Digital Print Manufacturing: Color Management Workflows and Roles

Ann McCarthy
Xerox Innovation Group
ICC Steering Committee

International Color Consortium®

What do we mean by workflow?

The term "color fidelity" refers to the successful interoperability of color data, from color object creation to output across multiple targets, such that color reproduction quality consistent with the user's intent can be achieved.

In this context, a workflow is a sequence of color object manipulations …

…that accomplishes a color capture to color production process.

What are the key 'color' questions?

- RGB vs. CMYK ?
  - Scanning
  - Design
  - Exchange
  - Archive
  - Re-use
- RGB vs. CMYK is not where the color workflow question starts…

Digital Color Control (ICC) Architecture Elements 4

- Device calibration *Color Calibration – Printing Aims*
  - Alters the color response of a device to return it to a known state
- Capture and visualization characterization *Response Measurement*
  - Describes the color response of an input or output condition
- Profile creation *Full Range – Color Response Specification*
  - Encodes a characterization and a color aim for use in a transform
- Image color encoding *Color Source Specification – Digital Capture*
  - Unrendered (e.g., capture a scene) vs. color-rendered (targeted)
- Profile selection and exchange *Color Communication – Virtual Film*
  - Profiles can be embedded with an image or document, or can be transmitted as separate files
- Profile use *Color Transformation – Automated Aid to Pressman*
  - Profiles are applied in pairs to transform an image from a current encoding (the source) to another encoding (the destination)
- Visualization – the human element *Printed Job Color Expectation*
  - What does the human expect?

The seven color control architecture elements are threads that interact throughout a color workflow.

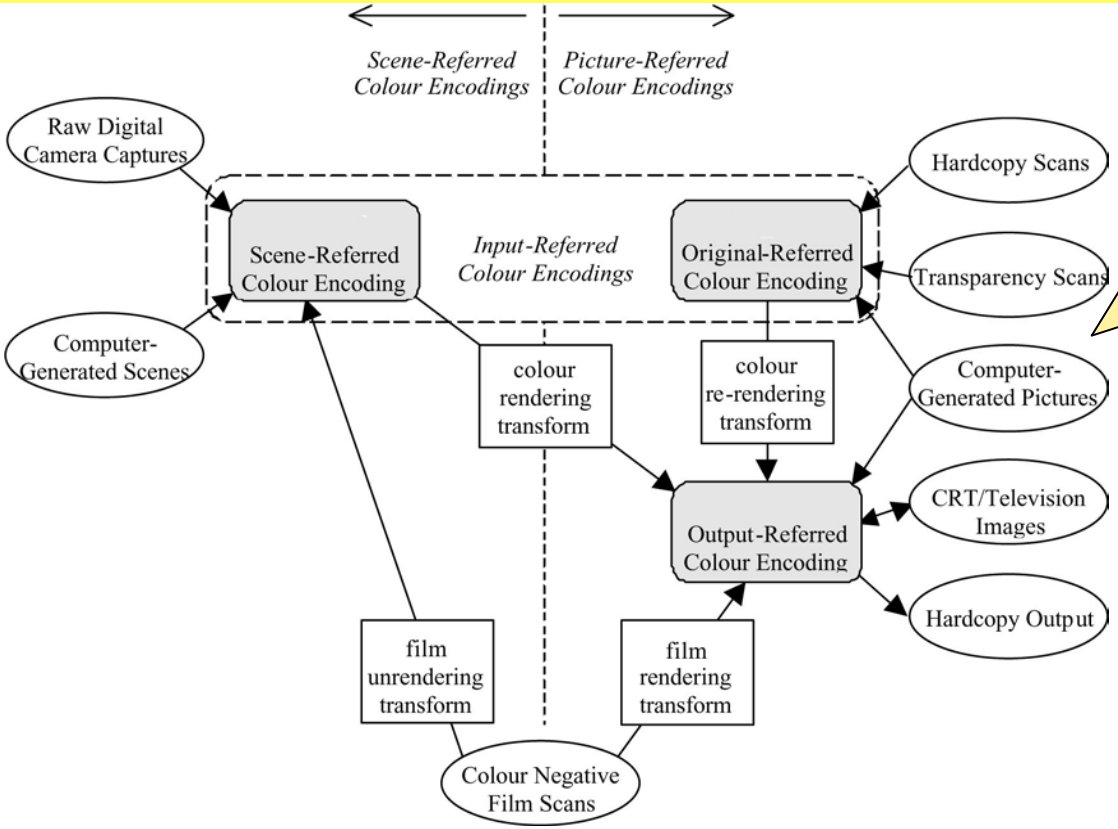
What is the next step in understanding a color workflow…

Analyzing workflow ➔ workflow primitives

- Workflow primitives are the building blocks of every workflow
- These 'normalized' workflow building blocks can be used to understand interactions in real world workflows
- Four dimensions describe the workflow primitives and differentiate them from each other:
  - Fundamental Imaging Operations
    - Correction, preference, gamut mapping, appearance mapping
  - Key Color Fidelity Requirements
    - Process control stability, transformations, metadata persistence
  - Image State Impact
    - Initialization, transition, preservation
  - Image System Interfaces
    - Proprietary/standard methods, proprietary/standard output format

Image State

- The degree to which image data has been constrained or purposed to a particular visualization mechanism, as compared to the degree to which the image data has been retained unaltered with respect to its capture condition



