

# 전자출판에서의 서체 및 출력기술

林 淳 範

(株) 三寶컴퓨터 技術研究所

## I. 서 론

전자 출판의 처리과정은 크게 3단계로 구분해 볼 수 있다. 원고나 그림을 입력하는 입력 과정(input process), 이를 각 페이지 내에서 원하는 형태로 구성하는 편집 과정(layout process), 그리고 편집된 내용을 볼 수 있는 형태로 인쇄하는 출력 과정(output process)의 3단계이다. 출판의 최종목적은 원하는 출력물을 얻는 것이므로 본 고에서는 이를 연기까지의 과정에서 서체와 출력기 언어가 어떤 역할을 하는지 살펴보기로 한다.

80년대 중반까지는 전자출판에서 입력, 편집, 출력과정에 호환성이 결여되어 있었다. 입력은 코드 변환에 의해 여러 시스템간에 호환이 되었지만, 편집과정 만큼은 각 출력기 시스템에 종속되어 있기 때문에 편집시에는 각 출력기의 제어 방식이나 서체종류에 따라 출력제어 명령을 선택하여 원하는 출력형태를 구성하였다. 따라서 고해상도의 전산사식기나 레이저 프린터를 보유하고 있는 인쇄소 또는 출력소에서 디스켓에 입력되어 있

는 원고를 받아 각 출력기 시스템의 제어 명령을 원고 사이에 삽입한 후 출력물을 인쇄하였다(그림 1 참조).

그후 컴퓨터 기술의 발전에 따라 출력 결과를 편집과정의 화면상에서 항상 볼 수 있는 WYSIWYG (what you see is what you get) 방식의 편집 시스템이 보급되었다. 이에 컴퓨터 성능의 발전이 필요조건이지만 또한 중요한 것은 최종 출력물과 화면사이의 출력이 일치하여야 한다는 점이다. 그리고 이러한 편집 시스템은 각 출력기에 종속되기 보다는 하나의 독립적인 시스템으로 구성되어 여러 출력기에서 같은 형태의 출력물이 인쇄되도록 개발되었다. 이를 위해서는 출력물내의 글자, 그림, 화상을 표현하는 명령이 여러 출력기들과 편집시스템 사이에 호환되어야 하는데, 80년대 중반 이후 발표된 몇가지 PDL(page description language)이 그 역할을 담당하고 있다. Postscript을 대표로 하는 이들 PDL은 글자, 그림, 화상의 표현 뿐만 아니라 소위 운곽선 글자꼴 기술을 포함하여 서체 표현도 풍부하게 되었고, 이들의 interpreter가 많은 출력기에 내장됨으로써 이를 근간으로 한 입력, 편집, 출력과정의 모듈화가 가능하게 되었다(그림 2 참조).

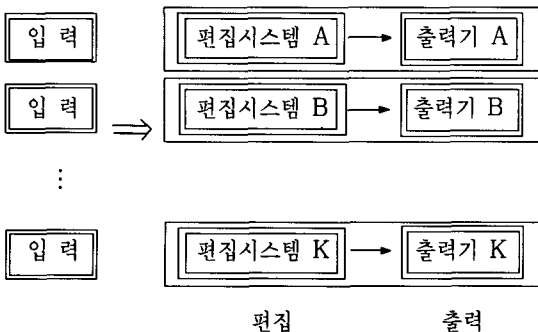


그림 1. 예전방식의 전자 출판 처리 과정

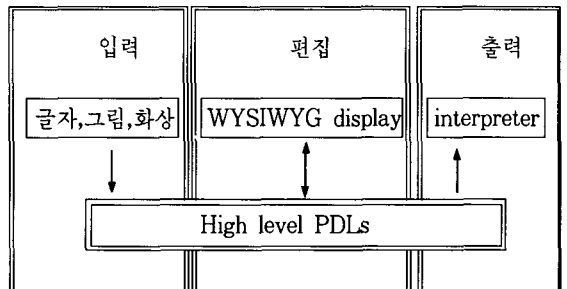


그림 2. 요즘 방식의 전자 출판 처리 과정

PDL이 표현하는 대상중 그림이나 화상의 출력결과 는 입력 데이터와 동일 하지만 글자는 코드로 입력받아 PDL에서 해당 서체의 글자모양을 생성하여 출력하게 된다. 결국 PDL에서 가장 핵심적인 부분이라고 볼 수 있는 글자꼴 처리 기술에 대해서 II장에서 보다 자세히 언급하고, III장에서는 몇가지 PDL에 대해 설명하였다. 그리고 IV장에서는 새로운 추세에 대해 언급하였다.

