10/88	10/89
CGM Addendum 2					
Functional description	ISO 8632-1/ADD.2	3/88	5/89	10/89	10/90
Character encoding	ISO 8632-2/ADD.2	10/89			
Binary encoding	ISO 8632-3/ADD.2	10/89			
Clear text encoding	ISO 8632-4/ADD.2	10/89			
CGI Parts 1-6	DP 9636		11/88	10/89	2/91
CGI Character encoding	SC24/N209	3/89	10/89		
CGI Binary encoding	SC24/N210	3/89	10/89		
CGI Library language binding					
Fortran	SC24/N192	1/89	8/89	4/90	6/91
C	SC24/N191	1/89	8/89	4/90	6/91
Conformity Testing	SC24/N185	7/88	3/89	12/89	10/90
Reference Model	SC24/N177	2/89	9/89		

GKS와 PHIGS의 차이점을 개념적으로 보여주고 있으며 [17], PHIGS가 GKS와 다른 점들을 열거하면 다음과 같다.

- 그래픽 데이터를 계층적 구조를 이용해서 저장한다. (Centralized hierarchical data store)
- 사용자의 조작에 대한 시스템의 응답을 더욱 자연스럽게 구현할 수 있다. (Dynamic and responsive nature of interactions)
- 표현하고자 하는 물체를 모델링하는데 필요한 기능들을 제공한다. 이것은 위에 열거된 계층적 데이터 구조 때문에 가능하다. (Modeling capability)
- RGB (Red, Green, Blue) 이외의 다른 칼라 모델을 이용할 수 있다.

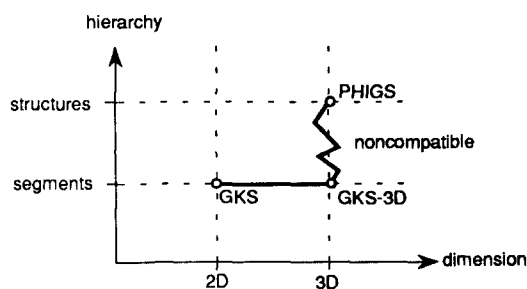


Fig. 3 Differences among GKS, GKS-3D and PHIGS [17]

PHIGS PLUS는 빛과 칼라를 이용해 물체를 실제의 것처럼 보여주는 Rendering 기능 등을 추가한 것이다. PHIGS에 비해서 다음과 같은 기능들을 추가로 가지고 있으며 [15], 현재 ISO에서 표준화를 위한 작업중에 있다.

- Non-uniform B-spline (NURBS)과 Parametric polynomial로 표시되는 곡선과 곡면을 처리하는 기능.
- 다각형의 집합으로 표현되는 표면(Polyhedral mesh surfaces)을 처리하는 기능.
- 물체간의 거리감을 느끼게 하는 Depth cueing과 같은 다양한 3차원 효과를 낼 수 있는 기능.
- 照明(Lighting)과 明暗처리(Shading) 기능.

### 3. 하드웨어 수준의 그래픽 표준

하드웨어 수준의 표준(Virtual Device Interface : VDI)은 장치독립성(Device independence)을 유지하는 마지막 지점이며, 표준화된 Device driver를 만드는 데서 그 개발이 비롯되었다.

假想의 그래픽 장치를 대상으로 설계되는 응용 프로그램 수준의 그래픽 표준(API)과는 달리, 몇가지 선정된 그래픽 장치를 대상으로 하여 설계되어진다. 따라서 그래픽 장비들의 각 부분을 직접 제어할 수 있는 기능을 갖고 있다.

이같이 하드웨어를 직접 고려하는 단계에서는 장치 독립성도 중요하지만, 특정 알고리즘의 성능(Performance)도 중요하다. 일반적으로 한가지의 그래픽 기능이 하드웨어에 따라 다른 성능을 보이게 되며, 보통의 Device Driver는 각각의 하드웨어에서 가장 효율적인 방법으로 구현화(Implementation)되는 경향을 보이게 된다. 이러한 경향은 표준화와는 반대의 방향이다.

이 수준의 그래픽 표준은 그 사용자로서 그래픽 시스템 프로그래머, 소프트웨어 공급회사, 그리고 그래픽 하드웨어 제작자를 포함한다. 따라서 이 표준을 System Programming Interface라고도 한다. 현재 이 수준의 표준으로는 Computer Graphics Interface(CGI)라는 것이 있어, ISO 표준으로 채택되어지기 위한 최종단계에 도달해 있다.

한편, 미국 MTT 공대의 Athena과제로 부터 시작되었고, 이제는 컨소시엄이 구성되어 계속 추진되고 있는 X-Window시스템이 CGI와 유사한 기능을 갖고 있다[27]. 이 X-Window는 현실적으로 가장 많이 사용되고 있는데, 특히 엔지니어링 워크스테이션을 중심으로 이용되기 시작하여, 이제는 IBM PC나 Macintosh에서도 사용되고 있다. 5월에서 좀더 소개되겠지만, Display PostScript라는 것도 비슷한 기능을 갖고 있어 주목을 받고 있다 [37].