Concepts of scene-referred, original-referred, and output-referred image state as shown in ISO CD 22028-1-Part 1

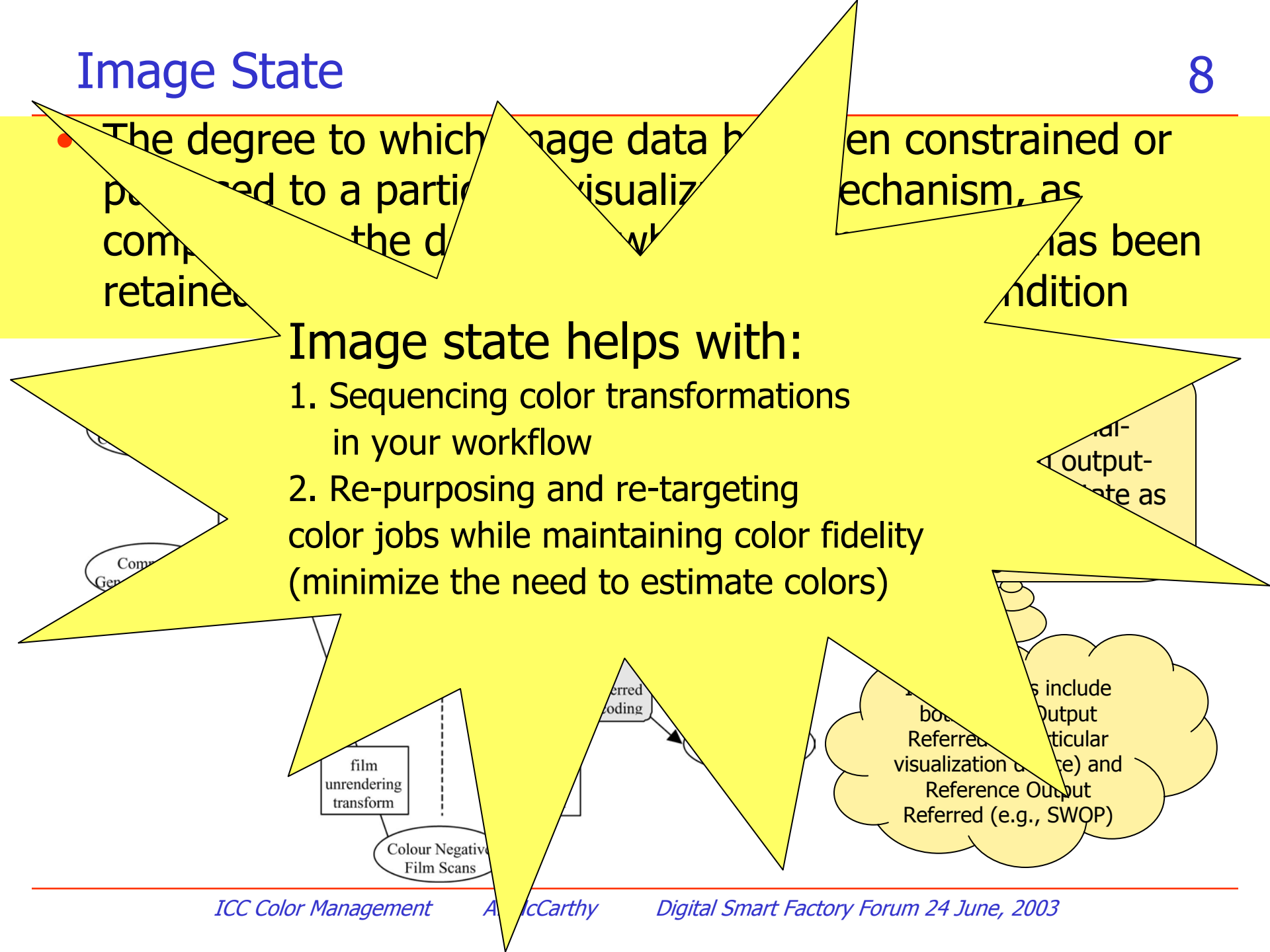
ICC workflows include both Actual Output Referred (a particular visualization device) and Reference Output Referred (e.g., SWOP)

Image State

- The degree to which image data has been constrained or prepared to a particular visualization mechanism, as compared to the data when it has been retained in a scene-referred condition

Image state helps with:

1. Sequencing color transformations in your workflow
2. Re-purposing and re-targeting color jobs while maintaining color fidelity (minimize the need to estimate colors)



Functional Workflow Primitives

- Capture
  - Device correction operations
  - Process control stability
  - Digitized format and data are device dependent and manufacturer controlled
- Color-Render (CMM)
  - Appearance and gamut mapping operations - input to output
  - Apply capture specific characterization transform
  - Apply 'reference' color space encoding transforms
  - Apply visualization condition specific transforms
    - Scum dot elimination
    - Separation⇔gamut interactions

What is that proprietary function that handles digitization?

What is the "primitive" functionality of a CMM?

