

An Introduction to Color Management Systems

Mutoh Advanced Engineering Group

Color Management Overview

- **The color reproduction problem**
 - Reproducing color for visual continuity
- **Fundamentals of color management**
 - Device-independent color, device profiles and color transformations
- **ICC profile format & limitations**
- **Color management systems**
 - ColorSync, ICM and PostScript
- **System, platform & device issues**

The Color Reproduction Problem

- **Devices, operating systems, applications and device drivers all interpret and reproduce color differently**
 - **Printing technologies**
 - Restricted color gamut, high spatial resolution
 - **Display technologies**
 - Different technologies, different color gamuts
 - **Capture technologies**
 - Illumination dependent, various technologies
 - **Applications and viewing conditions**
 - The “relativistic” nature of color

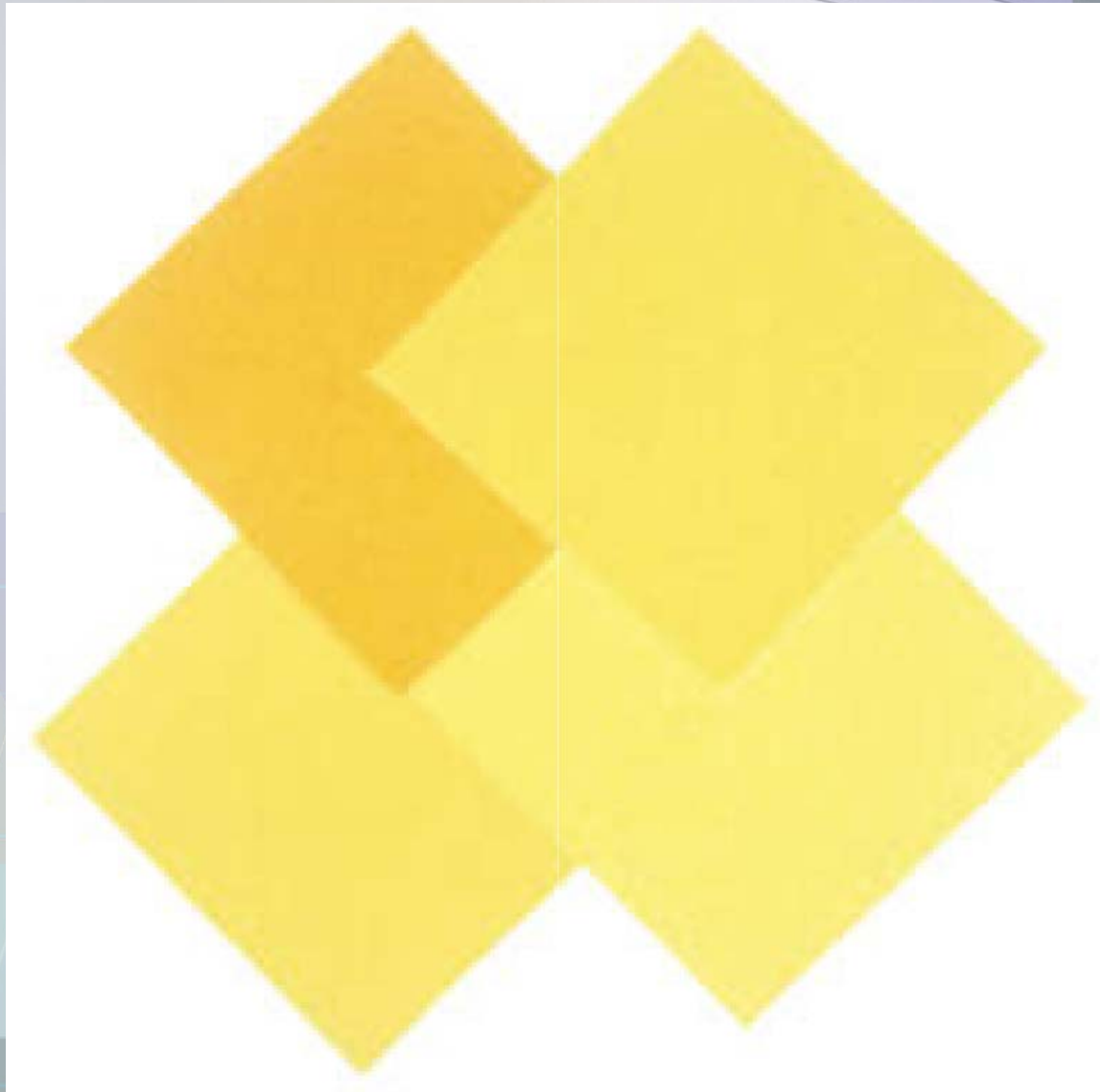
Everyday Examples

- **Print same image on two different printers: different ink, different media**
- **Display same document on two different monitors - technology**
- **Scan same document on two scanners**
- **Display same graphic image in two different applications**
- **Display same graphic image in office lighting vs. tradeshow exhibit center**