## II. 글자꼴 표현과 처리 기술

컴퓨터나 출력기에서 글자꼴을 표현하고 처리하는 방식은 기본적인 표현단위에 따라 크게 세가지로 분류할 수 있다. 글자를 그림 또는 화상(image)으로 간주하여 화소(pixel) 단위로 표현하는 점글자꼴(bitmap font), 글자의 윤곽선을 기하학적으로 표현하여 수학적인 직선이나 곡선식으로 구성하는 윤곽선 글자꼴(outline font), 그리고 글자의 구조를 획의 위상(topology)으로 표현하는 구조적 글자꼴(structural font)가 있다<sup>[4],[5]</sup>. 80년대 중반까지의 출력기에서는 주로 점글자꼴 처리 방식이었고 요즘은 주로 윤곽선 글자꼴 처리 방식이며, 구조적 글자꼴 방식은 아직 실용화되어 있지 않다. 따라서 본 장에서는 앞의 두 표현 방식에 대해 자세히 언급하고 어떻게 활용되고 있는지 보기로 한다.

### 1. 점글자꼴 (Bitmap Font)

펜으로 그리는 제도용 출력기를 제외하고 대부분의 전산출력기나 프린터는 마지막에 글자를 점으로 찍어서 출력을 하는데 이 점들을 그대로 표현한 방식이 점글자꼴이다. 이는 가장 간단한 방식으로서 글자의 내부에 해당되는 부분을 1로 하고 외부를 0으로 하여 한글자에 해당되는 사각형을 점행렬식(dot matrix)으로 기억한다.

이 방식의 장점은 글자 제작이 용이하다는 것이다. 특히 종이에 그려진 원도가 있고 이를 출력기에 내장시킬 때 유리한데, 원도를 scanner로 읽은 후 그대로 입력시키거나 필요한 경우 약간의 수정만을 거친 후 입력시키면 된다. 그러나 해상도가 높은 출력기에서 고품위 글자를 출력하려면 기억된 글자꼴의 크기, 즉 점의 갯수가 많아야 한다. 보통 100 \* 100 dot 이상의 크기를 사용하는데, 최소한 글자당 1Kbyte 이상이어야 매우 많은 량의 기억장소를 필요로 한다. 기억용량을 다소간 줄이기 위해서 연속점 길이 방식(run-length coding)이나 방향점 방식(chain-link coding) 등과 같은 호

울적인 표현방식을 쓰고 있다.

이 방식의 단점은 기억용량외에 글자의 크기, 기울임, 회전, 폭의 변화등 여러가지 글자꼴 변형에 대한 제약이다. 크기변화의 경우 축소의 경우는 무난하지만, 확대의 경우는 글자의 윤곽이 거칠어지므로 저장된 크기가 작을수록 출력기의 해상도를 충분히 활용하지 못하게 된다. 글자의 기울임이나 회전의 경우도 처리가 가능은 하지만 결과의 품질이 떨어지고 처리 시간이 매우 많이 소요되므로 출력기에서 이러한 변형은 거의 사용되지 않았다. 이와 같은 단점들을 해결하고자 윤곽선 글자꼴이 개발되었고 프로세서의 처리 속도가 증진되어 출력기에 윤곽선 글자꼴이 가능하게 되었다.