- Adj… …erations are arbitrary & use specific
- Assemble object/entity …ts …mation
- Visualize
  - Device correction operations
  - Manufacturer controlled
  - Process control stability
  - Includes halftoning, trapping
  - Can include black channel
    - UCR/GCR, black-only
    - Overprint/knockout
- Display the assembled entity

Functional Workflow Primitives

- Capture
  - Devic
  - Pr
  - Dig
    device dep
    m
- Co
  - App
    encoding trans
  - Apply visualiz
    specific transf
    - Scum dot el
    - Separation⇔gamut m ctions

How do we factor in to the workflow the image edits that people want to do to improve their images?

In many workflows, probably all workflows in the commercial world, we need to combine images, text, and vector…
…and we need to keep track of all of the data and metadata in a "job."

What is the proprietary function that handles display?

- Adjust image/object
  - Apply image preferences
  - Preference operations are arbitrary & use specific
- Assemble object/entity
  - Combination of objects
  - Control of job information
- Visualize
  - Device correction operations
  - Manufacturer controlled
  - Process control stability
  - Includes halftoning, trapping
  - Can include black channel
    - UCR/GCR, black-only
    - Overprint/knockout
- Display the assembled entity

What are the roles of printer, prepress, and creator in shaping a workflow…

Printer Responsibilities in a Distributed Color Managed Workflow

- Identify your print shop PRINT CONDITION capabilities
- Establish color calibration – process optimization and process control for each PRINT CONDITION
- Characterize and provide clients with an Actual-Output-Referred ICC profile for each preferred PRINT CONDITION
- Provide clients with a submission guideline defining your capabilities and workflow expectations
- Determine the capabilities of the data exchange methods available to clients and provide recommendations (TIFF, EPS, PDF, PDF/X, fttp to job folder, JDF, etc…)

A **Print Condition** is defined by

1. The Digital Front End (DFE) and print engine combination
2. The opacity, white point, and gloss of the paper stock
3. The screening method
4. The selected calibration Look-Up Table (LUT) in the DFE
5. Any selectable calibration parameters
6. Any selectable color-rendering options in the DFE

Printer Responsibilities in a Distributed Color Manage Workflow

- Identify your print shop PRINT CONDI N capa ilities
- Establis r calibrati roce optimizatio PRINT COND
- Ch Actu preferre
- Provide cli defini expectations
- Determine the bili the exchange met ods ava to cli and provide recommendatio TIFF, EPS PDF, PDF/X, fttp to job folder, DF, etc…)

If a print condition conforms to a Reference Print Condition:

1. Identify the standard characterization via the ICC Characterization Registry
2. Pre-press and creator may use the Reference-Output-Referred ICC profile

A **Print Condition** is defined by

Digital Front (DFE) and print gine combination

pacity, white

the

screening

d

ble (LUT) in the

y selectable ation parameters

6. A selectable color-rendering options in the DFE

Prepress Responsibilities in a Distributed Color Managed Workflow

- Identify client print shop PRINT CONDITION capabilities and submission requirements
- Obtain a Ref/Actual-Output-Referred ICC profile for each preferred PRINT CONDITION
- Calibrate and characterize each proofing device (soft/hard)
- Establish a calibrated re-targeting color workflow – stabilize the proof to production relationship
- Provide creative clients with submission guidelines (agree on workflow expectations)
- Determine the capabilities of the data exchange methods available to clients and provide recommendations (TIFF, EPS, PDF, PDF/X, fttp to job folder, JDF, etc…)

Re-targeting (proofing)

Maintaining image colorimetry while changing device code values to customize to a particular device of the same or larger gamut.

- Need for preserving black in hard proof?
- Acceptable viewing soft proof in RGB?