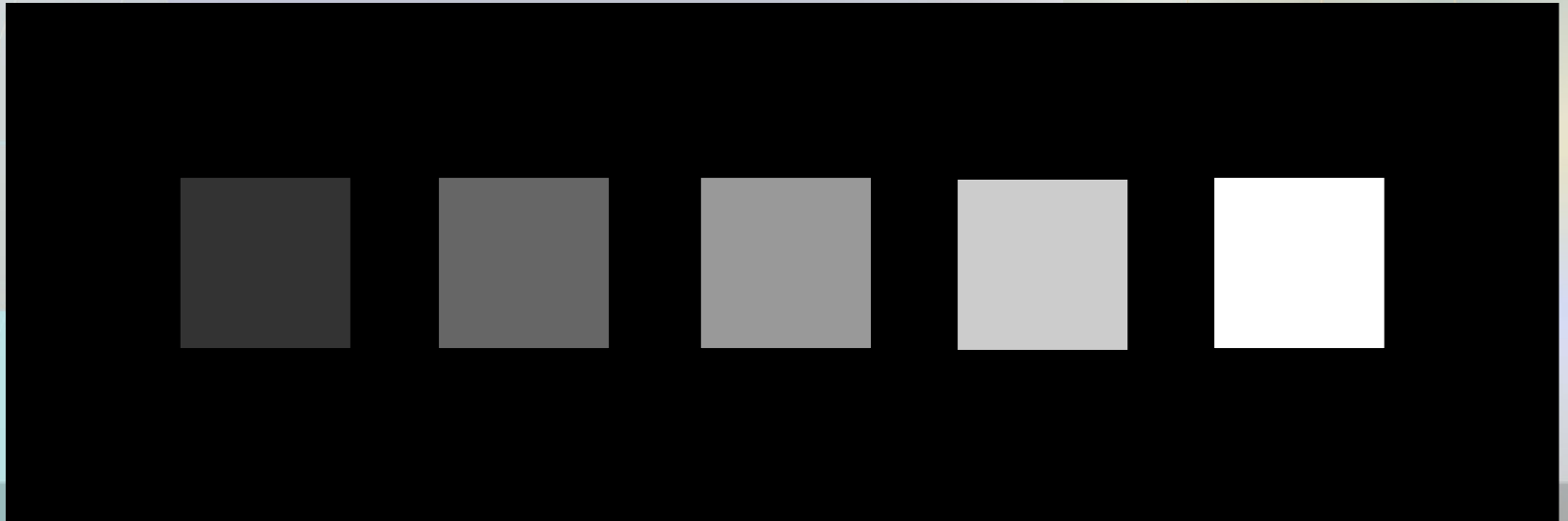
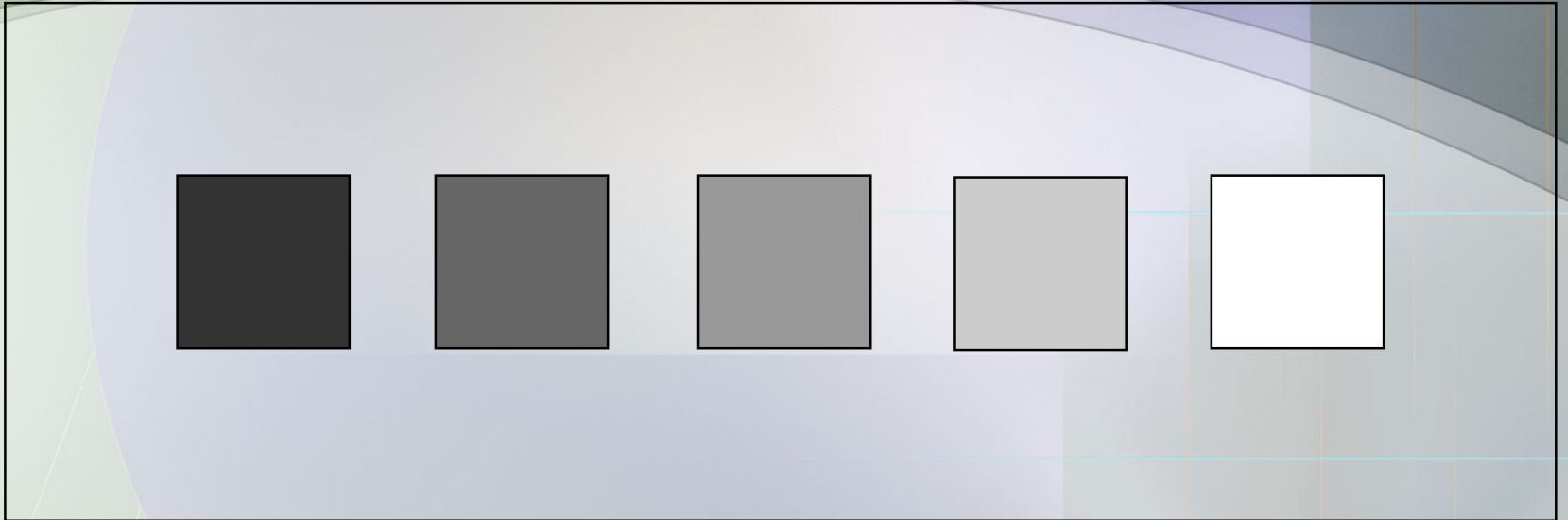
Display Example: Television



Printing Example: Ink & Media



Viewing Condition Example



The Fundamentals Of Color Management

- **Device-independent color**
 - Visually accurate color reproduction independent of technology
- **Profiles**
 - Models of how device / technology captures, displays or reproduces color
- **Color transformations**
 - Mapping color from one device to another to achieve the “best possible” visual match

A Transformation Is Needed

- **Accounts for color characteristics of source and destination devices**
 - Example: monitor vs. printer
- **Three approaches**
 - Device *dependent* color
 - Device *independent* color (two methods)
- **All three can give virtually identical results**
- **All three have tradeoffs...**
 - How do you choose the best approach?

What is a Transform?

- **R,G,B is a digital representation of color**
 - **“Devices” are physical technology**
 - **Electronic (CRT, LCD) and/or electromechanical**
 - **Color is a “physical” phenomenon**
 - **Digital values must be translated into physical phenomenon (color) by physical hardware**
 - **Physical limitations of different hardware result in translation of digital values into different physical colors**
 - **R = 256 = red, but red on a monitor is not the same red as that produced by a printer, or another monitor**
 - **“Transformation” is translation**
 - **Original digital values must be translated into new digital values that will reproduce the desired physical color on the target device, so that original and reproduction match**