#### 3.1 국제기구의 표준 CGI

CGI는, 미국에서 VDI라는 이름으로 개발되던 것이, ISO에서 다루게 되면서 그 이름이 바뀐 것이다. Fig. 4는 이 CGI 모델을 보여 주고 있다[32]. 이 표준은 기능정의(Functional specification)와 언어결합(Language binding) 및 데이터 인코딩(Data encoding)으로 구성되어 있다. 좌표계는 2차원의 가상장치 좌표계(Virtual device coordinate : VDC)만을 사용한다.

데이터 인코딩 표준은 Binary, Character 및 Clear text로 나뉘어져 있다. Clear text는 사용자가 읽을 수 있도록 화면상에 표시되는 것이고, Character는 컴퓨터들간에 정보교환을 위해 사용된다. Binary 인코딩을 구성하는 2진수의 비트 패턴(Bit pattern)은 한개의 컴퓨터 내부에서 이용되거나, 특정한 그래픽 장비를 구동하는데 사용된다.

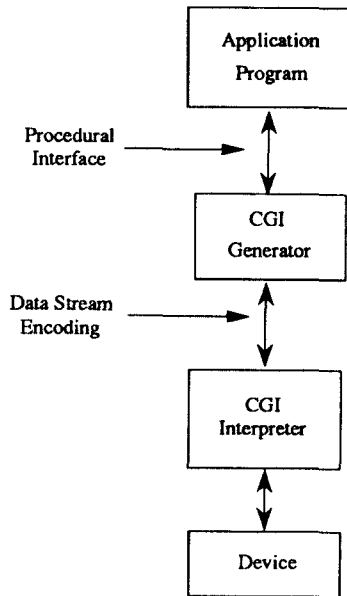


Fig.4 Model of CGI generator and interpreter[32]

3.2 실질적인 산업계 표준 X-Window System

X-Window는 표준을 제정하는 공공기관에서 만든 것이 아니지만, 현실적으로 널리 사용되므로써 저절로 표준과 같이 여겨지고 있다. 개발 초기부터 분산 컴퓨팅(Distributed computing)을 고려하여 설계되어 있으므로, 네트워크 지향적(Network-oriented)인 특성을 갖고 있고, 또한 객체지향(Object oriented)적인 개념도 내포하고 있다. Fig. 5는 이러한 네트워크 지향적인 특성을 보여주고 있으며, 그 기능이 Client와 Server로 분리되어 네트워크로 연결되어 있는 것을 보여준다. (데이터베이스에서 쓰이는 Client와 Server의 개념과는 그 용도가 뒤바뀌어 있음에 주의할 것.)

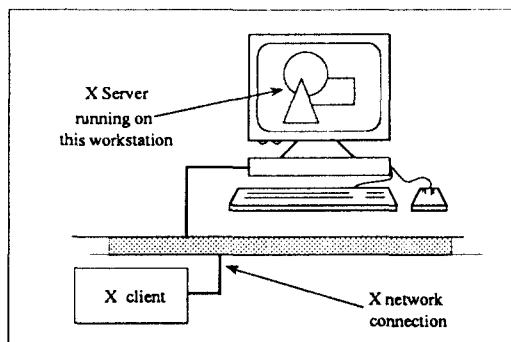


Fig. 5 Configuration of X-window system

이렇게 네트워크를 지향한 구성에서는, Client와 Server간에 정보의 이동을 가능한 적도록 해야 한다. X-Window에서는 필요한 모든 Protocol을 미리 정해 놓고 있는데, Client가 보내는 것을 Request라 부르며, Server가 보내는 것은 Event라고 부른다.

Request는 모두 120개이며 “윈도우 열기” 등을 포함하고 있다. Event는 모든 33개이며, “마우스 버튼 눌러 있음” 등을 포함한다. 이러한 Protocol은 네트워크를 연구하는데 쓰이는 용어이며, 객체지향의 관점에서는 객체간의 Message로 볼 수 있다.

Sever(스크린이 달린 쪽)는 Resource라 불리는 다음과 같은 객체(Object)들을 보유하고 있으며, Client는 Request를 이용해 이 Resource들을 조작하게 된다[27].

- Window : 별도의 작업을 할 수 있도록 컴퓨터 스크린의 일부를 분할한 것
- Graphics context (Rendering attributes와 유사) : 색깔, 선의 굵기 등을 저장해 놓은 자료구조.
- Font : 글씨체
- Color map (Color lookup table과 유사) : 사용할 색깔들을 미리 정해 놓은 것
- Pixmap : 화면정보의 일부를 파일로 저장한 것
- Cursor

4. 그래픽 데이터 교환을 위한 표준

제품의 설계에서 생산까지 그리고 제품 인도 후의 유지보수까지를 고려해 보면, 제품의 관련 정보가 여러가지 자동화 시스템을 거쳐가게 된다. 예를 들어보면 범용 CAD/CAM 시스템에 의해 설계된 제품이, FEM 등의 성능해석을 받게되며, 수치제어 공작기계에 의해 가공되고, 공정관리 시스템도 그 제품의 생산에 관련된 정보를 필요로 한다. 이렇게 다양한 하드웨어와 소프트웨어 간에 제품정보가 전달되어야 하므로, 제품정보를 저장하는데는 표준화된 방법이 필요하다.

