Accountable Color in Network Applications

Richard A. Holub*

Color, Control, E-commerce, Electronic Color Proof, Network

Abstract: Production in the publishing/printing industry is distributed in that it usually occurs at several sites which share and communicate page and image data. The ICC profile standard has enabled important steps toward portable color. However, additional concepts, data structures and tools are necessary to insure consistent color in a network. The concept of an electronic color proof (ECP) is explained. ECP resides in a network linking nodes involved in various phases of graphic preparation and production. It mediates the sharing of information about the capabilities of nodal color devices, the interpretation of color image data to the devices and the control of color reproduction by the devices to a common or a negotiated criterion. ECP is separable from image data and may have local (to a node) and shareable parts. Examples illustrate the importance of gamut data among components of ECP.

Introduction

In outline, this paper will cover new demands placed on systems by network computing. It will review conventional publishing workflow with regard to color and changes needed to move to digital workflows that rely on accountable color in networks. An overview of software and hardware aspects of an Electronic Color Proofing system will be presented, including flow diagrams, software architecture and instrumentation concepts. The paper will conclude with practical examples.

Analogous Developments

The rapid emergence and evolution of the Java programming language highlights the new demands placed on software in a networked environ-

---

* IMAGICOLOR

ment (Horstmann & Cornell, 1997.) Some comparisons between ECP and "the programming language of the World Wide Web" are instructive and motivate a discussion of ECP.

Capabilities of Java that make it well-suited for web applications include
1) accessibility of objects with equal ease whether they are across the net or part of a local file system,
2) security features which innoculate against viruses and tampering; these complement extensive error checking and automatic memory management that contribute to overall *robustness*, and
3) architecture-neutrality through which a Java compiler produces output which has nothing to do with a particular computer architecture but can be translated into native machine code by specific interpreters. The term "*portability*" comes to mind, in this regard, but is used with a different connotation in the Java community.
I will touch on these in considering some of the distinguishing properties of ECP.

Conventional Workflow

The production of printed color, as a manufacturing process, can be thought of as having two phases. In a first phase, the product is designed and prototyped in an iterative way. Then the prototype is fashioned into a tool or die which is used to stamp out pieces in volume. In the foregoing respects, printing has much in common with other discrete manufacturing processes such as those used to make textiles or plastic parts. The process is distributed in two respects. First, the steps involved in preparing the prototype are, traditionally, at least, carried out at a number of sites. The sites often are housed in different organizational and business entities. Second, the mass production often occurs at a number of facilities; this is likely to be increasingly true in the future as just-in-time (also known as Print-on-Demand) production paradigms mature.

Figure One illustrates the foregoing thoughts. The lines connecting the various agents in the process represent channels of typical communication. Since so much of what is printed is advertising, it is useful to explain the figure in terms of ads. A print-buyer (advertiser) usually approaches an agency who designs the ad and subcontracts the detailed development of graphical and image components of the ad. The agency recommends a series of publications whose readers are good targets for the ad and establishes a relationship with their publishers. It usually contracts out the pre-press work to a separator, or "engraver." The distinctions among these business categories are increasingly blurred as printers, publishers and agencies all bring both creative and pre-press functions in-house.

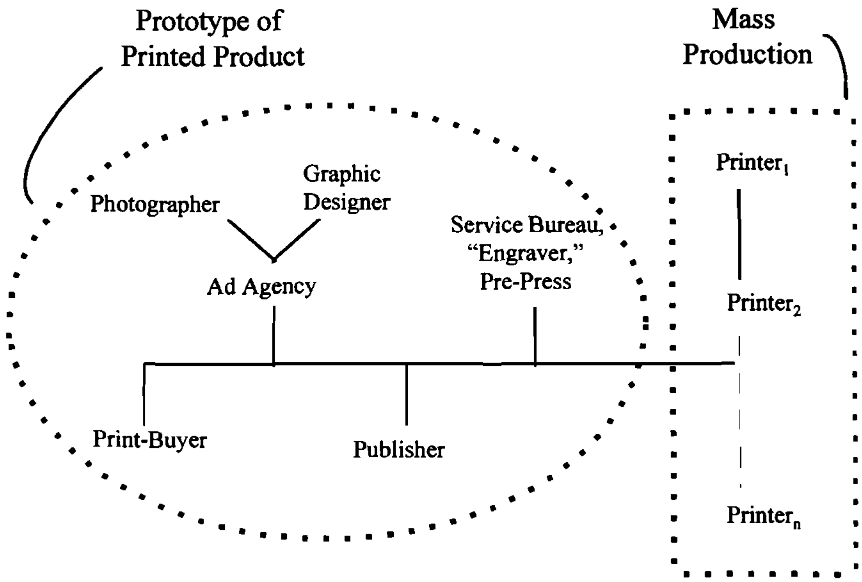

Figure One: is a schematic of the organization of the publishing and printing industry. Some of the functions shown as distinct business activities are being absorbed into other business entities. Lines connecting entities suggest a conduit ("sneakernet") for the proof. Conventionally, the proof has been a physical instance of the current working copy of the prototype. The conduit will be replaced by an electronic medium for conveying proof.