### 2. 윤곽선 글자꼴 (Outline Font)

윤곽선 글자꼴 방식의 레이저 프린터는 1985년 Apple사가 Postscript을 탑재한 LaserWriter를 발표함으로써 널리 보급되기 시작하고, 그후 많은 출력기 회사들이 Postscript을 출력기 PDL로 채택하여 윤곽선 글자꼴 기술이 각광 받고 있다. 윤곽선 글자꼴은 글자의 윤곽을 여러부분으로 나누어 각각 직선, 원호, 자유곡선등으로 표현하는데, 자유곡선으로는 주로 베지어(Bezier)곡선, 3차윤형(cubic spline) 곡선, B 윤형(B-spline) 곡선등의 수학적인 표현이 주로 사용된다. 이들 자유곡선은 몇개의 제어점(control point) 만으로 표현이 가능한데, 예를 들어 하나의 베지어 곡선은 4개의 점으로 정의되며, 이때 바깥쪽의 점이 곡선상의 양 끝점과 동일하고 중간에 있는 2개의 제어점이 곡선의 모양을 결정한다.

이 방식으로는 글자의 크기나 기울기, 회전등의 변화를 모두 표현할 수 있어서 하나의 크기에 대한 데이터만 필요하므로 기억용량이 매우 감소하고, 고해상도의 출력기 일 수록 글자의 품질은 더욱 정교해 진다. 윤곽선 글자꼴의 변화는 기존의 사진식자에서 렌즈가 처리해 주는 부분이 모두 가능하므로 전산 출력기에 사용하는데 매우 알맞은 표현방식이다. 이는 글자의 윤곽에 대한 직선과 곡선부분의 좌표값만 기억하고 글자를 변형할 때는 이들 좌표값에 기하학적 변형 행렬식(geometric transformation matrix)만 곱해주면 되기 때문이다. 한편 글자를 출력할 때 이들 좌표로부터 윤곽선의 직선이나 곡선을 계산하여 점으로 그려주고 그 내부를 점으로 채우는 과정이 필요하므로 속도가 느려진다.

### 3. 글자꼴 기술 경쟁

Postscript을 개발한 Adobe사를 둘러싸고 1989년 봄

소위 “폰트전쟁 (font war)”이 시작되었다. Apple사의 레이저 프린터에 탑재되어 널리 보급된 Postscript은 일반기준을 훨씬 넘는 금액을 기술료(royalty)로 요구하고 있었는데 Apple사와 Microsoft사가 이에 대항하여 TrueType과 TrueImage를 개발하여 사용하겠다고 발표한 것이 폰트전쟁의 발단이다<sup>[7]</sup>. 이에 Adobe사는 그동안 대외비로 하고 있던 Postscript의 글자표현 형식인 Type 1 font format을 공개하여 여러 출력기 회사들이 자사 기술을 보다 용이하게 사용토록 하였다<sup>[1]</sup>. 한편, 1990년 봄에는 전세계 레이저 프린터 시장에서 60% 이상의 시장 점유율을 가지고 있는 Hewlett Packard사가 윤곽선 글자꼴을 포함하는 새로운 PCL 5를 발표하였다<sup>[3]</sup>. 따라서 폰트전쟁은 Adobe사의 Type 1, Microsoft사의 True Type, 그리고 HP의 PCL 5에서 사용되는 Intellifont가 주축이 되어 치열한 경쟁을 하고 있는 상황이다. 여기에서 보듯이 윤곽선 글자꼴 기술과 이를 표현하는 PDL이 사무용 레이저 프린터나 고해상도 출력기에서 가장 핵심적인 기술 중 하나이다.

### III. Page Description Languages (PDL)

#### 1. PDL의 역할

I장에서 언급한 바와 같이 WYSIWYG 편집 시스템은 PDL을 통해 출력기에 필요정보를 보낸다. 하나의 편집 시스템이 여러 출력기에 호환성을 가지려면 문서의 구성이 출력기 제어 명령 무관하게 편집 가능해야 하며 입력된 화상이 출력기 해상도에 무관하게 출력 가능해야 한다. 이런 목적을 위하여 출력기와 편집시스템 사이에 공통된 protocol이 필요하므로 PDL이 그 역할을 담당하는데 중요한 기능은 다음과 같다.