Fig. 6은 이러한 정보의 교환을 위한 기준모델을 보여주고 있다[31]. 응용환경(Application Environment)에서는 제품의 각 부분의 기하학적 형상 뿐만이 아니라 연결정보 등 그들의 상관관계들도 유지하게 되며, 그밖에 재질이나 제작에 관련된 정보들도 저장하게 된다. 가상환경(Virtual Environment)에서는 물체의 기하학적 형상들이 기초적인 기하요소들의 순차적인 나열로 저장될 뿐, 그들의 상관관계들은 포함되지 않는다. 하드웨어 환경(Physical Environment)에서는 기하학적 형상은 잊

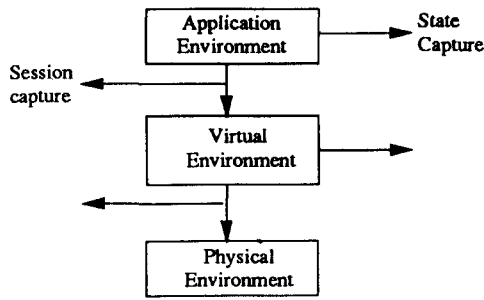


Fig. 6 A reference model for graphical data storage [31]

혀지고, 화면을 구성하는 구성 단위 즉, Pixel 수준의 정보만이 저장된다.

현재 응용환경 단계의 표준으로는 IGES(Initial Graphics Exchange Specification)[11]와 PDES(Product Data Exchange Standard)[19], STEP(Standard for the Exchange of Product Model Data)[22]이 있으며, PHIGS의 저장(Archive) 화일도 여기에 속하는 것이다. 가상환경 단계의 표준으로는 CGM(Computer Graphics Metafile)이 있고, 하드웨어 환경의 표준으로는 TIFF(Tag Image File Format), TRIF(Tiled Raster Interchange Format), ODA(Office Document Architecture) 등을 들 수 있다.

Fig. 7은 자동화 시스템간에 이루어지는 정보교환의 단계들을 보여준다[10]. 제일 초보적인 단계에서는 화일 단위의 교환이 이루어지며, 이 경우에는 구분(Syntax)만을 다루게 된다.

FTAM은 File Transfer, Access, and Management Standard의 약자로 데이터 통신 프로토콜의 표준인 OSI(Open Systems Interconnection)의 일부이다. 그 위의 중간단계에서는 그림과 문자가 함께 교환되며, 이 경우에는 구분뿐만 아니라 의미론(Semantics)도 시스템간에 공유하여야 한다. ODA는 Office Document Architecture Standard의 약자로, 역시 ISO 표준의 하나이다. 최상의 단계에서는 제품정보의 데이터베이스를 교환할 수 있다. 이것은 형상정보 뿐만 아니라, 제품의 설계과 생산에 관련된 모든 정보를 포함하는 것이다.

만일 이와같은 표준화된 교환방법이 없다면, 서로 다른 시스템간의 접속모듈 또는 번역모듈을 마련해야 한다. Fig. 8에서 볼 수 있는 것과 같이 N개의 서로 다른 시스템들이 연결되어야 한다면, N<sup>2</sup>개의 접속모듈을 필요로 한다[18]. 그러나 표준화된 저장방법을 사용한다면, 각 시스템마다 표준화된 화일을 액세스 하는 모듈을 한개씩만 만들면 된다. 즉, 모두 N개의 접속모듈만을 필요로

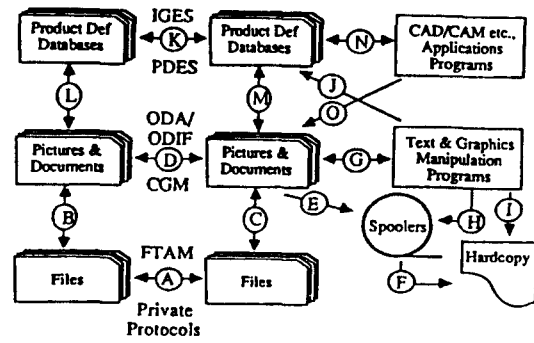


Fig. 7 Levels of data interchange [10]

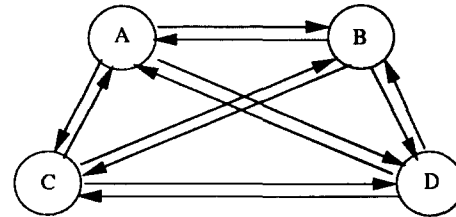


Fig. 8 Direct translation :  $N*(N-1)$  translators required [18]

하게 된다.

#### 4.1 2차원 그래픽 데이터 교환을 위한 표준 CGM

CGM은 1987년에 ISO표준으로 제정된 것으로, 2차원의 정(靜)적인 그림을 저장하는데 사용되는 표준이다. 정적인 그림은 동(動)적인 그림에 대비되는 것으로, 애니메이션 광고의 전체를 Session이라하면 그 중의 한 컷을 말하며, Fig. 6에서는 State Capture라고 했다. 즉, 시간의 차원이 없는 것을 의미한다.