The goal of the people and processes to the left in Fig. 1 has been to prepare a contract proof, the prototype of what the mass-produced pieces should look like. The prototype is then sent on to the printer, along with the films that made it, to serve as a model for production. Although pressure to change the foregoing workflow is growing, it remains fairly prevalent. Color approval, traditionally, consists of movement of the physical instances of iterations of the proof among the various players.

Recent proofing technologies are breaking the connection between the separation films that have been used to make the production tool and the proof. CTP is eliminating the separation films. Even in the good old days, the correspondence between the colors of the proof and those created with the tool in production was tenuous, often necessitating a re-negotiation of color approval at press-side.

The electronic color proof depends on a computer network - the lines in Fig. 1 link the various prototyping and production nodes electronically. It depends on being able to coordinate a variety of devices of potentially different sorts so that the color rendered at different sites is a credible substitute for passing around a particular physical instance of copy. ECP enables work in the prototyping phase to be done with an awareness of what can be realized in production. This paper explores the considerations involved in achieving accountable color for e-commerce.

Automatic Control of Color Reproduction in a Network of Devices to a Common Criterion

This section reviews three general considerations regarding automatic control in a network.

1. Device Independence

Automatic control in a network setting is best implemented in a device-independent and, preferably, a colorimetric framework. One of the goals of the developers of the CIE Standard Observer was to create a system in which human color matching behavior could be simulated instrumentally. It may be that neither of a pair of individuals at different nodes in a proofing network match colors exactly as the Standard Observer, even if both have normal color vision. However, as long as the instruments involved in calibrating their devices are consistently calibrated and, preferably, match colors as would the Standard Observer, the devices can be brought to a common state. Ideally, the observers will then debate differences in their color perception or aesthetic preferences, given that the devices produce very nearly the same output, physically. This is analogous to architecture-neutrality in Java.

2. Accuracy and Robustness

In the situation described in (1), the word "ideally" is important. Two devices at different nodes may each employ ICC (International Color Consortium) profiles in rendering. Although the profiles are "portable" (architecture-neutral) and color translations are mediated by a device-independent "connection space," the output from the two devices may appear very different for at least two reasons: a) The profiles do not embody an accurate model of the devices' color reproduction or do not reflect the current state of one or both devices or b) The devices have different gamuts for any of a variety of intrinsic or extrinsic reasons and/or the profiles exploit the gamuts in very different ways. In (a) the problems are related to device variability or profile inaccuracy. In (b) the problems have to do with differences in the way colors are reproduced when they cannot be reproduced exactly.

Device variability can be addressed by continual calibration and verification, preferably by instruments "built-in" to the rendering devices. This will be addressed further in what follows.

An example of the effects of profile inaccuracy may be found in Komori's (1997) Master's thesis. He reported results for one commercial color management application, but obtained similar results with other software packages in pilot work. In carefully controlled experiments conducted with a sheet-fed press in the School of Printing at RIT, he questioned whether a color managed work flow would improve the extent to which a pre-press proof forecast a press run. He used an analog proofer (which employs separation films) and a digital proofer ("dye sublimation" technology,) both in native mode and with the benefit of color management.

Komori calibrated the press and the proofers and prepared profiles to make the proofers predict press sheets on a subsequent, controlled press run. The agreement between proof and press sheet was better, for both types of device, in native mode than it was with the benefit of custom profiles. It is likely that most of the problem lay with the accuracy of the device models underlying the profile. Komori's results demonstrate clearly that portable color is not the same thing as accurate network color.

Intrinsic factors affecting gamuts include things like colorant sets and internal states of device calibration that affect tone reproduction or gamma in each of the colorant channels. Temperature and humidity may affect internal states of calibration. Extrinsic factors affecting gamuts include things like viewing conditions, ambient illumination and the algorithms employed to render unprintable colors. Exchange of information about and resolution of gamut differences of devices is a vital function of Electronic Color Proofing.

Holub (1997) reviews methods of modelling devices which have been shown to be accurate (Holub and Kearsley, 1989) to within the repeatability of the device. This is desired if a User at one node is to be confident that she appreciates colors as they will appear at another node. This corresponds to the design principle of Java that objects should be accessible with equal ease whether they are across the net or part of a local file system. Robustness is assured by extensive, automatic error-checking of color measurement and modelling results and by procedures for verifying the adequacy of color transformations.