Re-purposing

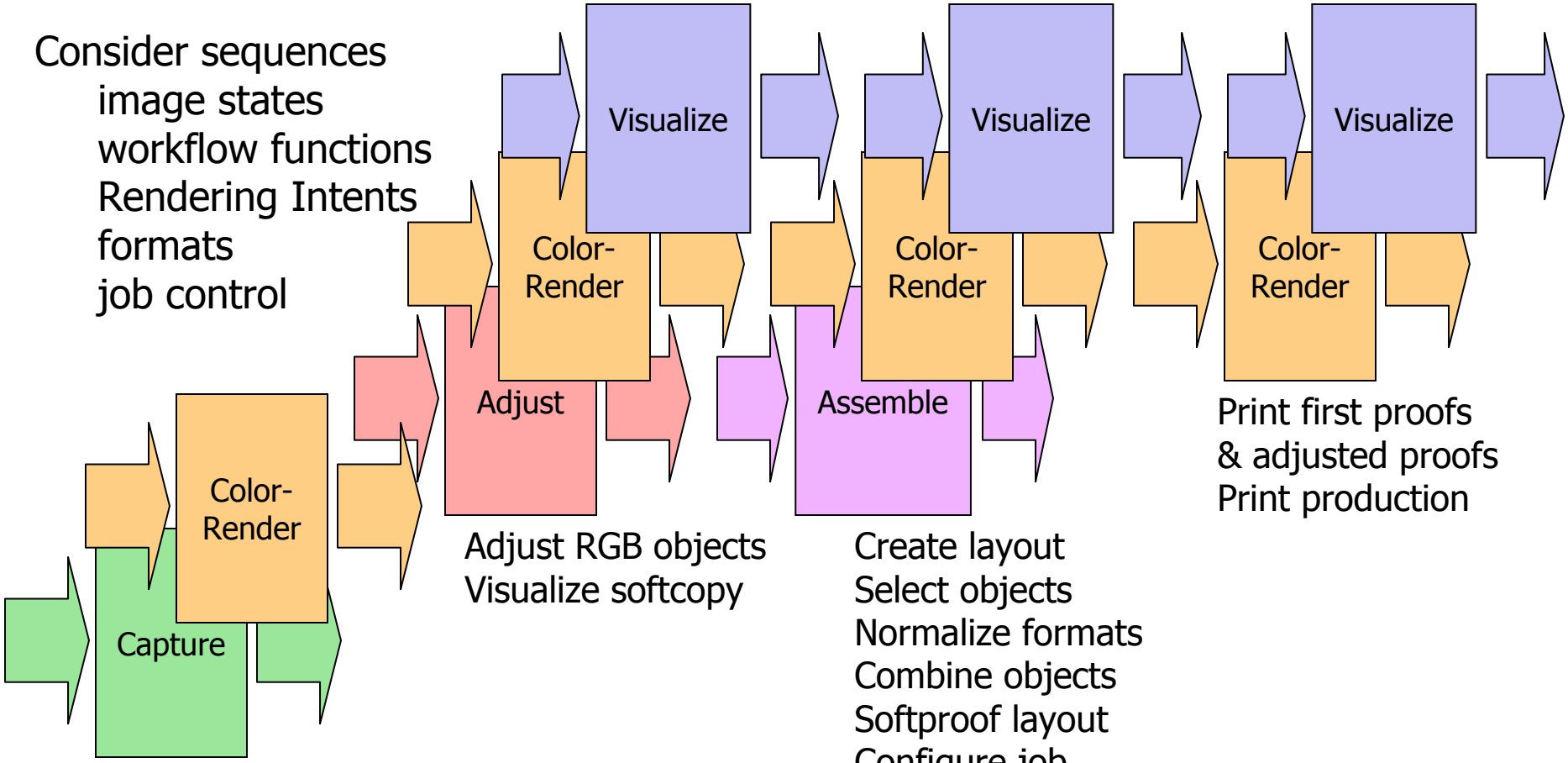
Re-color-rendering data that has been color-rendered for a particular output gamut to optimize for a different output gamut.

Creator Responsibilities in a Distributed Color Managed Workflow

- Determine your color fidelity, content re-use, needs
  - Re-purposing and re-targeting needs?
  - Who will be responsible for preference adjustments?
  - Who will be responsible for color-rendering target colors?
  - The choices of color encoding and workflow configuration should be based on these understandings
    - E.g., will you direct content to both www and press?
    - Or, will you reprint small volume runs?
- Establish calibration and characterization for each design visualization device (soft display / concept proof printer)
- Select a prepress and print shop that supports your re-purpose-ability and re-use needs – obtain the Ref/Actual ICC Profiles
- Determine the capabilities of the data exchange methods and select a method that fits your requirements (TIFF, EPS, PDF, PDF/X, fttp to job folder, JDF, etc…)

Deferred Rendering - Sequential View

Consider sequences
- image states
- workflow functions
- Rendering Intents
- formats
- job control



Capture originals
and
Color-Render to Wide Gamut
Standard RGB

Adjust RGB objects
Visualize softcopy

Create layout
Select objects
Normalize formats
Combine objects
Softproof layout
Configure job

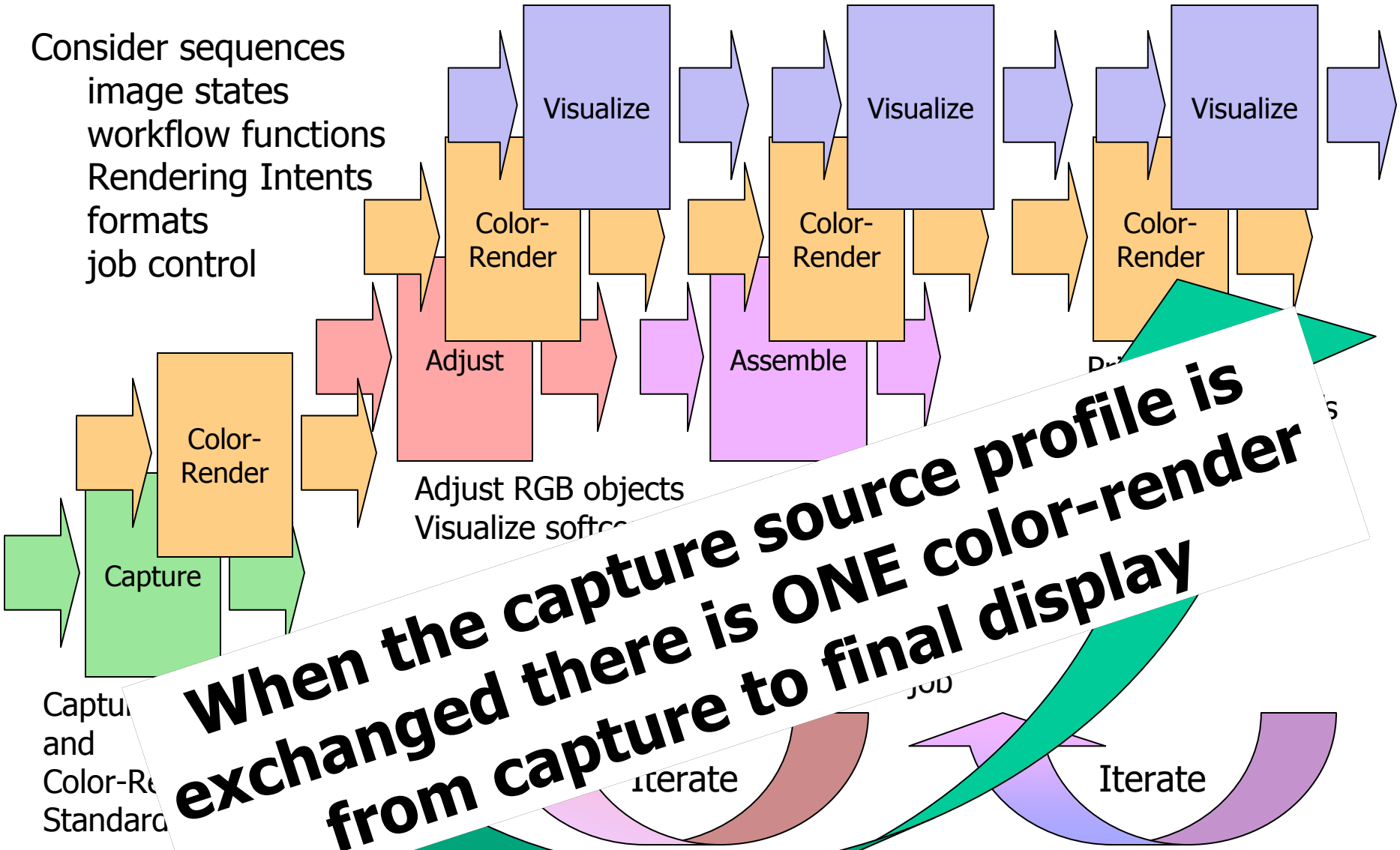
Print first proofs
& adjusted proofs
Print production