Fig. 9는 CGM 표준의 대상인 Metafile의 구성을 보여주고 있다[31]. 한개의 Metafile은 여러개의 Picture들로 만들어지며, Element는 Metafile을 구성하는 기본 단위이다. CGM은 현재 여러가지 시스템들에서 채택되어 그 사용범위가 넓어지고 있으며, 1990년까지 2차에 걸친 부록이 나와 있다.

#### 4.2 제품모델의 정보 교환을 위한 표준 PDES와 STEP

제품모델(Product model)이란 기하학적 정보만을 갖고 있는 형상모델(Geometric model)에, 생산에 관련된 재질이나 공작순서 등의 정보도 포함하는 것이다.

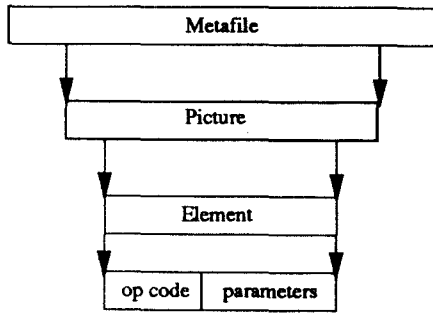


Fig. 9 A metafile at different levels of abstraction [31]

IGES와 PDES는 미국에서 개발된 것으로, IGES는 현재 5차의 개정판이 나와 있고[11], 산업계에서 많이 사용되고 있다. PDES는 IGES의 다음 세대로 볼 수 있으며, 현재는 여러 관련 회사들이 컨소시엄을 형성하여 개발하고 있다. PDES는 IGES보다 내용을 확장하여, 그 적용범위를 넓힌 것이며, 각 시스템 간의 접속(Interface)의 개념을 넘어서 통합화(Integration)를 추구하고 있다. 컴퓨터 통합생산(CIM) 체계를 위해서는 반드시 필요한 표준이다. Fig. 10은 이 PDES의 적용범위를 보여준다 [25].

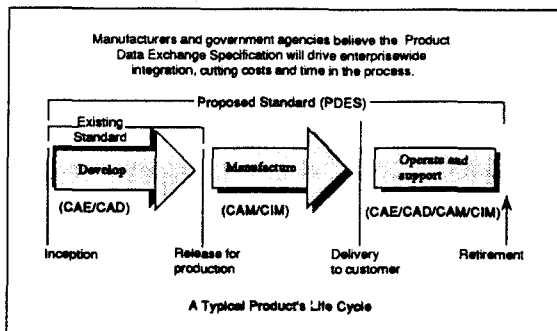


Fig. 10 Integration with PDES [25]

STEP은 ISO의 TC184에서 추진하고 있는 표준이며, PDES를 근간으로 해서 개발되고 있고, 유럽 쪽의 연구 내용도 추가되고 있다. STEP은 기준모델(Reference model)과 형식정의 언어(Formal definition language)로 구성되어 있으며, 하드웨어(Physical), 개념(Logical), 응용(Application)의 3계층(Layer)으로 나뉘어져 있다. 응용계층에서는 각각의 응용분야에 따라 기준모델이 있다.

이러한 기준 모델들이 개념계층으로 번역되고(Mapping), 또다시 하드웨어 계층으로 옮겨지게 된다. 이때 개념계층은 각 제품을 정의 하는데 필요한 정보들을 규정하고 있으며, 이것을 표현하는데 Express라는 형식 언어를 사용하고 있다. 기준모델은 일상적인 자연언어로 표현되고, 이것이 개념계층에서는 Express로 번역되며, 하드웨어 계층에서는 일반 프로그래밍 언어로 변환된다. Express라는 언어의 구성원은 Schema, Type, Entity, Algorithm, Rule이다[28,30].

5. 전 망

Fig. 1은 현재까지 나와있는 ISO 그래픽 표준들 간의 상관관계를 보여주주고 있다. 이 그림에 나타나지 않는 X-Window는 CGI와 비슷한 위치에 속한다고 볼 수 있다. IGES, PDES, STEP은 그 적용범위가 넓어 CGM의 위치에 있다고 하기는 어렵고, 오히려 STEP 등이 CGM의 기능을 포함하고 있다고 할 수 있겠다.

여기까지 소개된 시스템들 외에도 가까운 장래에 그래픽 분야에 영향을 미칠 것으로 예상되는 것들을 간단히 소개한다.

5.1 X-Window에 PHIGS 기능을 포함시킨 PEX

X-Window는 2차원 그래픽 만을 다루고 있으므로, 3차원 그래픽을 다룰 수 있게 하고 모델링의 개념을 도입할 필요가 있다. 이러한 작업이 X-3D라는 이름으로 X 컨소시엄에서 추진되어 왔다. 그러나 PHIGS가 ISO 표준으로 채택되면서 그 사용자가 많아졌으므로, X-Window에 PHIGS의 기능을 포함시키는 쪽으로 바뀌었으며, 그 결과가 PEX이다.

이미 PHIGS를 구현(Implementation)하는데 있어 X-Window를 기반으로 이용한 경우가 여러개 있는데, Fig. 11은 그래픽 전용 워크스테이션인 Stardent 컴퓨터에서의 구현 예를 보여준다. 이러한 구현 예를 "PHIGS in X"라는 용어로 부르고 있으며, PEX는 이것을 더욱 일체화 시키게 될 것이다. 한편, GKS를 X-Window 위에 구현한 것으로 X-GKS라는 것도 있다.