3. Ease-of-Use

The word "automatic" has two senses in this discussion. First, an instrumental simulator of the Standard Observer automates the process of color matching. Second, an instrument which calibrates itself and requires a minimum of human involvement to do its work is automatic in the sense of "easy-to-use." Instruments that meet the second condition are more likely to be used by a system for continual calibration and are more likely to be accurate, both because they will be used correctly and will be in calibration when they are used. The next section should lend concreteness to the foregoing concepts.

Desirable Instrumental Properties

Figure Two shows an instrumental configuration, that will be discussed at length in another paper, which addresses the foregoing concerns. A Cathode Ray Tube Display is shown in cross section as viewed from the side. However, the ideas apply to any video display technology. The configuration has the following, desirable features:

1) It supports unattended calibration; the ease-of-use requirement is satisfied,

2) Viewable area is not occluded during measurements (non-contact,)
3) It enables measurement of ambient light reflected from the screen of the display,
4) Continual monitoring of the effects of ambient and of calibration parameters is possible and
5) It is self-calibrating.

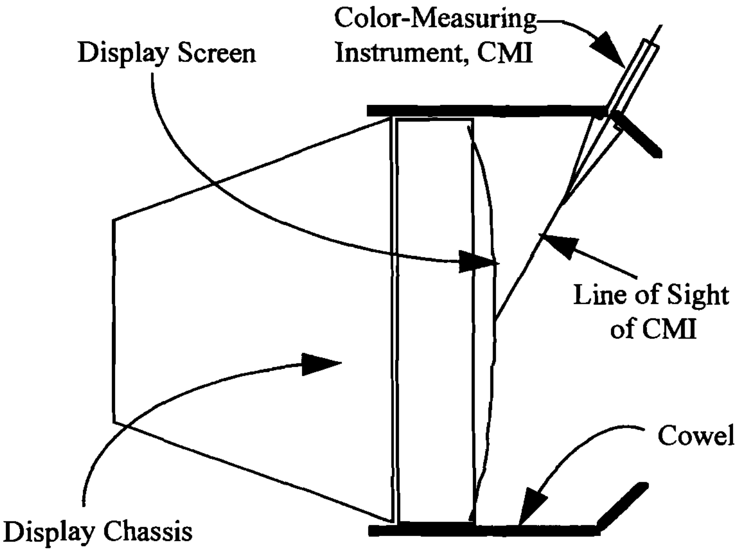

Figure Two illustrates desirable properties of color measuring instrument in a video display application.

Prototyping Nodes and Production Nodes

I hesitate to distinguish between prototyping and production nodes in the Electronic Color Proofing network because the distinction is not fundamental. However, Figure Three shows a piece of the network connecting two nodes. One might be at a digital photographer's studio in Connecticut while the other is at an art director's office in New York City. Both video and hard copy proofing devices are shown at each site; sites are linked by a network connection. In this application, both sites would probably have a general purpose computer. However, the system can work with an intelligent rendering device that is connected to a general purpose computer only through a network interface.

Much of the cycle time could be drained from the traditional workflow if photographer and art director could confer over a soft proof of the photo subject, each at her own site, with confidence that they were seeing substantially the same thing.

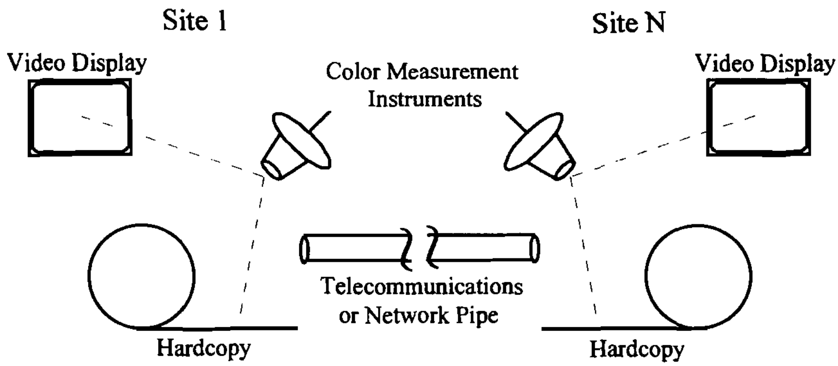

Figure Three illustrates two nodes in an electronic color proofing network with the kinds of rendering devices that might be present. Although a single color measuring instrument is shown measuring both devices, at each node, it is more practical to associate specialized instruments with the rendering device, as shown in Fig. 2. Software/visual calibration techniques might be used, but are less automatic and less accurate, if only due to individual differences from the Standard Observer.