- 출력기나 편집시스템의 해상도등 device에 independent 해야 한다.
- 글자(text), 그림(graphics), 화상(image)을 다룰 수 있어야 하고, 이들이 자유 자재로 회전(rotation), 확대 축소(scaling), 기울임(shearing) 등의 변형(transformation) 될 수 있어야 한다.
- 한 페이지내 임의의 위치에 이들 편집 대상이 놓일 수 있는 페이지 개념을 포함해야 한다.

PDL은 이런 기능들을 표현하는 instruction set으로서 프로그램 언어 또는 escape sequence 형태를 가진다. 편집 시스템에서 입력 데이터에 페이지 구성 정보가 추가되어 문서 화일에 저장되고, 이 화일은 선택된 출력기에 따라 편집 시스템에 포함되어 있는 driver 프

로그램에 의해 적절한 PDL 명령으로 변환된 후 출력기로 보내져서 해석이 된다(그림 3 참조). 이를 위해서는 출력기 내에 PDL 명령을 해석하는 interpreter가 있어서 출력기 해상도에 맞도록 각 페이지의 image를 재생시켜 주어야 하는데 이 프로그램을 보통 raster image processor(RIP)라고 한다. 출력기 내에는 한 페이지에 해당되는 bitmap RAM이 있고, 출력기에 보내진 PDL 명령에 따라 RIP가 해당되는 위치에 글자의 image나 graphic image 등을 만들어 RAM에 채운 후 한 페이지가 완성되면 이 RAM의 내용이 출력되어 인쇄된다.

가장 대표적인 PDL은 단연코 Adobe 사에서 제작한 Postscript이고, 그 외에는 Interpress, ImPRESS, DDL, PCL 5, TrueImage 등이 있다. 많은 PDL들은 70년대 중반 XEROX PARC에서 개발된 Interpress라는 언어 또는 거기에 사용된 기술을 모태로 발전하였다. 최초로 상업화된 시스템은 1981년 Imagen사에 의해 제작된 ImPRESS 언어인데, 이 당시 시장에서 PDL 환경이 성숙되어 있지 않아서 편집 시스템이나 소프트웨어 개발자에게만 관심이 국한되었다.

한편 Postscript 언어의 개발은 소형 PC-즉 Apple의 Macintosh 시스템에서 고품위 저가 프린터의 필요가 근간이 되었다. 여러 종류의 프린터에서 글자나 그림의 묘사와 출력에 호환가능토록 하는 표준 PDL 언어가 요구되었던 것이다. 많은 사용자들이 최종 인쇄물을 고해상도의 사식기나 레이저 프리터에서 출력하기 전에 저가의 프린터에서 먼저 출력하여 확인해 보기를 원하였다. 1984년 Postscript이 처음 발표되었을 때의 시스템 구성은 Apple Macintosh, 300dpi Postscript 레이저 프린터, 그리고 100dpi 이상이고, Postscript이 내장된 Linotype 100과 300 계열의 사식기였다.

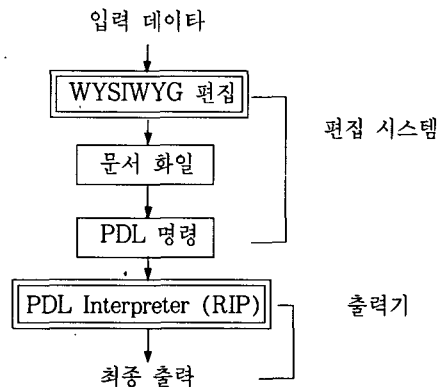


그림 3. PDL 흐름도

## 2. Postscript<sup>[1,2]</sup>

Postscript 역시 근원은 XEROX PARC에서 개발한 PDL에 있으며, 그 형태는 완벽한 프로그래밍 언어로서 FORTH라는 언어와 표기법이 유사하다. 그래서 Postscript 언어는 일반 user가 프로그램 작성에 사용하기도 하는데, 보통의 상황에서는 편집시스템등 응용 소프트웨어의 driver 프로그램이 Postscript 프로그램을 생성해서 출력기에 보내준다. 이에 반하여 일부 다른 PDL들은 프로그램 언어 형태라기 보다는 escape sequence와 같이 출력기 명령의 정형화된 set이므로 일반 user가 직접 사용하는데 적합치 않다.