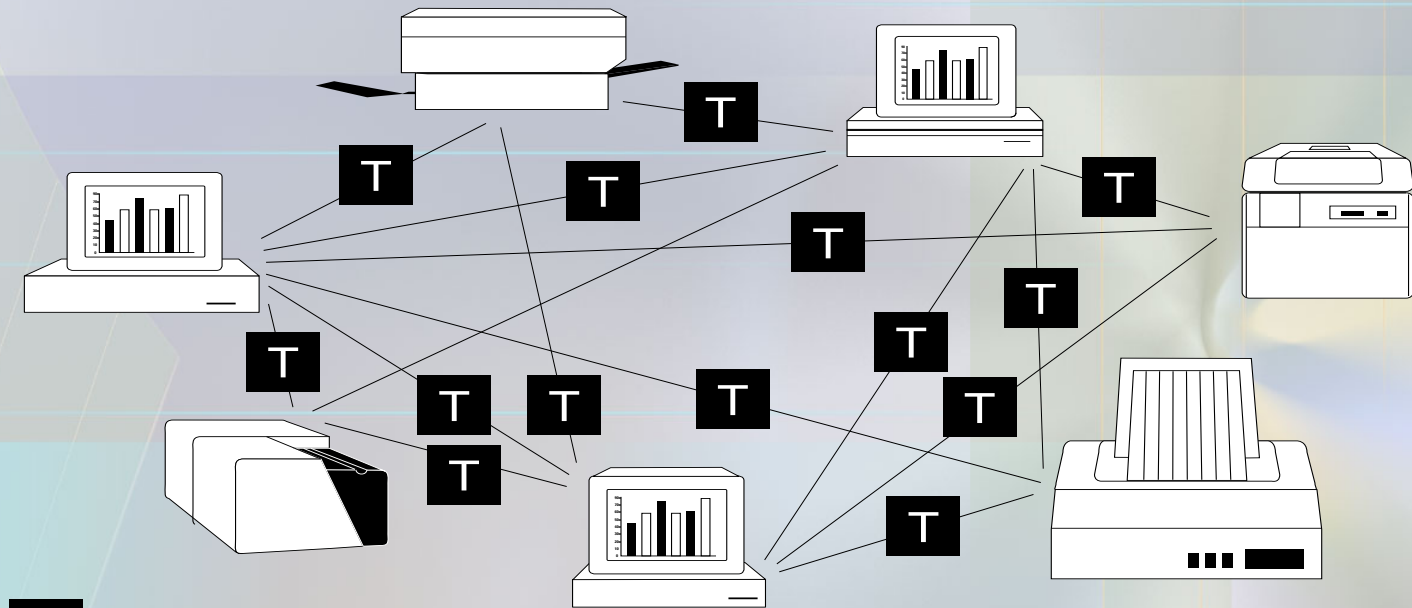
Device-Dependent Approach

- Single transformation from source to destination
- Transformation has both source and destination information - simple, near fool-proof
- Popular in high-end proprietary, and low-end consumer systems - “closed loop” environment
- Cannot archive files that are portable; files are committed to a specific destination device



Device-Dependent Approach

- Difficult to construct and maintain for plug-and-play environments
 - Intended for dedicated device-to-device environments, trades portability for reliability



T Each is a custom pairwise transformation

Device-Independent Approach

- **Option #1 – The Direct Approach**
 - Only accept standard color space(s) as source, such as sRGB, Adobe98, ROMM (Kodak)
 - Everyone converts to the “standard”
 - Example: digital cameras that standardize on sRGB
 - Solves file archive problem

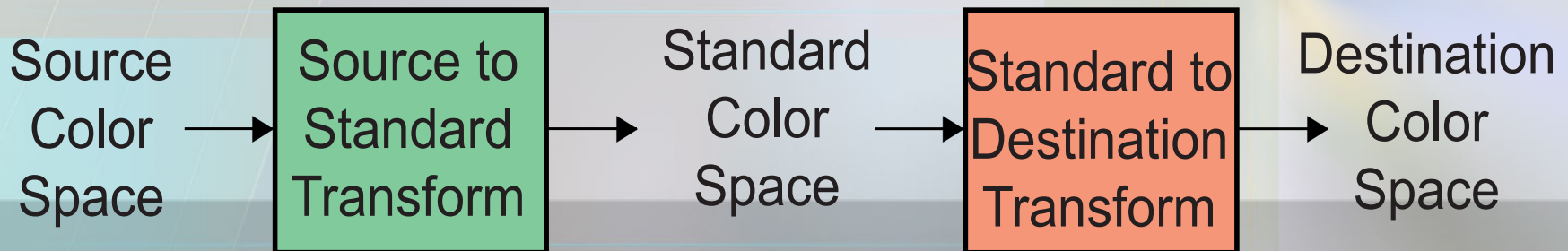


Device-Independent, Option #1

- **Application and device manufacturers must agree to industry-wide standards**
 - **May limit innovation & competitive edge**
- **Many requirements for a “standard” color space; may require multiple standards**
 - **Display efficiency, gamut reproduction**
 - **24 / 48-bit encoding and compression issues**
 - **Luminance-related channel for grayscale**
 - **Conversions always needed to exchange data; performance and quality issues**
 - **“Lowest common denominator” factor**

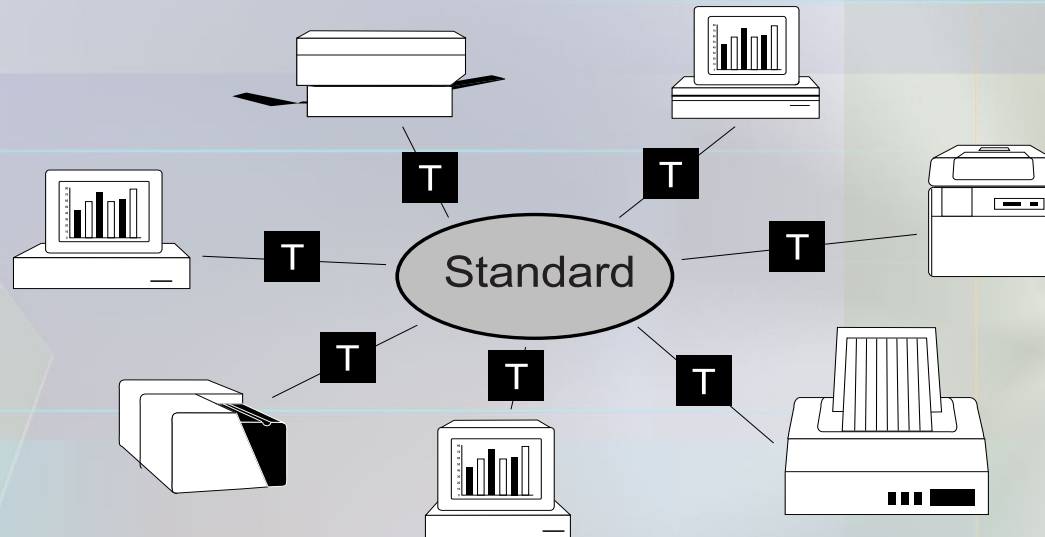
Device-Independent, Option #2

- Break the device-dependent pairwise transformation about a standard color space
 - No longer a direct device-to-device transform, but instead device to “virtual” device space
 - Each half contains data about only one device
 - The device “profile” or model describes the gamut and color-related features of the device, independent of any other device to which it may be linked



Indirect Approach

- Easier to construct and maintain in plug-and-play environments
 - Device manufactures are free to innovate, yet still maintain functionality with existing technologies