Fig. 12는 PEX의 구조를 보여주고 있다[36]. 이 그림에서 PHIGS+와 PEX라는 덩어리를 제외하면, 기존의 X-Window 시스템의 구조를 나타내게 된다. 여기서 PHIGS+ 부분은 모델링과 Rendering에 주로 쓰이고, 기존의 X-Window는 입력과 사용자 인터페이스 기능을 주로 맡게 된다.



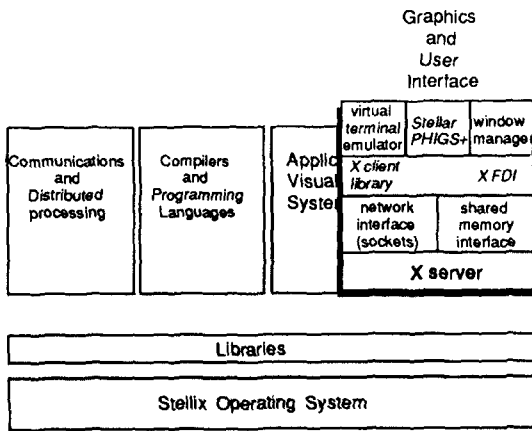


Fig. 11 Stellar graphics software architecture

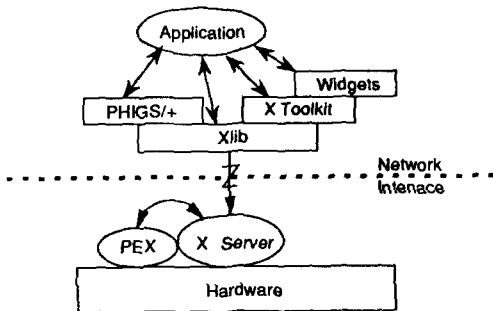


Fig. 12 X/PEX system diagram[36]

5.2 그래픽 사용자 인터페이스를 위한 표준

그래픽 사용자 인터페이스(Graphical User Interface : GUI)는 Macintosh의 Finder나 IBM PC의 Window같은 것이다. 기존의 MS-DOS에서는 문자(Alpha-Numeric)로서 컴퓨터를 운용했으나, GUI에서는 화면의 위치를 지정하는 Mouse만을 가지고 컴퓨터를 운용할 수 있도록 한 것이다.

엔지니어링 워크스테이션의 운용 시스템(Operating system)으로는 문자로만 처리되는 UNIX가 주로 쓰이고 있는데, 현재 X-Window를 기반으로 하여 UNIX용 그래픽 사용자 인터페이스들이 개발되고 있다. AT&T와 SUN 컴퓨터를 중심으로 Open Look이라는 Widget Set이 만들어졌고, 여기에 대응해서 IBM, HP 등의 타 회사들이 OSF(Open Software Foundation)이라는 컨소시엄을 형성하여 Motif라는 Widget set을 만들었다[33].

이 Widget set은 Fig. 12에서 볼 수 있듯이 X-Window 라이브러리를 기반으로 하여 형성한 그래픽 객체(Ob-

ject)들로 이해할 수 있으며, 사용자 인터페이스를 만드는 데 도움을 준다. 예를 들면 Window라든지 그래픽 누름 단추(Button), 또는 슬라이드 숫자 입력기(Slide valuator) 등의 Widget이 이미 만들어져 있어서, 프로그래머들이 요구되는 사용자 인터페이스를 쉽게 구축할 수 있다. 이러한 사용자 인터페이스를 중점적으로 다루는 작업을 위해 UIMS (User Interface Management System)라는 용어가 쓰이고 있으며, 관련 제품도 나오고 있다.

한편, Arnold[32] 등은 ISO의 그래픽 표준과 UIMS 및 Window Manager의 상관관계를 Fig. 13과 같이 설명하고 있다. 즉, GKS 등의 그래픽 표준은 X-Window 위에 구현될 수 있고 (X-GKS와 같이), UIMS는 이 GKS를 이용하여 구축될 수 있다는 것이다. 그러나 Fig. 12에서 볼 수 있듯이, Window Manager인 X-Window는 GKS 수준의 그래픽 기능을 갖고 있고, Widget Set을 UIMS의 한 형태로 볼 수 있으므로, 이러한 구분이 명확하지는 않다.

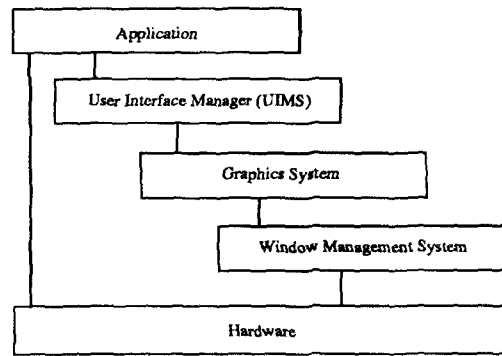


Fig. 13 Relationship between different software layers [32]

5.3 Display PostScript의 가능성

1989년에 시판되기 시작한 NEXT 컴퓨터는, UNIX를 쓰는 워크스테이션을, Macintosh처럼 사용하기 쉽게 만들겠다는 목표로 제작된 것이다. NEXT에서는 X-Window 대신에 Display PostScript를 사용하고 있고, Open Look이나 Motif 대신에 Next Step이라는 GUI를 쓰고 있다.