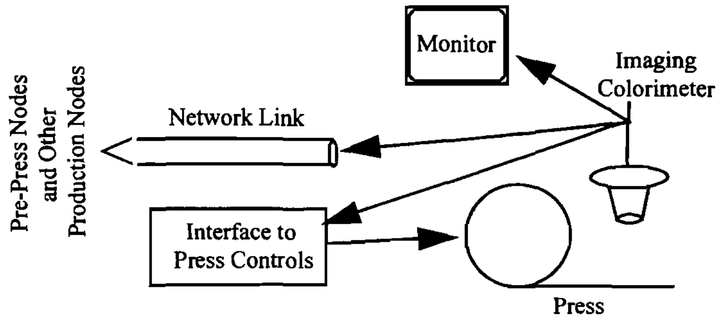

Figure Four illustrates a node that is involved in volume production. It suggests that an imaging colorimeter is employed to monitor the color reproduction of the press.

Figure Four depicts one of a series of nodes on the production side of Fig. 1. Electronic Color Proofing provides two important functions:
1) coordination of production to a common criterion that is established in the production network automatically or is negotiated, primarily by exchange of gamut information of the presses and
2) communication of data expressing the capabilities of the production equipment to prototyping nodes. In time, this may result in "remote press checks" in which agents for the print buyer qualify color over the net. More immediately, soft proofing at creative nodes benefits from current information about the state of the presses.

An imaging colorimeter is shown in Fig. 4, suggesting the advantages of analyzing color in the image area of press sheets during production. It is a device which apprehends CIE TriStimulus Values (TSVs,) or values that can be converted to TSVs by way of a 3X3 matrix of constant coefficients, at each pixel of the image. Holub (1997) discusses a resolution-independent way of analyzing color errors in the reproduction; i.e., the analysis of errors is independent of the spatial resolutions of printed and reference images. The design of an imaging colorimeter having desirable properties discussed earlier will be the subject of a future paper.

A Flow Diagram For Electronic Color Proofing

A profile (as specified in the ICC standard) can be thought of as a translator or interpreter of color data to a device. Like structures have been known (Neugebauer, 1957) and used for decades. It is a compact way of representing essential, image-wide rendering information, independently of the image data. The idea of modifying an image rendering by editing the color translator, rather than the image data itself, goes back at least as far as Wellendorf (1982) and the basic approach was used, in a device-independent framework, in the Designmaster 8000 (Masia, 1984.)

An important aspect of Electronic Color Proofing is to transmit in the proofing network only the essential information regarding image interpretation. This is particularly important when the interpretation is the subject of negotiation or interactive negotiation (two or more parties are on-line simultaneously, conferencing over a desired color reproduction.) Therefore, the ECP should be *separable* from bulk image data and may consist of *local and shareable constituents*, so as to minimize the amount of data to be transferred across the net in order to make the system work. As will be seen, EC Proofing involves more than sending device profiles back and forth across the net.

Applying the foregoing thoughts to the situation of on-press inspection discussed at the end of the last section, it is preferable to transmit, to prototyping nodes, a color to color' translator which expresses the state of the production equipment rather than the image as captured from the press sheet, in most cases. The color to color' translator then becomes part of the process of interpreting the image data to particular rendering devices at one or more creative nodes in such a way that the appearance of press sheets is adequately forecast. Consistent with the application described at the end of the last section, the image being proofed at the prototyping node need not be on the press, but merely one destined for the given network of presses in the near future.