T Each is a transform between device and standard color space

Indirect Approach

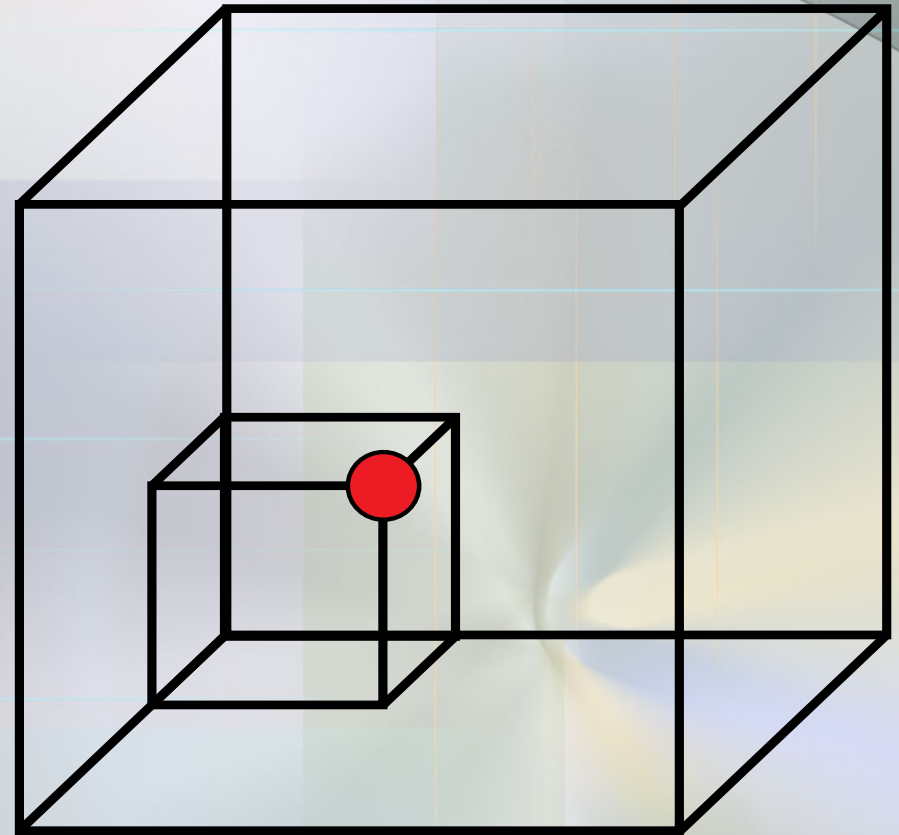
- **A single color transformation is performed from source to destination**
 - Better performance, higher (potential) quality
- **Files archived or data exchanged in source color space (not limited to a “standard”)**
 - Tag(s) describe source color space of file
- **Less effort for applications and drivers**
 - Add tagging feature; do not modify the data
 - However, does make certain assumptions that all applications & drivers interpret the tags

Color Profiles

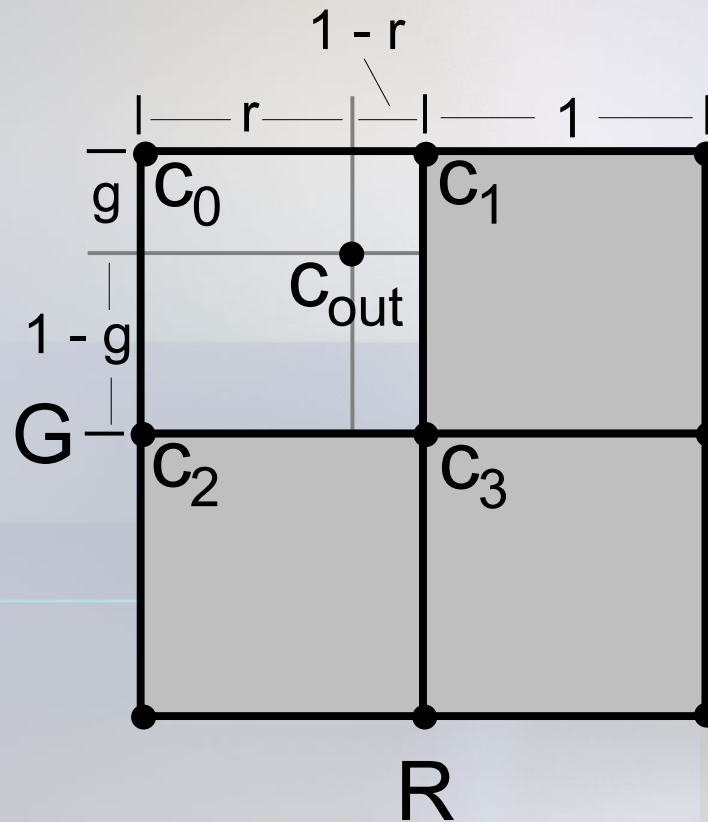
- Encodes the relationship between a source or destination color space and the standard (device independent) color space
 - Requires a specific definition of standard color space including *illuminant, viewing conditions, direct measurements, device color gamut, numeric encodings, lookup tables ... etc.*
 - Model-based & table-based approaches
- Only valid for one instance of the device's state (i.e. media, ink, resolution, etc.)

Trilinear Interpolation

- Use all 8 corners of the enclosing cube to interpolate between points
 - Potential “graininess” in neutrals (achromatic)
 - 7 multiplies per output component
- Can change between 4 and 7 corners as you go from cube to cube
 - Smoothness problems



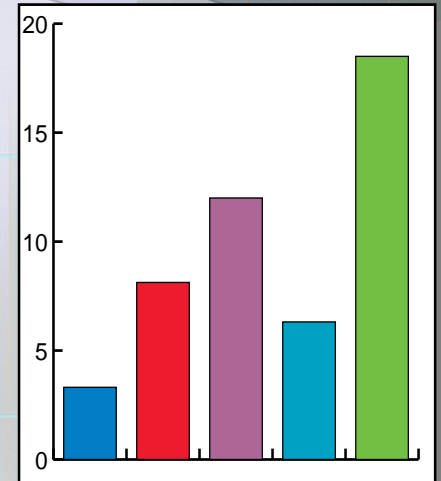
Trilinear Interpolation: 2D Example



$$C_{\text{out}} = (1 - r)(1 - g) C_0 + (1 - r)g C_2 + r(1 - g) C_1 + rg C_3$$

Not True Color Matching

- *Color Management is not truly color matching*
 - We are matching desired appearances viewed in a specific *reference environment*
 - Different environments require changes in the colorimetry
- **Color gamut limitations**
 - Different devices have different ranges of reproducible colors
 - Example.. 100% monitor red not available on CMYK printer
 - Rendering intents