Iterate

Iterate

Deferred Rendering - Sequential View

Consider sequences
- image states
- workflow functions
- Rendering Intents
- formats
- job control



When the capture source profile is exchanged there is ONE color-render from capture to final display

JDF 1.2 Color Conversion and Correction

- ICC Color-Render ➔ JDF ColorSpaceConversion process
  - Identify the color management system (per ICC registered name)
  - Identify the use of embedded and/or job collection profiles
  - Rendering Intent – Source AND Destination (per V4 ICC)
  - Reference Printing Condition (simulation)
    - Identify a desired "Reference-Output-Referred" printing condition, e.g., RGB workflow with intent to look like SWOP
  - FinalTargetDevice ➔corresponds to "Actual-Output-Referred" condition
    - Identify a desired (non-standard) printing condition
    - May apply during proofing or during production
    - Printing condition attributes defined to enable correct profile selection
  - Source profile (+optional Abstract profile) per object type
    - Specify the source interpretation for each color object type
- ICC Adjust Image/Object ➔ JDF ColorCorrection process
  - Color adjustments in an Abstract profile allow device-independent changes
  - Color adjustments in a DeviceLink profile allow device specific 'tweaking"
  - Either method can be used to apply changes to all or part of a job

JDF 1.2 Color Conversion and Co ection

- ICC olor-Render → J ColorSpa version process
  - Ide color man t syst CC registe name)
  - Identi f en es
  - Rendering C)
  - Refer
  - Final d" condition
    - Ide
    - Printing cond
  - Source profile
    - Specify the
- ICC Adjust Imag s
  - Color adjustm
  - Color adjustm
  - Either method

ICC Assemble Object/Entity
→ JDF encoding and UI

1. Convey originator intent
2. Convey and apply job-specific metadata

ICC Capture → JDF Scanning

...

ICC Visualization → JDF RIPping
JDF Screening

...