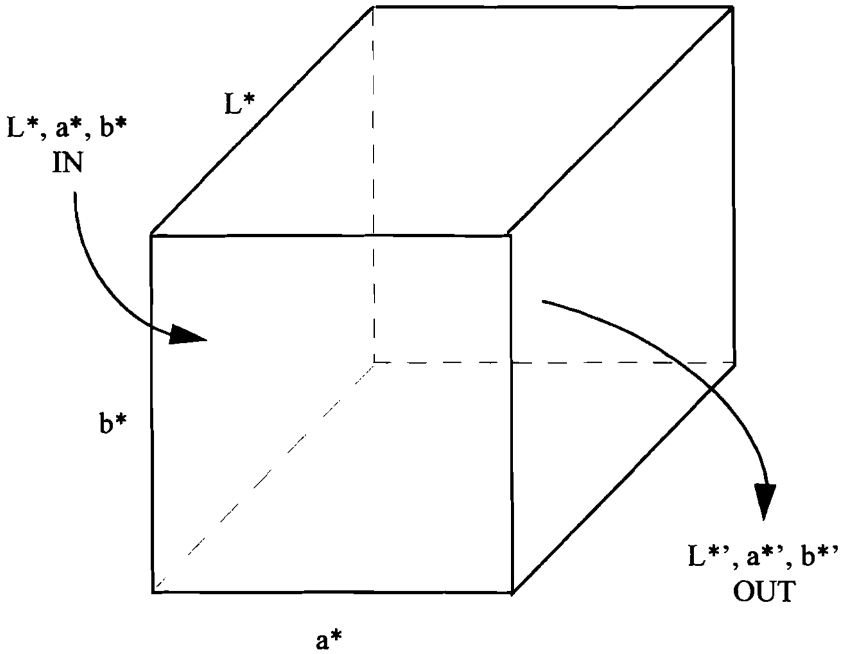

Figure Five illustrates an implementation of a color-to-color' converter.

The color to color' translator is a kind of profile. However, its essence is not conversion of image data to or from specific device coordinates. It translates colors from values in one device-independent coordinate system into modified values in the same coordinate system, although coordinate conversion could be included in the processing. An illustration of an implementation by Look-Up Table (LUT) is shown in Fig. 5. Holub (1997) details varied applications of color to color' transformations along with algorithms for computing them with accuracy sufficient to support accountable color in networks.

Figure Six, A and B is a flow diagram of a comprehensive process. It is complicated, but it should be realized that most of the processing can be automatic and transparent to the User.

At the top of Fig. 6, one or more Users invoke the application software; a single User can run the program in order to revise ECP as it relates to the local node, or to other nodes which are accessible. Multiple Users may run the program simultaneously to negotiate the ECP interactively, extending the functionality of remote annotation programs such as imagexpo (trademark of Group Logic, Inc.) Network readiness is established by making sure that the relevant, participating nodes all share current data. Color Measurement Instruments (CMIs) are put through (preferably automatic) calibration checks.

Then, verification of device calibration is attempted by rendering and analyzing color. The details of this step depend on the nature of the device and of the CMI. If the device is a press, particularly one with fixed information on the plate, the verification is most likely to involve "live" imagery rather than a predefined calibration/verification form consisting of tint blocks. The color error data produced by verification are supplied to the press controls, if appropriate, as well as to the program to support decisions about whether color variations can be compensated by modification of existing rendering transforms of the ECP or whether recalibration of the device is required.

If recalibration ("from scratch") is required, branch to C in Fig. 6B, where systematic characterization of the device is pursued. In contemporary parlance, "characterization" includes analysis of a device's color mixing properties, in addition to processes such as linearization which are often referred to as "calibration."