Postscript이 Interpress에서 발전하였지만 가장 큰 변화는 글자를 다루는데 있어서 bitmap으로 처리하지 않고 graphic으로 간주하여 처리함으로써 윤곽선 글자꼴이 가능해졌다는 점이다. 이것은 RIP에 저장된 각 서체에 대해 4 point에서 127 point까지 어떠한 크기의 글자도 하나의 명령으로 효과적인 처리가 가능해진 것이다. 점글자꼴로 처리되는 PDL에서 같은 품질을 얻으려면 각 크기별로 글자 데이터를 가지고 있어야 하므로 많은 storage를 필요로 한다. 더욱이 Linotype 100 사식기와 같은 1270dpi의 출력기는 300dpi의 프린터보다 16배 이상의 storage가 필요하지만 Postscript에서는 Postscript 언어로 묘사된 하나의 글자꼴 데이터가 모든 크기를 생성할 수 있다. 한편 윤곽선 글자꼴의 속도저하 문제는 bitmap 글자 cache라는 기법을 사용하여 어느 정도 극복하였다. 한번 생성된 글자의 bitmap image는 일정한 용량의 RAM에 기억된 후, 같은 서체의 같은 크기 글자가 사용될 때는 여기에 기억되어 있는 bitmap을 그대로 사용하므로써 속도를 개선하였다. 그러나 cache RAM의 용량이 적거나 같은 글자의 사용빈도가 작을 경우는 효과가 줄어들다.

Postscript에서 글자꼴을 표현하는 형식은 2 가지가 있는데 Type 1 과 Type 3이다. Type 3는 사용자 정의 형식으로서 일반 Postscript 언어로 표현되며 해석도 같은 방법으로 한다. 이에 반해 Type 1 형식은 Postscript 프린터에 내장된 글자꼴에 사용되는 것으로서, 일부 특수 명령을 제외하고는 같은 방법으로 표현이 되지만 Postscript에서 해석할 때 "BuildChar"라는 특수 routine으로 처리된다. 그리고 일부 특수 명령은 "hinting" 등 글자 처리의 특수 기능에 관련된 것으로 Adobe사의 글자 생성 알고리즘과 관계가 있다.

## 3. ImPRESS와 DDL<sup>[6]</sup>

ImPRESS는 상업화된 최초의 PDL로서 Imagen사에

서 사식기용으로 특별히 개발되고 60여개의 소프트웨어의 지원이 있었다. 그러나 Postscript은 여러 회사의 출력기에 RIP이 탑재되어 널리 보급된 것에 반하여 ImPRESS RIP은 자사에서 개발된 출력기에만 탑재함으로써 널리 보급되지 못하였다. 한편 ImPRESS 언어는 속도를 중요시하여 설계되었지만 몇가지 단점이 있는데, 점글자꼴을 사용하였고 많은 량의 memory를 필요로 한다는 것이다.

이러한 단점을 극복하고자 Imagen사는 DDL(document description language)이라는 새로운 언어를 개발하였다. DDL은 ImPRESS의 모든 기능외에 추가로 몇가지 기능이 첨가되었는데 대형 문서의 처리가 가능하도록 독립된 페이지 개념을 도입하여 매 페이지가 독립적으로 출력이 가능하다. 또한 그래픽 기능을 중점적으로 강화하고, 글자는 점글자꼴 뿐만 아니라 윤곽선 글자꼴도 처리가 가능하도록 하였다. DDL 언어는 읽기 쉽고 코드가 짧으므로 응용소프트웨어 개발이 용이하며 통신에도 유리한 장점이 있다.