Four Meta-Systems Deliver Color Fidelity

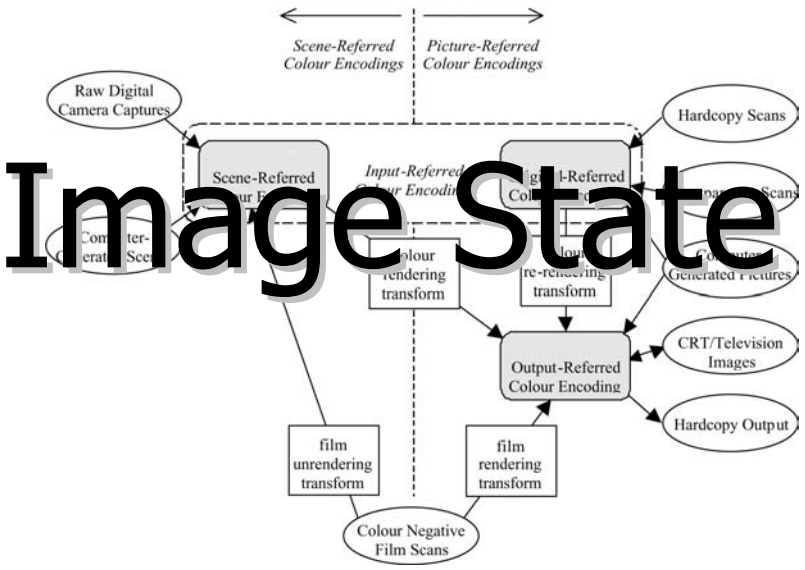


Image State

Workflow Primitives

Color Transformation

Color Calibration

Color Characterization

Color Aim Implementation

Color Communication

Color Aim Expectation

Color Control Architecture