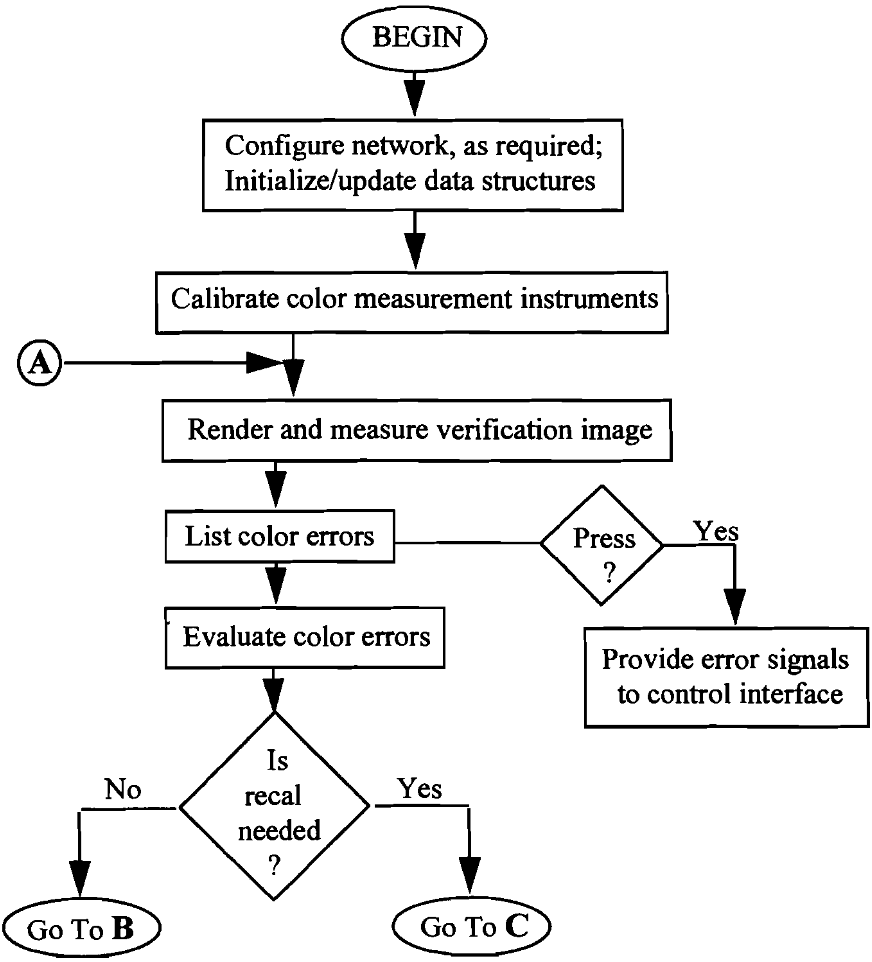

Figure Six A is the first half of a flow chart of an Electronic Color Proofing process. **B** and **C** point to continuations in Figure Six B.

If color errors are not so great as to require complete re-calibration (or "re-characterization,") branch to **B** in Fig. 6B, in order to revise a color-to-color' transform which is a part of ECP. This has the effect of modifying the translators, so that rendering at network devices served by ECP produce consistent colors. Data regarding User-preferences include Gray Component Replacement specifications, customization of gamut operators and the like. All the foregoing may be modified through a User-interface provided in the application software. At **D**, in Fig. 6B, these preferences are consulted and incorporated into ECP, as needed, to effect changes in rendering consistently across the network.

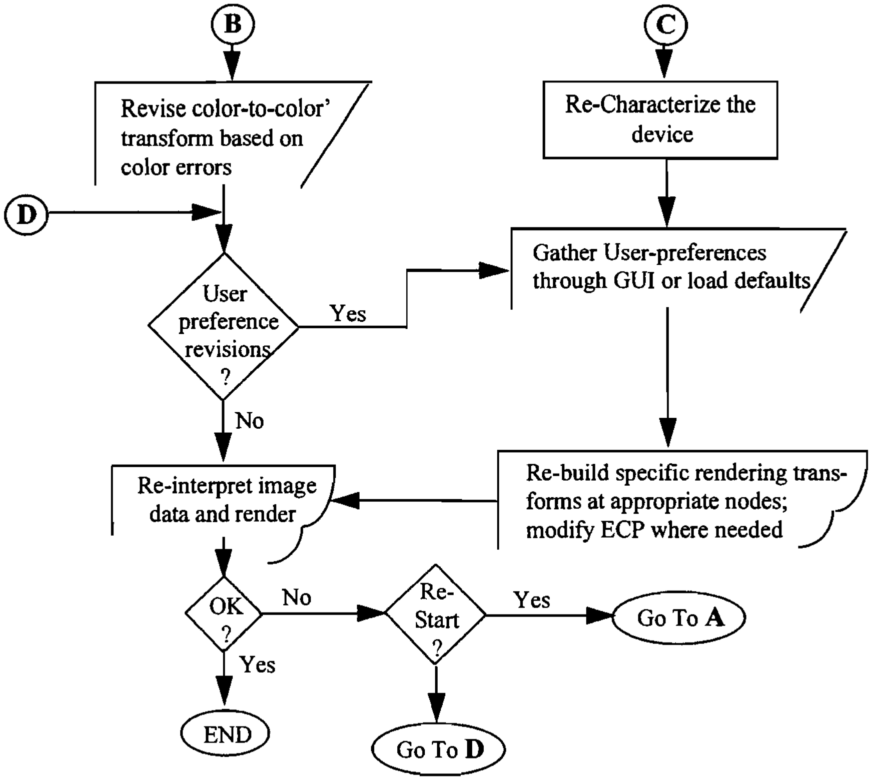

Figure Six B is the second half of a flow chart of an Electronic Color Proofing process. Go To **A** (bottom right) points to a return indicated at the upper left of Figure Six A.

The end result is modification of the rendering transforms maintained in the local portion of ECP at each active node. If the results of rendering are satisfactory (cf., "OK?" in Fig. 6B,) the process concludes. Satisfactoriness depends on the circumstances, but may involve verification.

File Manifestation of ECP

The data structures appropriate to ECP are suggested by an object-oriented model of color devices (see Holub, 1999) and by a TIFF-like manifestation outlined in this section. Other file structures are possible; TIFF was chosen due to its flexibility and extensibility.