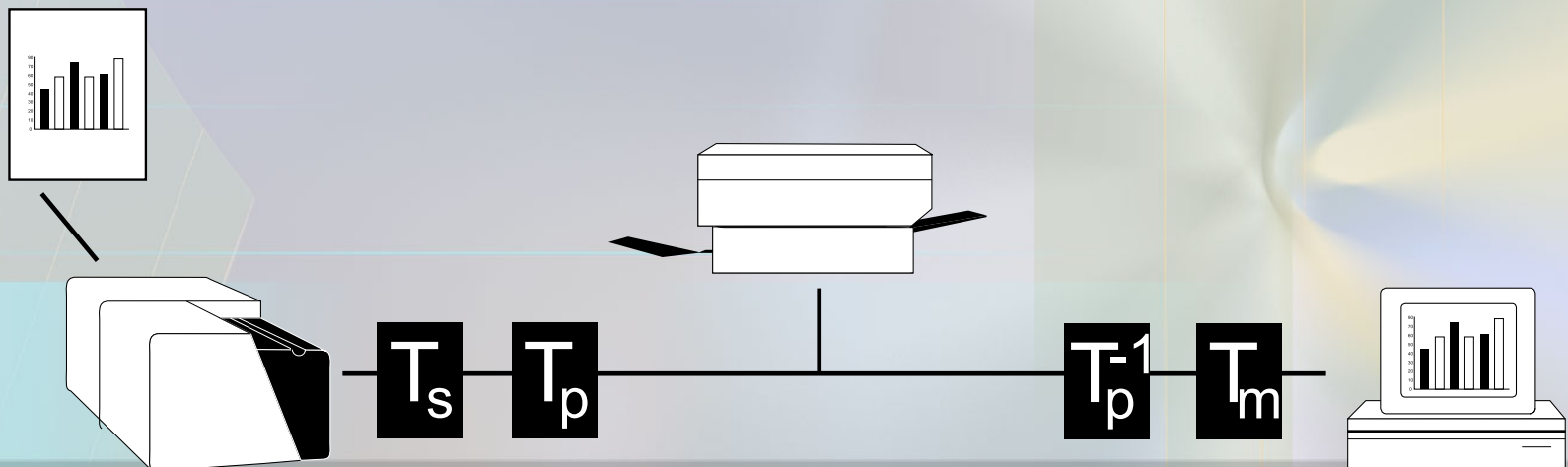


Color Transformations

- Goal is to transform color data from a source to a destination color space
- Color transformations constitute a concatenation of profiles and 3D interpolation + gamut mapping
- Software performance of greater than 1M transformations / second
 - Faster CPUs = more transforms / time
 - Dedicated “soft” logic - FPGA, GPUs

Emulation and Simulation

- Make one printer emulate another printer - typical proofing application
- Simulate on a monitor what a scanned original will look like when printed on a specific printer - *soft proofing*

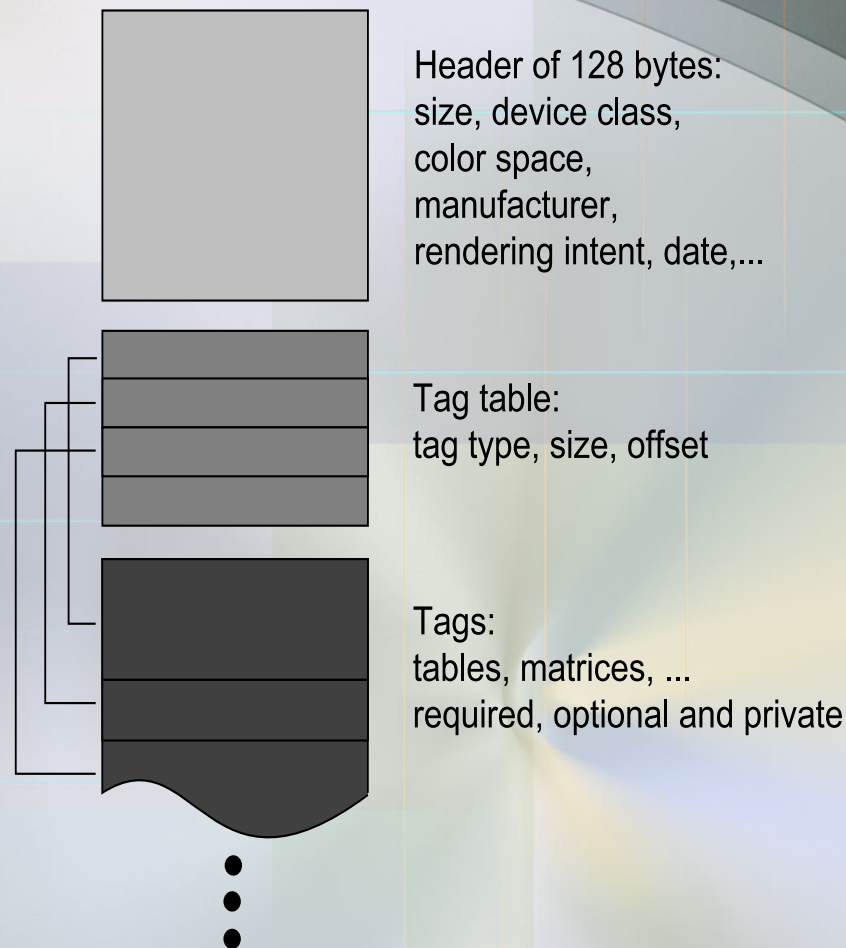


International Color Consortium

- Began as an outgrowth of Apple's original ColorSync 1.0 efforts
- Goal to develop, promote and use a platform-independent profile format
- Original eight members: Adobe, Agfa, Apple, Kodak, Microsoft, SGI, Sun and Taligent. Many more today
- Used by ColorSync, ICM, Solaris, printer drivers, and many applications

ICC Profile Format

- 128 byte header
- Tag-based
- Public required tags
- Public optional tags
- Private tags