PostScript는 원래 Apple사의 Macintosh에 연결된 레이저 프린터를 구동하기 위하여 개발된 전용 언어였다 [37]. 이러한 언어를 Page Description 언어라고 부르며, 레이저 프린터의 사용이 증가됨에 따라 그 중요성이

인식되어, 현재는 ISO에서도 SPDL(Standardized Page Description Language)이라는 표준을 만들고 있다. PostScript는 원래 프린터를 구동하도록 만들어진 것이므로 출력(Output)만을 다룰 수 있다. 그러나 사용자의 응답을 다루기 위해서는 입력기능도 필요하게 되며, 이러한 입력(Input)기능을 추가시킨 것이 Display PostScript이다.

6. 국내현황과 선박 설계 및 생산 전산화에의 활용방안

국내의 그래픽 표준에 대한 관심은 아직 큰 활동을 발견하기 어렵다. 외국의 경우에서도 그래픽 표준에 대한 관심이 그리 오래되지 않았으며, 국내에서는 컴퓨터 그래픽 자체에 대한 관심도 최근에야 나타나고 있을 뿐이다. 또한, 표준화의 필요성은 소프트웨어 시스템을 개발하는 경우에, 그것도 그 예상되는 사용자 범위가 넓은 경우에만 느끼게 된다. 컴퓨터 하드웨어와 소프트웨어를 턴-키(Turn Key) 방식으로 도입해서 사용해온 우리의 실정을 감안할 때, 그래픽 표준에 대한 국내의 논문들[8, 13, 20, 21, 23]은 중요한 역할을 한 것으로 여겨진다.

조선분야에 있어서의 그래픽 표준에 대한 관심은 최근에야 부분적으로 나타나고 있는데, 독일 선급협회에서는 STEP을 이용하여 선체구조의 모델링을 시도하고 있다[29, 30]. 이 선체구조모델은 기능별 모델과 조립

단위별 모델로 나뉘어져 있으며, 기하학적 형상정보 뿐만 아니라 생산에 관련된 정보와 건조단계에서 필요한 블록단위에 관한 정보까지를 포함하고 있다.

한편, 스웨덴의 Kockums사에서는 다음 세대의 선박용 전산시스템(Ultimate Shipbuilding System : USS)을 개발하고 있는데, 여기에 OSF/Motif, PHIGS, X-Window 등을 사용할 것이라고 언급하고 있다.

美 해군에서는 RAMP(Rapid Acquisition of Manufactured Parts)라는 시스템을 개발하고 있다[28]. 군함에 탑재된 장비의 부품이 필요할 때 재고가 없을 경우, 이것을 공급받는데까지 걸리는 시간이 평균 300일이 소요된다고 하며, 이것을 개선하기 위하여 RAMP를 개발한다는 것이다. 각각의 장비를 설계할 당시부터 PDES 사양에 맞춰서 작성하면, 그 전산화되어 있는 제품정보를 주문할 때에 이용하고, 생산 공장에서도 그 정보를 제조에 직접 이용할 수 있다는 것이다.

물론, 이러한 표준들이 실제로 활용되기 위해서는, 여러가지 관련 기술들이 종합적으로 다루어져야 한다. Fig. 14는 제품 모델링 기술의 계통도를 보여주며[39], 여기서 STEP 등의 표준이 차지하는 위치를 알 수 있다.

이 글은 앞으로 CSDP 사업에서 사용될 그래픽 표준을 선정하기 위하여 시도된 것으로, 그 목적을 위해 몇가지 제안을 해본다. 그 시도는 많은 위험성을 갖고 있으며, 이 글에서는 그 선정에 대한 타당성을 뒷받침할 자세한 내용을 생략한다.