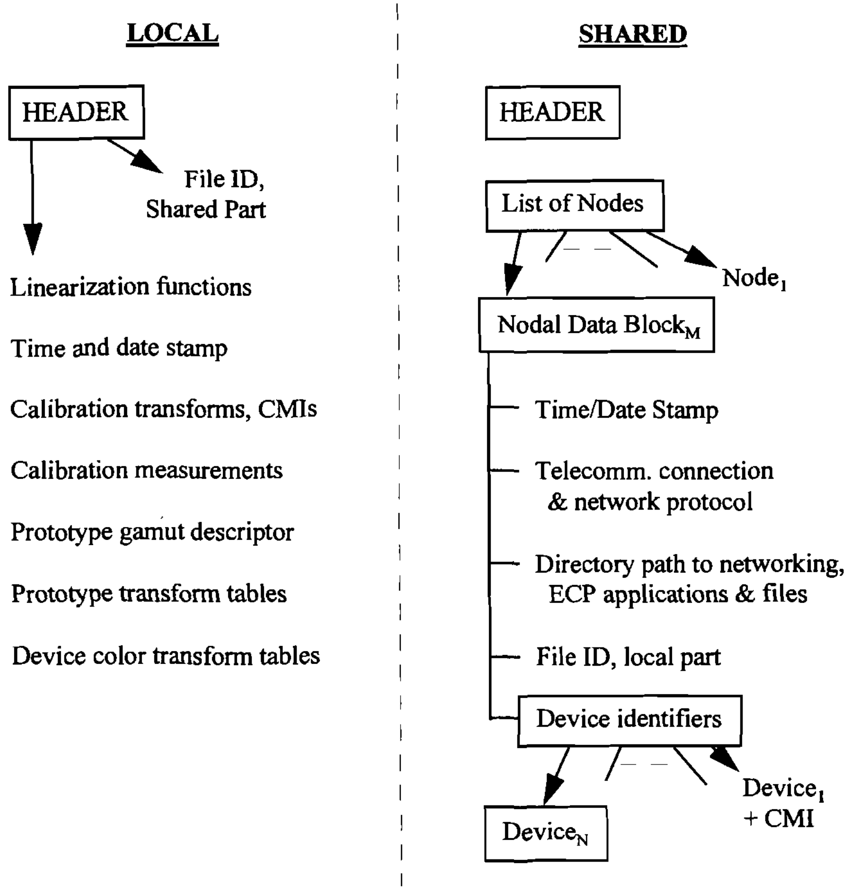

Figure Seven illustrates the organization and constituents of a file manifestation of ECP.

TIFF describes a means of image storage and ECP can certainly be embedded in an extended TIF format along with image data. However, it is preferable that the ECP be separable, so that the elements which control the interpretation of color image data can be modified and moved in the network without having to move all the relevant page and image data. It is also desirable that ECP have local and shareable constituents for the same reason. However, the distinction between local and shareable is not fundamental; some constituents are much more likely to be shared than others, but any constituent might be moved in the network for one reason or another.

To the application software which administers the Graphical User Interface Software, both operating at a node, ECP is a set of data structures based around the classes Node and Device/Transform (Holub, 1999.) The data structures map onto a general file structure which is not bound to a particular operating system, computer platform or processor. Object-oriented conventions of data-hiding are employed in the software to insure overall security and the integrity of transformations which are manipulated at a node.

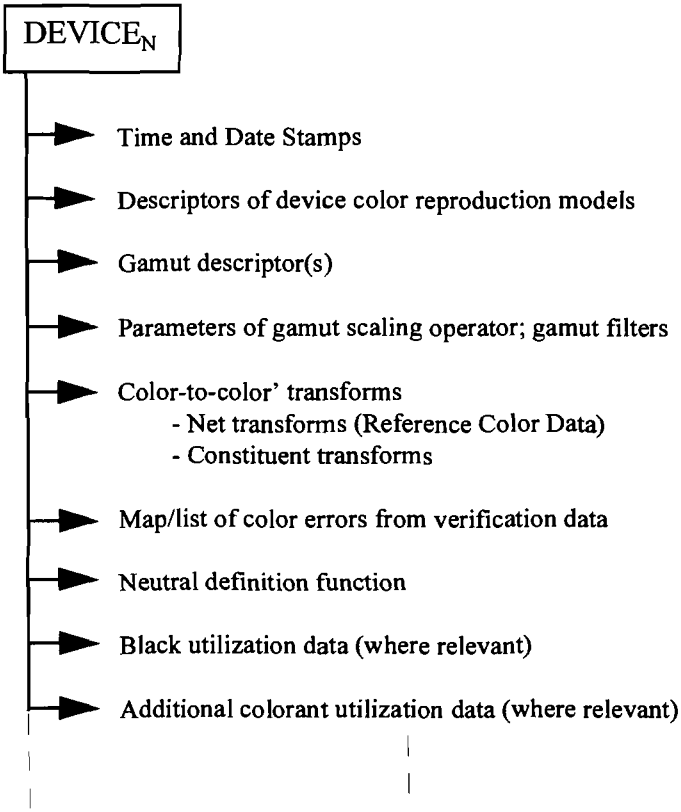

Figure Eight is a continuation of Fig. 7, zooming in on some of the shareable attributes of the nth device's calibration/characterization.