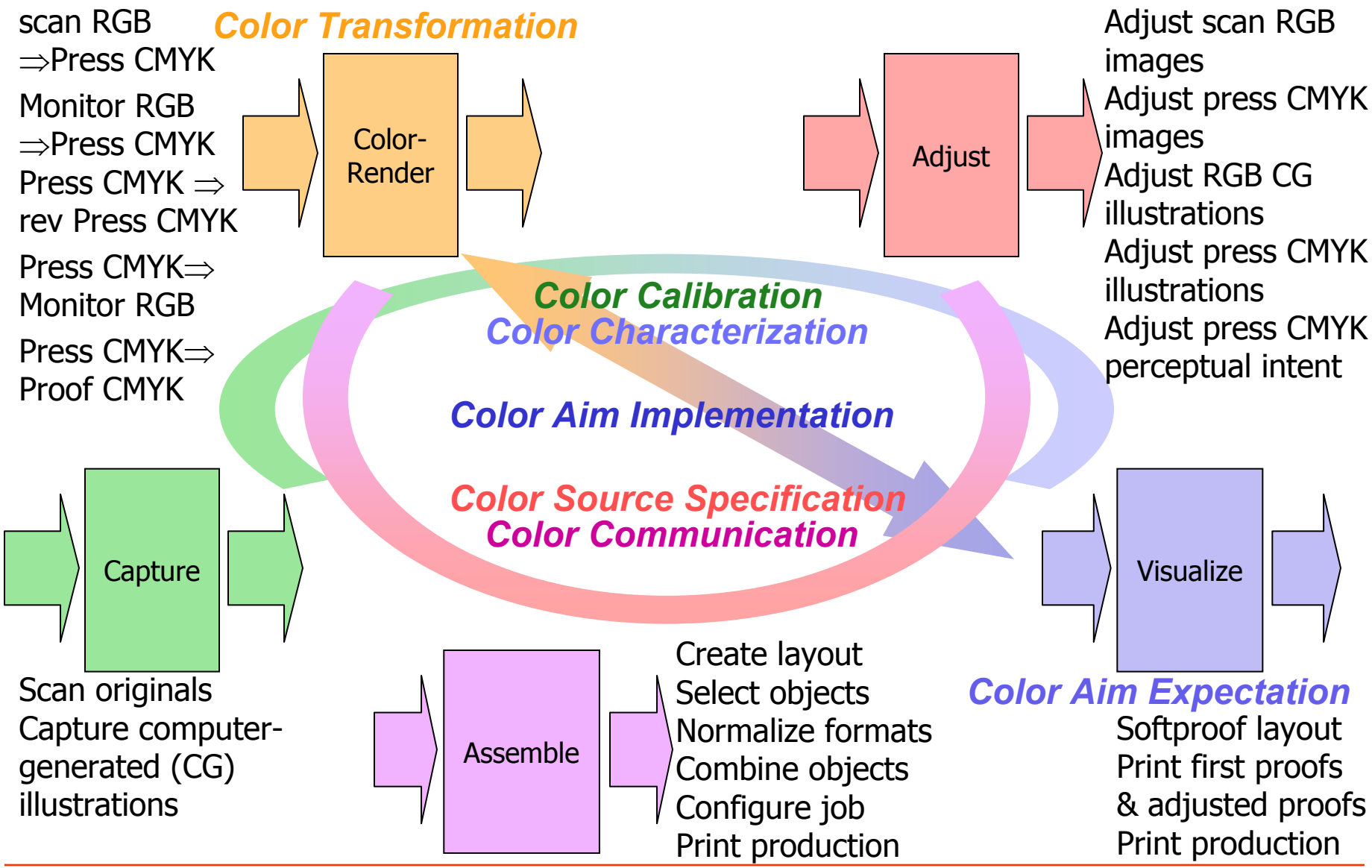

Surplus may fund retreat center

UPD enforces

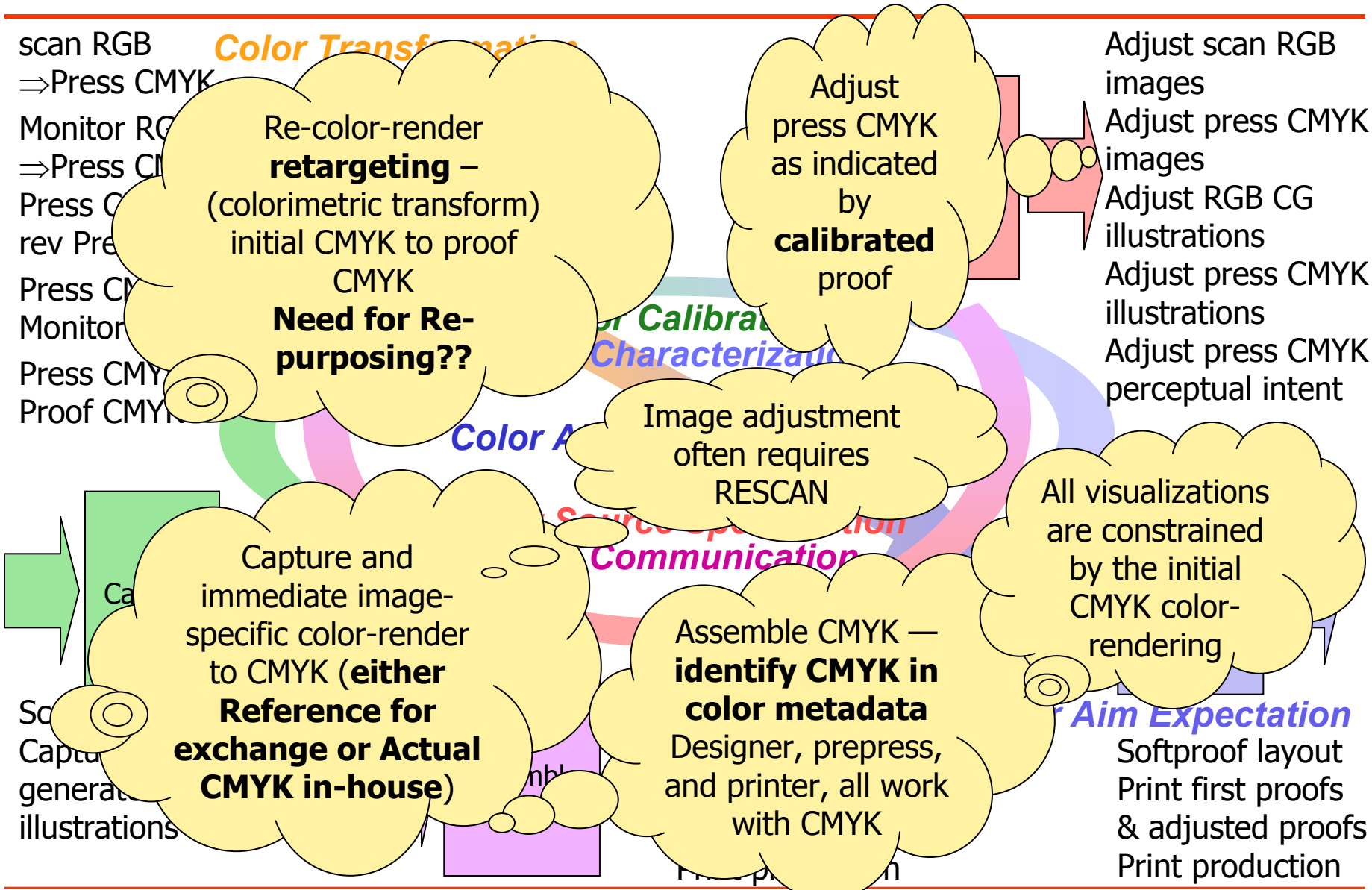
Formats & Protocols

workflow examples and color management scenarios

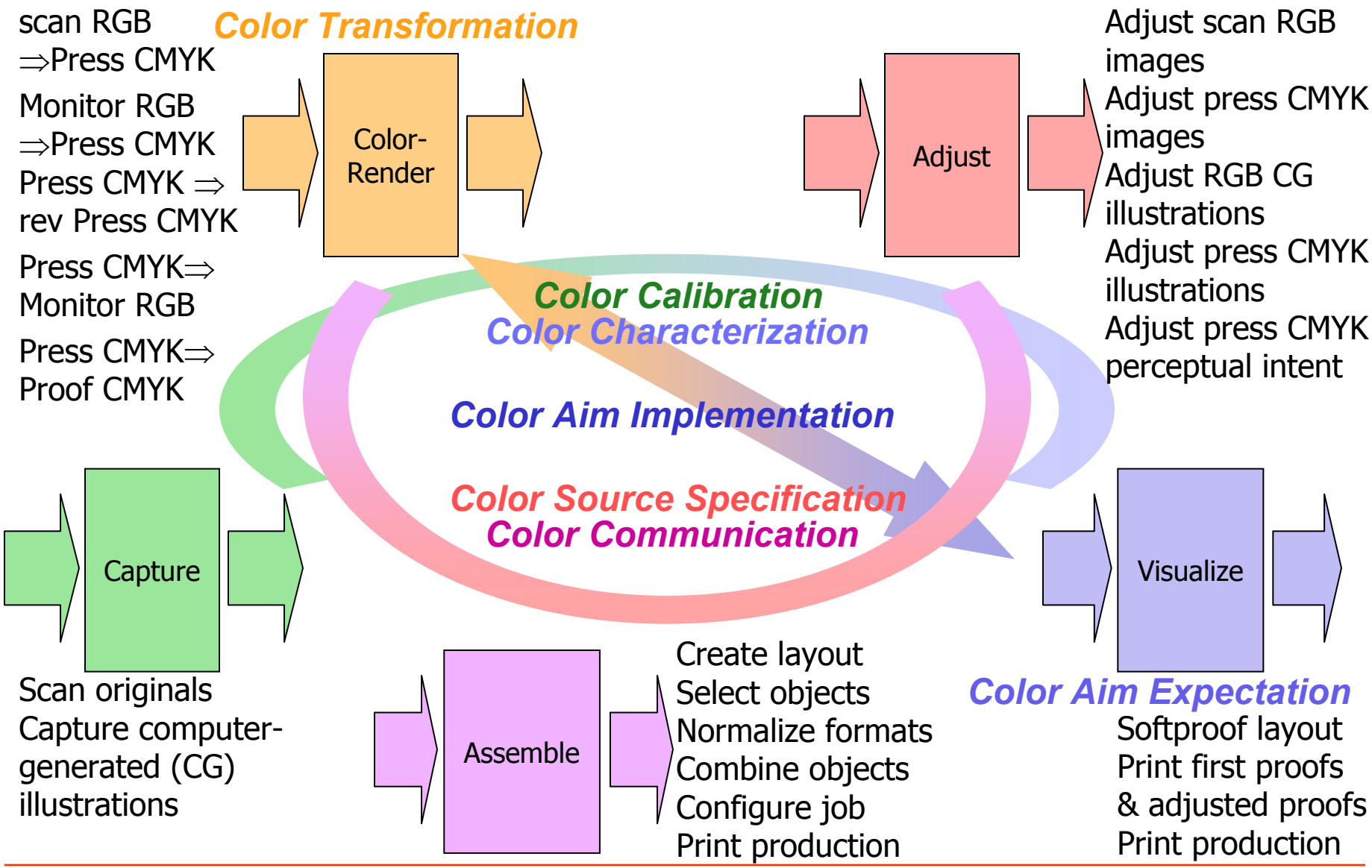
A Commercial Print Workflow – Design CMYK