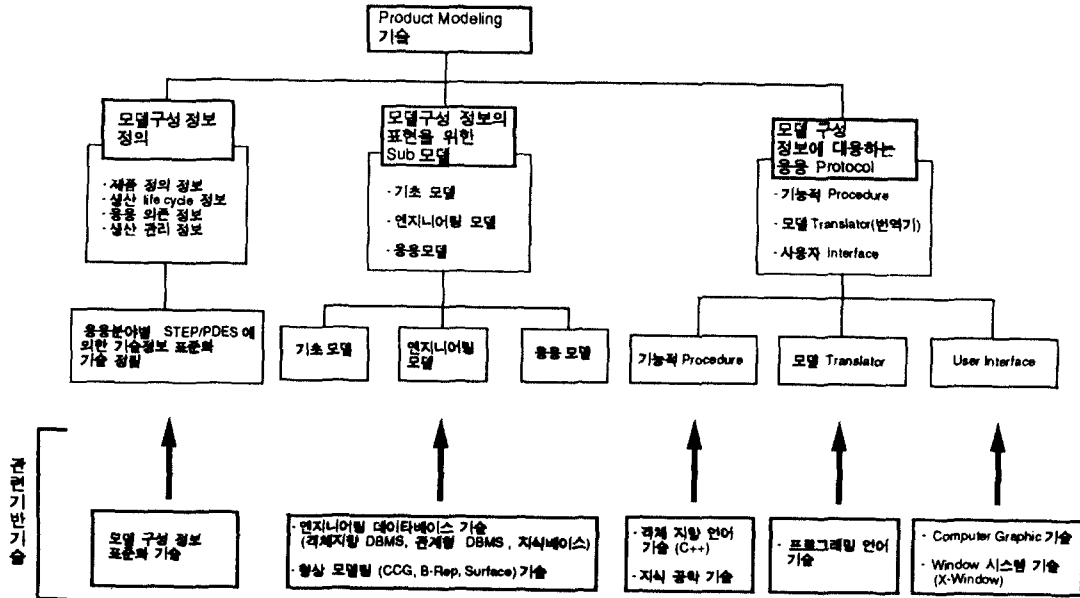


Fig. 14 Technology tree of product modeling [39]

응용프로그램을 위해서는 "PHIGS in X"가 좋은 도구인 것으로 판단되며, 이것은 앞으로 PEX로 대체될 것으로 예상된다. 조선 CIM 개발의 핵심으로 여겨지는 제품모델 (Product Model)을 위해서는 STEP이 중요한 역할을 할 것이며, 조선 CIM용 데이터베이스 설계를 위한 정보도 제공할 것이다. 설계된 모델을 도면화하기 위해서는 CGM과 PostScript를 계속 필요로 하게 될 것이다.

한편, CAD/CAM 시스템 뿐만이 아니라 형상모델링을 필요로 하지 않는 일반 프로그램에서도, 사용자 인터페이스가 중요해 질 것이다. 사용자 인터페이스를 개선하는데는 GUI가 큰 역할을 할 것이고, 여기에는 Motif와 Open Look이 당분간 같이 발전할 것으로 보인다.

## 7. 결 언

이 글은 CSDP 사업에 필요한 그래픽 표준의 전체 윤곽을 보기 위한 시도로, 가능한 넓은 범위를 다루었기 때문에 각각의 시스템에 대한 설명이 부족하게 되었다. 참고문헌을 많이 인용하였으므로 좀더 자세한 자료를 얻을 수 있을 것이다.

공인된 기관(ISO)의 그래픽 표준들 뿐만이 아니라 실질적인 산업계 표준(*de facto standard*)들을 함께 다룬 것은 처음 시도된 것으로, ISO/IEC의 JTC1, ISO의 TC184, X Consortium, OSF 등 서로 다른 조직에서 추진되고 있는 그래픽 표준들이 함께 조사 되었다. 그것은 GKS, PHIGS, CGI, X-Window, CGM, IGES, PDES, STEP, PEX, Motif이다. 이들간의 상관관계를 밝혀 보았으며 특히, CGI와 X-Window가 유사한 기능을 갖는 것이라는 것이 관찰 되었다.

이 글을 통해 앞으로 프로그램들의 이식성을 높이기 위해 사용되어야 할 그래픽 표준이 몇가지 선정되었다. 그것은 PEX, STEP, 그리고 Motif이다. 앞으로도 그래픽 표준의 발전동향을 계속 추적하여야 할 것이며, 더 나아가서는 기술역량을 키워서 국제표준의 제정작업에 참여하여, 우리의 입장을 포함시키고 불이익을 받는 일이 없도록 하여야 할 것이다.

이 글은 과학기술처의 국책연구사업으로 추진되고 있는 '선박설계·생산 전산시스템 개발사업'(CSDP)의 일부로서 수행되었으며, 이 원고를 준비하는데 CSDP 사업단 권재자씨의 도움이 많았으므로 이에 감사드린다.

## 참 고 문 헌

- [ 1 ] "Special issue : Graphics standards", *ACM Computing Surveys*, Vol. 10, No. 4, Dec. 1978
- [ 2 ] "Status report of the graphics standards planning committee", *Computer Graphics (ACM/Siggraph)*, Vol. 13, No. 3, Aug. 1979
- [ 3 ] Bono, P.R., Encarnacao, J.L., Hopgood, F.R.A., Hagen, P.J.W., "GKS-The first graphics standards", *IEEE Computer Graphics and Applications*, Vol.2, No.5, July 1982
- [ 4 ] Brown, M., Hech, M., "Understanding PHIGS - The hierarchical computer graphics standard", Megatek Corporation, 1985
- [ 5 ] Bono, P.R., "A survey of graphics standards and their role in information interchange", *IEEE CG&A (Computer Graphics and Applications)*, Oct. 1985
- [ 6 ] Bono, P. (Ed.), "Special issue on Graphics Standards", *IEEE CG&A*, Aug. 1986
- [ 7 ] Wilson, P.R., "A short history of CAD data transfer standards", *IEEE CG&A*, June 1987
- [ 8 ] 김원순, 김상중, "워크스테이션 구성을 위한 그래픽 표준 및 윈도우 시스템의 검토", *전자통신*, 제9권, 2호, ETRI, 1987
- [ 9 ] Herzog, B., "PEX protocol specification, Version 3.0", PEX Architecture Team, Dec. 1987
- [ 10 ] Arnold, D.B., Bono, P.R., "CGM and CGI - Metafile and Interface Standards for Computer Graphics", Springer-Verlag, 1988
- [ 11 ] National Institute of Standards and Technology, "Initial Graphics Exchange Specification (IGES)", Version 5.0, USA, Sept. 1990
- [ 12 ] Chorafas, D.N., Legg, S.J., "The engineering database", Butterworths, 1988
- [ 13 ] 임경식, 안병준, 최병수, 김상중, "CGI 바이너리 인코딩의 설계와 인터프리터의 실현", 과학기술처 '87 특정연구결과 발표회 논문집, 1988
- [ 14 ] Rix, J., (Ed.), "Special issue on 3D Standards", *Computers & Graphics*, Vol 12, No.2, 1988
- [ 15 ] van Dam, A., "PHIGS+ functional description Revision 3.0", *Computers Graphics*, (ACM/Siggraph), July 1988
- [ 16 ] Carson, G.S., "The future of ISO graphics standards", *IEEE CG & A*, July 1988
- [ 17 ] Encarnacao, J., Rix, J., Schonhut, J., "Open issues and trends on graphics and geometry standards", Wozny, M.J., McLaughlin, H.W., Encarnacao, J. L. *Transactions of SNAK*, Vol. 29, No. 2, May 1992