Accordingly, the ECP files which store the data and transformations have local and shared components, as noted previously. These are highlighted in Figures Seven and Eight. Shared components consist of data which are read-only to all nodes except the one assigned responsibility for the data. During initialization, in preparation for Electronic Color Proofing as outlined in Fig. 6, participating nodes insure that appropriate data are written to the relevant nodal fields.

There is not space in this forum for an extended discussion of the constituents of the files. Two of importance to Electronic Color Proofing are the device gamut descriptor and the record of operations performed in the course of gamut scaling. Either may have image specificity, which is represented by a field in the ECP file not shown in Figs. 7 or 8. The gamut descriptor is a representation of a three-dimensional solid whose surface points consist of the most saturated colors reproducible (or sensible, to be general) by a device. Diagrams of this sort date back at least to Rydz and Marquart (1954.) It is convenient to store values of maximum Chroma as a function of Hue and Lightness where the coordinates are those of one of the CIE's uniform color spaces.

Gamut scaling involves a color-to-color' transformation controlled by certain parameters. The record(s) of parametric values of the mapping function(s) needed to relate the gamuts of two (or more) devices are useful to establishment of a common aim or control point for color reproduction on a variety of devices in a network. The gamut filter alluded to in Fig. 8 is a simple data/file structure enabling gamuts of two or more devices to be compared as part of the process of establishing common control points.

Summary with Examples

An overview of a system for coordinated control of color reproduction in a network of diverse devices has been presented. The system is based on technologies of color measurement, networking, color calibration and characterization, graphical user interface and software architecture. The system has been shown to meet the design goals of device-independence and (computer-) architecture neutrality, accuracy and robustness, ease-of-use and suitability for network applications. The following examples illustrate the usefulness of the system in contemporary applications.

Example One: In a distribute-then-print operation, it will be desirable to have consistency of color of production across the nodes. One strategy for defining a color set point for production control is to use the least common gamut, i.e., the smallest print gamut that can be realized by all the devices in the network. This may be determined automatically by operations on gamut filters or can be negotiated interactively by using visualization tools with the filters.

Example Two: A merchant sells clothing over the World-Wide-Web. Images of apparel are distributed from its web site and it is desired that consumers get a realistic sense of color or, at least, be advised of color inaccuracies that might influence buying decisions. Among other things, this requires that the intended gamut of reproduction be compared with the gamut of the consumer's monitor (and/or printer.) The latter depends on prevailing viewing conditions (measurable with the configuration of Fig. 2) as well as on intrinsic properties of the device. In one form, the negotiation would compare the intended with realizable gamut, possibly by way of gamut filters, and indicate to the User what intended colors are unavailable on his device. In a second form, the User might be advised to modify ambient illumination conditions until the entirety of the intended gamut can be realized.

Literature Cited

Holub, R. A. (1999) An object-oriented model of color devices. These *TAGA Proceedings*.

Holub, R. A. (1997) System for distributing and controlling color reproduction at multiple sites. Pending U.S. and International patent applications. International publication No. WO 97\PCT 34409.

Holub, R. A. and W. Kearsley (1989) Color to colorant conversions in a colorimetric separation system. SPIE Vol. **1184:** 24-35, K. Sayanagi, Ed., Neugebauer Memorial Seminar on Color Reproduction.

Horstmann, C. S. and G. Cornell (1997) Core Java. Vol. 1: Fundamentals. Upper Saddle River, New Jersey, USA: Sun Microsystems Press / Prentice Hall, pp. 630.

Komori, Y. (1997) The performance analysis of color management systems: Simulation of colors from sheetfed offset press by analog and digital proofer. Master's Thesis, School of Printing Management and Sciences, Rochester Institute of Technology, Rochester, NY, pp. 230.

Masia, A. (1984) A digital color separating system based on principles of colorimetry. *TAGA Proceedings*, pp. 346-361.

Neugebauer, H. E. J. (1957) "Color correction selector," U. S. Patent No. 2,790,844.

Rydz, J. S. and V. L. Marquart (1954) "Photographic and photomechanical aspects of electronic color correction," *TAGA Proceedings*, pp. 139-150.

Wellendorf, K. (1982) Method and a device for recorrecting standard color corrections in a color picture recording. U. S. Patent No. 4,328,515.