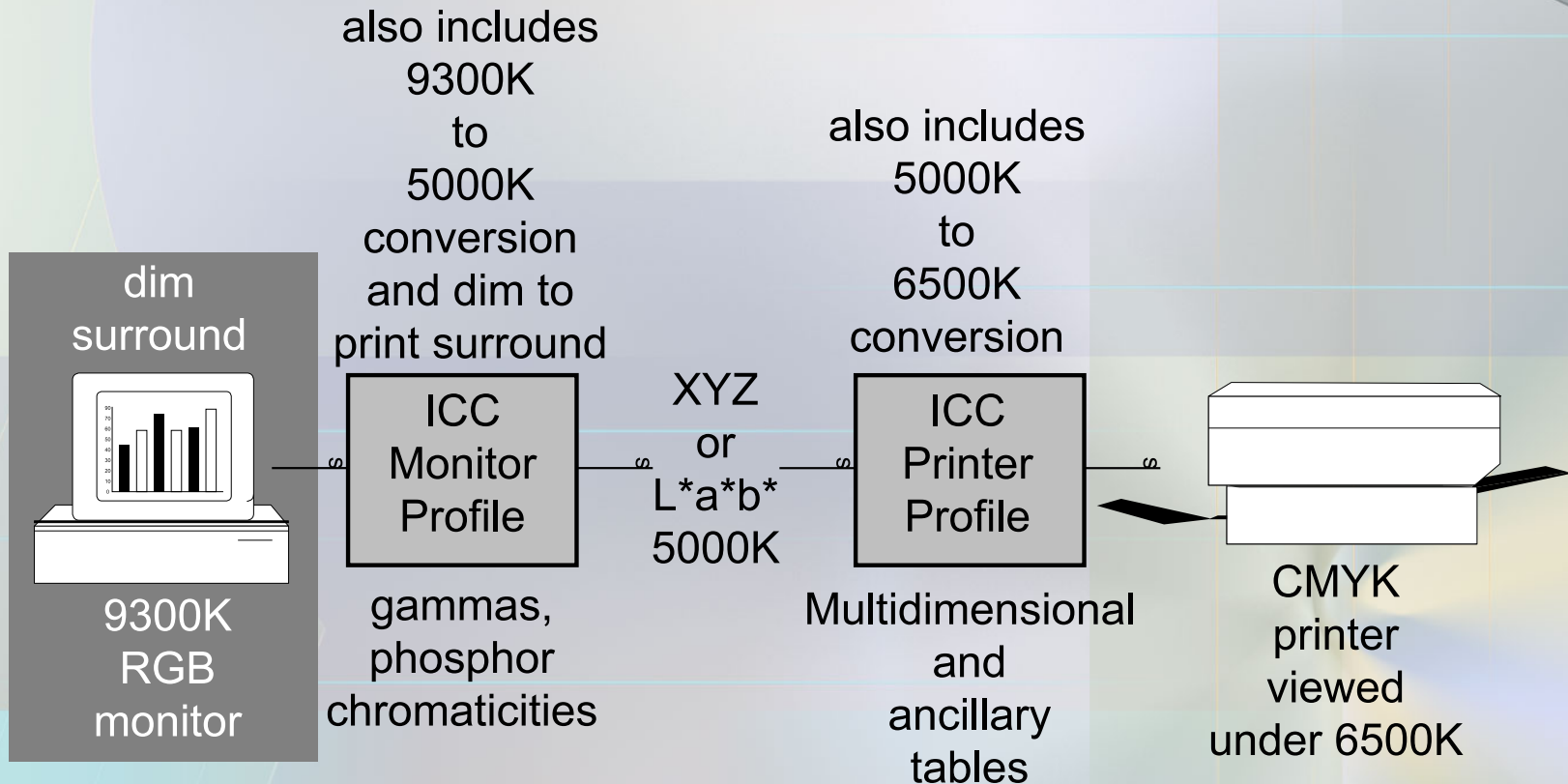


ICC Profile Format (continued)

- **Supports both device-specific and non-device color spaces**
- **Provisions for embedding in EPS, TIFF, JPEG, PICT and PDF**
 - **Related structures in PostScript**
- **Spot color support**
- **Tags for PostScript CSAs and CRDs**
- **Both model-based and table-based profiles supported**

Typical Example

9300K Monitor To a Print Viewed under 6500K...



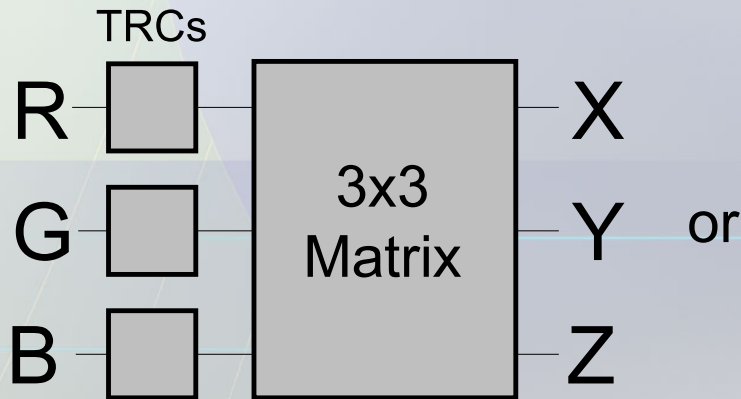
Rendering Intents

- **Absolute Colorimetric**
- **Relative Colorimetric**
 - The *profile connection space* (PCS) for relative colorimetric is defined as:
$$X' = (X_{\text{paper}} / X_{\text{D50}}) X$$
$$Y' = (Y_{\text{paper}} / Y_{\text{D50}}) Y$$
$$Z' = (Z_{\text{paper}} / Z_{\text{D50}}) Z$$
- **Perceptual: photographic reproduction**
- **Saturation: business graphics**