- (Eds.), 'Geometric Modeling for CAD Applications', Elsevier Science Publishers B. V. (North-Holland), IFIP, 1988
- [18] Ho, C.Y., Tsuei, S.-D., Yang, F.-G., "IGES and PDES, the current status of product data exchange standard", International Conference on Computer Integrated Manufacturing (1st), 1988
- [19] "Product Data Exchange Specification (PDES) - First working draft", National Bureau of Standards (now National Institute of Standards and Technology-NIST), U.S. Department of Commerce, Dec. 1988
- [20] 이원우, 신성용, 김형명, "그래픽스 커널 시스템 (GKS)과 컴퓨터 그래픽스의 제표준", 컴퓨터기술, 대한전자공학회 전자계산연구회, 1988년 12월
- [21] 임경식, 김상중, 강철희, "문서처리를 위한 표준 그래픽 인터페이스의 응용", 컴퓨터기술, 대한전자공학회 전자계산연구회, 1988년 12월
- [22] Wilson, P.R., "PDES STEP's Forward", IEEE CG&A, Mar. 1989
- [23] 이원우, 이성진, 최경일, 백낙훈, 신성용, 김형명, 이철동, "CGI 표준과 PC상에서의 그래픽스 패키지의 구현", 한국정보과학회 그래픽스 및 설계자동화 연구회 학술논문 발표회 논문집, Vol.1, No.1, 1989년 6월
- [24] Rost, R.J., Friedberg, J.D., Nishimoto, P.L., "PEX : A network-transparent 3D graphics system", IEEE CG&A, July 1989
- [25] Francis, B., "Competitiveness : A new standard", DATAMATION, July 15, 1989
- [26] Kroszynski, U.I., Palstrom, B., Trostmann, E., Schlechtendahl, E.G., "Geometric data transfer between CAD systems : Solid models", IEEE CG&A, Sept. 1989
- [27] Jones, O., "An introduction to the X window system", Prentice-Hall, 1989
- [28] Kirksharian, A., "RAMP : A test bed for the Navy's use of PDES product data", *Naval Engineers Journal*, May 1990
- [29] Bronsart, R., "Design and management of a product model", *Schiffstechnik*, Bd.37, 1990
- [30] Bronsart, R., "STEP application model-Ships structural model : Functional submodel", Release 1.1, Germanischer Lloyd, June 1990
- [31] Henderson, L.R., Mumford, A.M., "The computer graphics metafile", Butterworths, 1990
- [32] Arnold, D.B., Duce, D.A., "ISO standards for computer graphics - The first generation", Butterworths, 1990
- [33] Young, D.A., "The X window system : Programming and application with Xt, OSF/Motif edition", Prentice-Hall, 1990
- [34] Henderson, E., et al., "PHIGS PLUS : Advanced three-dimensional graphics with a standard application programmer interface", SIGGRAPH Tutorial Course Notes #9, Aug. 1990
- [35] Reed, T., Chauveau, K.S., Chin, J., "The computer graphics interface (CGI) - The next international graphics standard", SIGGRAPH Tutorial Course Notes #14, Aug. 1990
- [36] Hess, M., et al., "X3D-PEX(PEX) : 3D graphics in a distributed window system", SIGGRAPH Tutorial Course Notes #19, Aug. 1990
- [37] Hourvitzs, M., et al., "The PostScript page description language", SIGGRAPH Tutorial Course Notes #20, Aug. 1990
- [38] "컴퓨터 용어 대사전", 크라운 출판사, 1991년 1월
- [39] 서승완, "CAD/CAM 시스템을 위한 제품 모델링 (Product Modeling) 기술 개발사업-사업 계획서", 해사기술연구소 내부 미출판 자료, 1991년 1월
- [40] Mumford, A. (Ed.), Special Issue : Computer Graphics Standards, *Computer-Aided Design*, Vol.23, No.4, May 1991