## 4. PCL 5<sup>[3]</sup>

Hewlett Packard(HP)사에 개발된 PCL(printer control language) series는 자사의 레이저 프린터인 LaserJet 계열에 탑재되어 발전해 왔다. PCL 1 부터 PCL 4 까지는 PDL 이라기 보다는 serial printer의 제어 명령을 위한 escape sequence set이었지만 PCL 5 부터는 Postscript에 대하여 PDL 개념의 여러 기능들을 추가하여 1990년 LaserJet III 부터 탑재하여 보급하고 있다.

기존의 PCL 4 에서부터 추가된 주요 기능은 윤곽선 글자꼴 처리, 그래픽 기능, 화상처리 기능의 보완 등이다. 윤곽선 글자꼴은 Agfa Compugraphic사의 Intelifont 형식을 채택하였으며, 그래픽 기능은 기존의 plotter용 언어인 HP-GL/2의 기능을 그대로 포함하였고, raster image에 대해서는 몇가지 새로운 압축 방법을 사용하였다. 사무용 레이저 프린터 시장에서의 HP사의 점유율이 매우 크므로 PCL 5가 이 시장에서는 널리 보급이 되었지만, 고해상도의 출력기등 타사의 제품에 아직은 널리 탑재되어 있지 않아서 전자출판용 PDL로서는 잠재력이 있을 뿐 아직 Postscript 만큼 활성화 되지 않고 있다.

## IV. 통합된 편집시스템

완벽한 개념의 WYSIWYG 시스템이 되려면 출력될 내용을 그대로 화면에 보여주는 것이지만 이것은 해상도의 차이로 불가능하다. 그러나 RIP의 기능이 편집시스템과 같이 host computer에 있어서 화면에서 편집하는 도중에는 화면의 해상도에 맞추어 display할 페이지 image를 만들고 출력시에는 출력기 해상도에 맞추어 image를 만들어 보내주면 가능하게 된다(그림 4 참조). 현재 사무용 또는 교정용의 300-600dpi 프린터에서는 이와 같은 방식을 사용하기도 하는데 1000dpi 이상의 고해상도 출력기에서는 일반 표준 HW interface와 RIP 소프트웨어로는 전송속도 문제가 심각하다. 이 경우에는 프린터나 사식기와 video 전기신호 또는 출력장치 제어신호를 직접 보낼 수 있는 특수 HW를 제작하여 신호 전송에 사용하는 경우도 있다. 앞으로 이와 같은 전송기술이 더욱 발전하면 편집 시스템내에 RIP 등을 통합하는 방법이 전자 출판 분야에 널리 보급될 것이다.

현재 이와 같은 방법은 Postscript을 중심으로 전개되고 있다. 1990년 화면 제어 명령을 포함한 display Postscript 등 기존의 기능을 보완하여 Postscript level 2가 발표되었다. 이를 이용하여 최종 출력물과 같은 형태의 문서를 화면에서 자유자재로 볼 수 있고, 완성된 문서는 host 내에 있는 RIP를 통하여 출력 페이지의 image가 완성된 후, 출력기에 맞은 video 전기 신호로 직접 변환되어 빠른 속도로 전송되어 인쇄된다. 이의 가장 대표적인 시스템은 Next 시스템 또는 SUN의 NewsPrint 시스템으로 빠른 처리속도를 요구한다.

다른 하나의 방향은 Microsoft Windows를 중심으로 발전하고 있는 소프트웨어 폰트 매니저들을 이용하는 것이다. 기본적으로 Windows 시스템은 응용 소프트웨어와 출력장치 사이에 독립성을 제공해 준다. 즉 Windows 내부에는 PDL에서와 같은 출력제어 명령과 user interface를 제어하는 명령등을 표현하고 처리하는 기능이 있어서 응용 프로그램은 오직 Windows 시스템만 접촉하고 그 위에서 수행이 된다. 그리고 화면이나 특정 출력기등의 정보는 device driver형태로 Windows에 제공되므로 모든 응용 프로그램은 Windows를 통하여 공통으로 출력기에 접속된다. 이런 추세하에서 그동안 출력기의 서체를 공급하던 여러 회사들이 자사의 글자꼴들을 Windows에서 처리가 가능하도록 한 것이 소프트웨어 폰트 매니저이다. 대표적인 것으로는 Adobe의 ATM(Adobe type manager), Bitstream의 FaceLift 등이 있고 Windows 자체에도 TrueType 형식의 글자꼴 처리 기능을 제공한다.