A typical example is a PowerPoint presentation

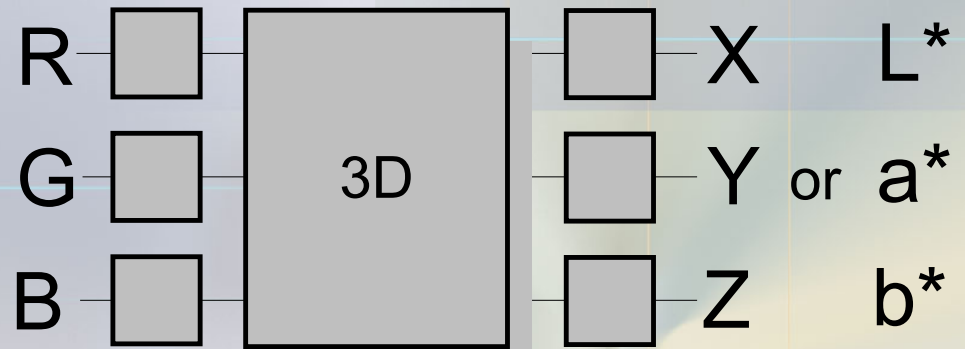
RGB Input and Display Profiles

Monitors



must include
source white
to D50 correction

Scanners,
more complicated
white point corrections
and others

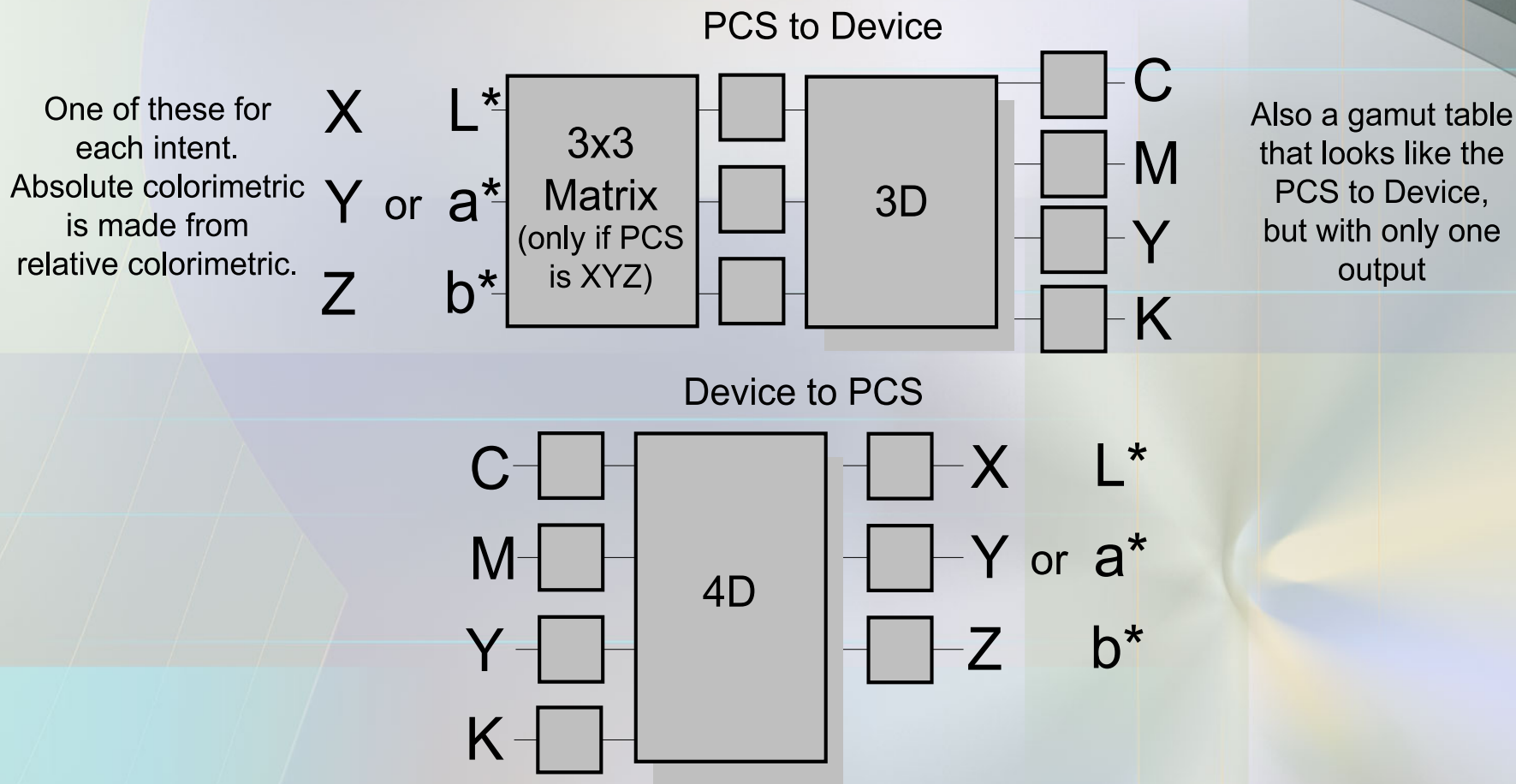


embed more
complicated white
point corrections
in the 3D table

Output Profiles

- **Contain data to go both to and from PCS**
 - The device-to-PCS transform is what permits profile chaining for simulation and emulation
- **RGB and CMYK output**
 - Contains tables for three rendering intents
 - Colorimetric
 - Perceptual
 - Saturation
 - Not all profiles contain all intents; may duplicate
 - Contains gamut tables

RGB and CMYK Output Profiles



sRGB

- **Standard RGB color space promoted by HP and Microsoft in 1996**
 - **Uses CRT model – 3 x 3 matrix with gamma correction (approx. 2.2 overall)**
 - **Uses the ITU-R BT.709-5 primaries, same as those used in studio monitors and HDTV**
 - **Recommended color space for the Internet**

$$\begin{bmatrix} R_{\text{linear}} \\ G_{\text{linear}} \\ B_{\text{linear}} \end{bmatrix} = \begin{bmatrix} 3.2410 & -1.5374 & -0.4986 \\ -0.9692 & 1.8760 & 0.0416 \\ 0.0556 & -0.2040 & 1.0570 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

ColorSync

- **Introduced in 1993, Macintosh-centric**
 - **Standard part of OS-X today**
 - **Robust API (application programming interface) to support & promote third-party developers**
 - **System display profile; profile info extraction, creation and access; searching with custom filters; concatenation; transforms for single colors and images; integrated PostScript support**
 - **Default color engine from Linotype-Hell AG**
 - **Originally LinoColorCMM (color matching module)**
 - **Same CMM was used in Microsoft Windows 2000 and XP, dubbed ICM (Image Color Management)**

Microsoft ICM 1.0

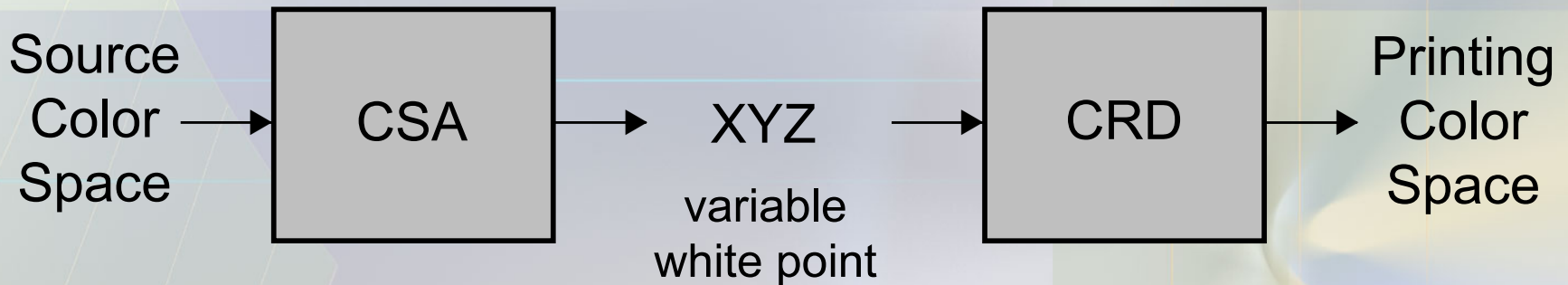
- **Introduced in Windows 95, but not NT 4.0**
- **Not as robust as ColorSync**
 - Limited table sizes
 - Device context and RGB centric
 - Profile registry to shield users from picking profiles directly... good and bad idea
 - Made color a “black box” for average consumer
 - But, limited options & flexibility for professionals
- **Adopted by relatively few applications**
 - Did support alternate CMMs

Microsoft ICM 2.0

- **Introduced in Windows 98 and NT 5.0**
 - Carried forward with Windows 2000 & XP
- **Greater similarity to ColorSync**
 - Supports color spaces other than RGB
 - Conversions outside device contexts
 - Greater flexibility for professional users, and developers of more advanced applications
- **Default CMM from Linotype-Hell AG**
 - Default engine is LinoColorCMM
 - Plug-in support for other CMMs
 - Greater adoption by third-party applications

PostScript Level 2 Color Model

- **Introduced in 1991**
 - **Original PostScript Level 1 was 1984**
 - **Outgrowth of work begun by John Warnock at Evans & Sutherland in 1976, then adopted for use with Xerox Star and InterPress systems, late '70s**



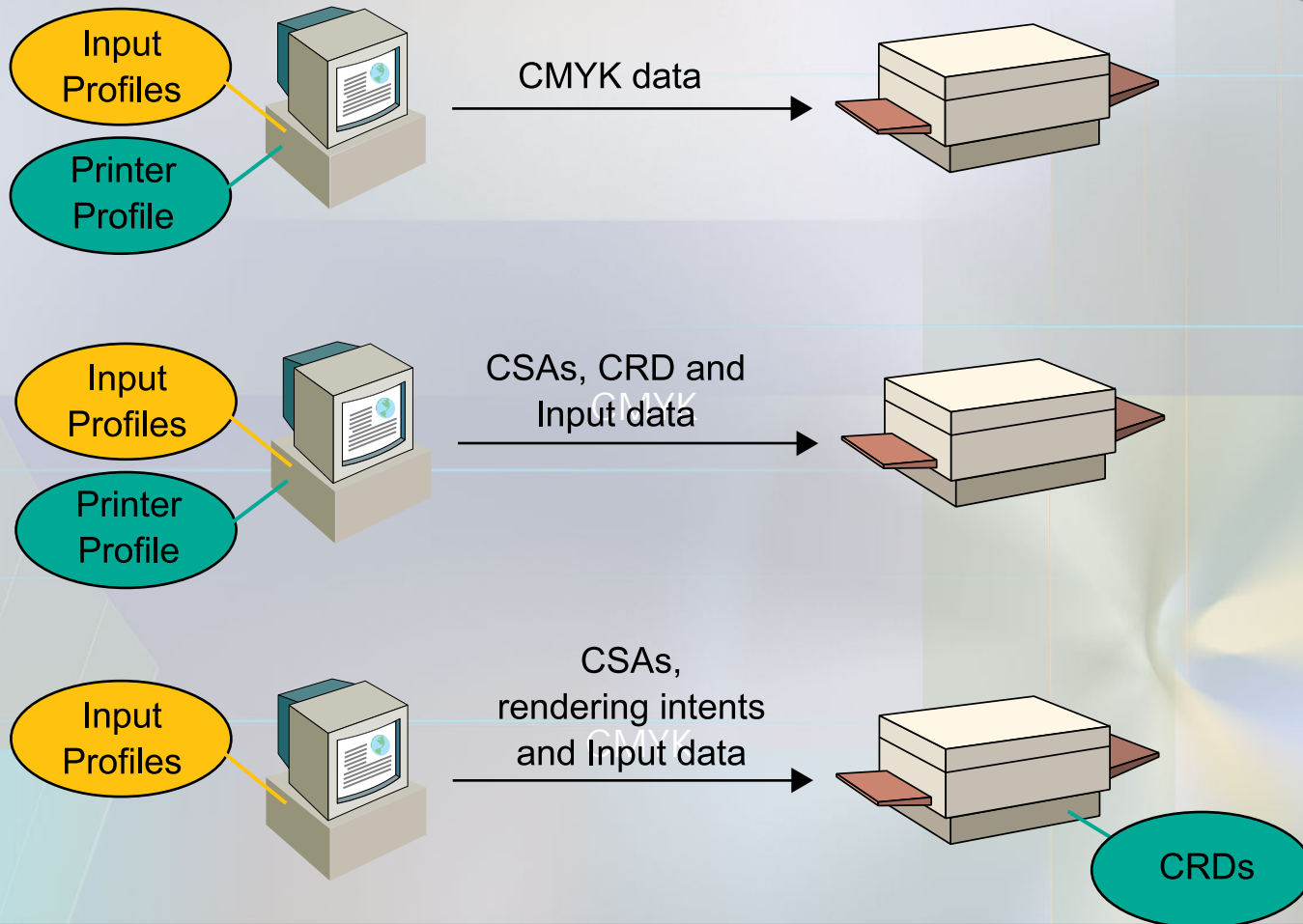
CSA - color space array

CRD - color rendering dictionary

Rendering Intent Support

- **Extension to PostScript Level 2**
- **Takes an intent as an operand and finds a CRD for that intent**
 - **CRD = *Color Rendering Dictionary***
 - 3D look-up table, a “model” of the device
- **Also accounts for halftone and device parameters like media type, resolution**
 - **Permits requesting a rendering intent in a device-independent manner instead of downloading a CRD**

Host vs. Printer-based Color Transformation Options



Tradeoffs of Host vs. Printer

- CPU power and sharing
 - Who has the most memory, the biggest CPU?
- Data representation, i.e. file size and compression = bandwidth restrictions
- Profile storage and updating
 - New profiles to account for device “drift”
- Synchronization with other device parameters (media, ink, print modes)
- Device independence for networks, file archiving, file transfer to remote sites

To Learn More...

- **IS&T Annual Color Imaging Conferences**
 - *Society for Imaging Science & Technology*
 - URL is “www.imaging.org”
 - **IS&T/SPIE Symposium on Electronic Imaging; held annually in San Jose**
 - International society for optics and photonics
 - URL is “www.spie.org”
 - **Journal of Electronic Imaging from IS&T/SPIE**
- **International Color Consortium (ICC)**
 - URL is “www.color.org”
- **Rochester Institute of Technology**
 - **Munsell Color Science Lab**
 - URL is “mcsl.rit.edu”