사용자가 원하는 글자꼴의 원하는 서체를 Windows 시스템에 install 하면 여러 응용 프로그램에서 이 글자들을 항상 사용할 수 있게 되고 출력시에는 각 폰트 매니저에서 글자의 image를 생성하여 해당 출력기의 interface에 보내준다. 그러나 이러한 시스템들은 현재 사무용 시스템을 위주로 개발되어서 전문 출판용 고해상도 출력기에는 제약점이 있다. 편집이 끝난 후 출력기에 전송되는 데이터는 각 페이지에 해당되는 bitmap image이므로 300dpi 근처의 사무용 프린터에는 효과가 크지만 앞에서 언급한 바와 같이 고해상도 일수록 전송 시간이 많이 소요되므로 효과적이지 못하다. 그러나 소프트웨어 개발과 출력기의 개발이 독립적으로 이루어질 수 있으므로 기술발전에 따라 전자 출판에도 널리 보급될 가능성이 충분하다.

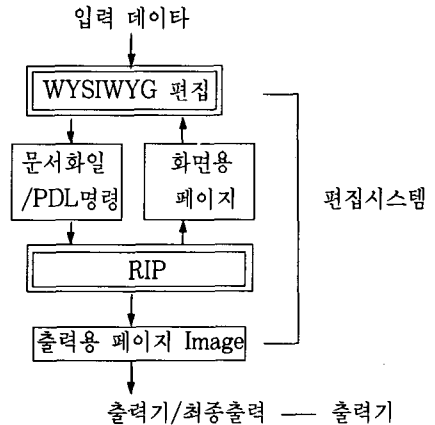



그림 4. 통합된 편집시스템

### V. 결 론

본 고에서는 전자출판에서의 출력관련 기술을 설명하였다. 그중 글자꼴과 PDL이 가장 중요한 요소임을 살펴보고, 이들의 처리기술이 발전함에 따라 전체 출판시스템의 구성이 효율적으로 변천함을 보았다. 앞으로도 Postscript을 중심으로한 PDL 기술, 글자꼴 처리 기술, Windows에서의 출력기능 등을 계속 주시하여 전자출판 시스템의 개발 뿐만 아니라, 사용자 입장에서의 선택에도 도움이 될 수 있도록 관심을 가져야 할 것이다.

參考文獻

- [ 1 ] Adobe Systems Inc., Postscript Language Reference Manual, 2nd Ed., Addison Wesley, New York, 1990.
- [ 2 ] Adobe Type 1 Font Format, Adobe Systems Inc., 1990.
- [ 3 ] HP LaserJet III Printer Technical Reference Manual, Hewlett Packard, 1990.
- [ 4 ] 임순범, “글자체 설계 및 자동 생성 시스템의 개발”, 폰트 개발과 표준화 워크샵 발표논문집, 한국 정보 과학회 SW 공학 연구회, PP. 3-6, 1989.
- [ 5 ] 임순범, “윤곽선 글자꼴의 처리 기술 및 활용 추세”, 전자공학회지, 제18권 제11호, pp. 69-76, 1991. 11.
- [ 6 ] A.L. Oakley and A.C. Norris, “Page Description Languages : Developments, Implementation and Standardization”, Electronic Publishing, vol. 1, no. 2, pp. 79-96, Sept. 1988.
- [ 7 ] True Type Font Files, Microsoft Corporation, 1990. 

筆者紹介



林 淳 範

1959年 1月 30日生

1982年 2月 서울대 계산통계학과(학사)

1983年 8月 한국과학기술원 전산학과(석사)

1992年 2月 한국과학기술원 전산학과(박사)

1989年 2月 ~ 1992年 2月 (주) 휴먼컴퓨터 이사

1992年 3月 ~ 현재 (주) 삼보컴퓨터 기술연구소 실장

주관심분야 : 컴퓨터그래픽스, Font, LBP, Printer, 전자출판