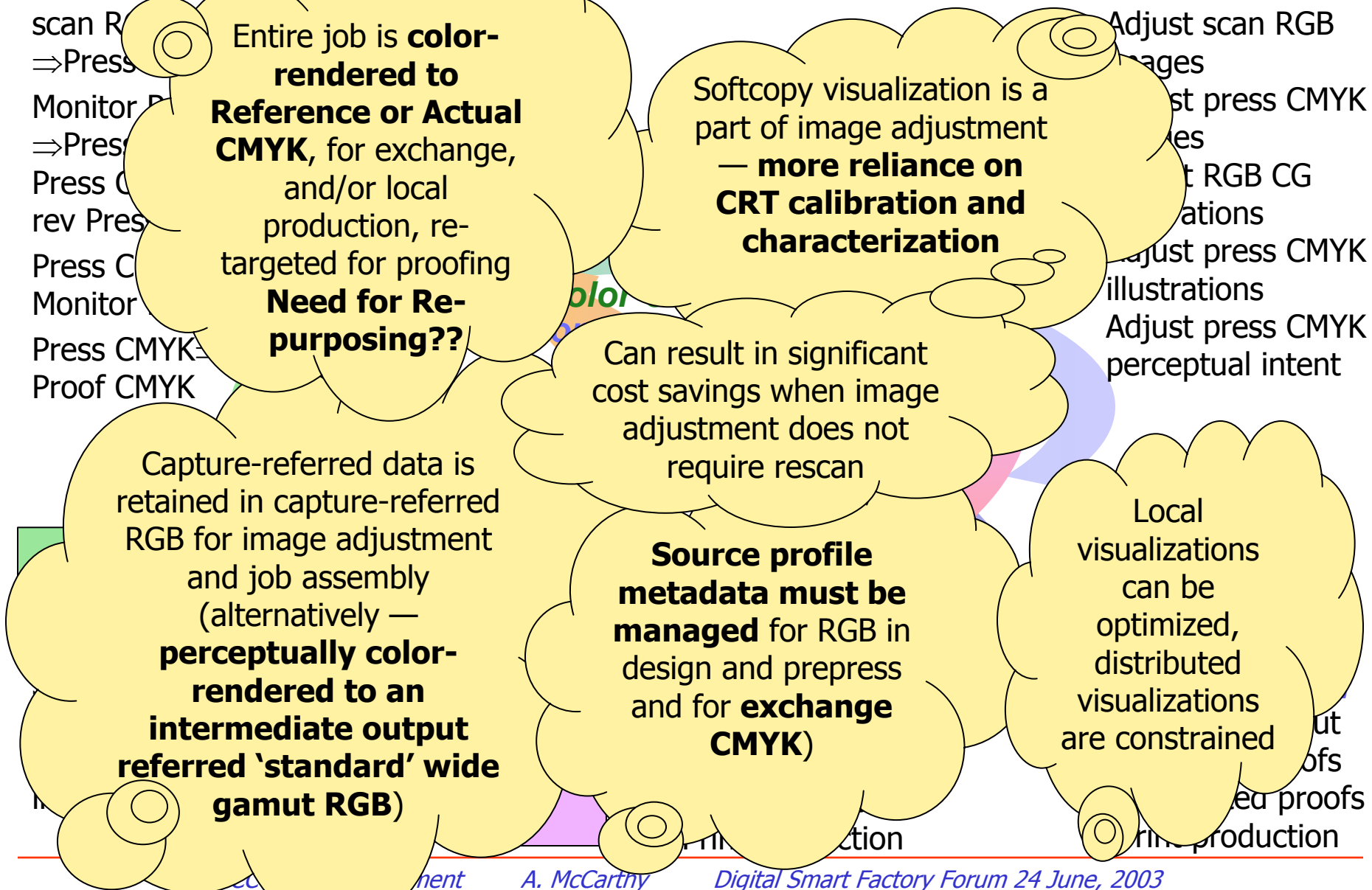
A Commercial Print Workflow – Design CMYK



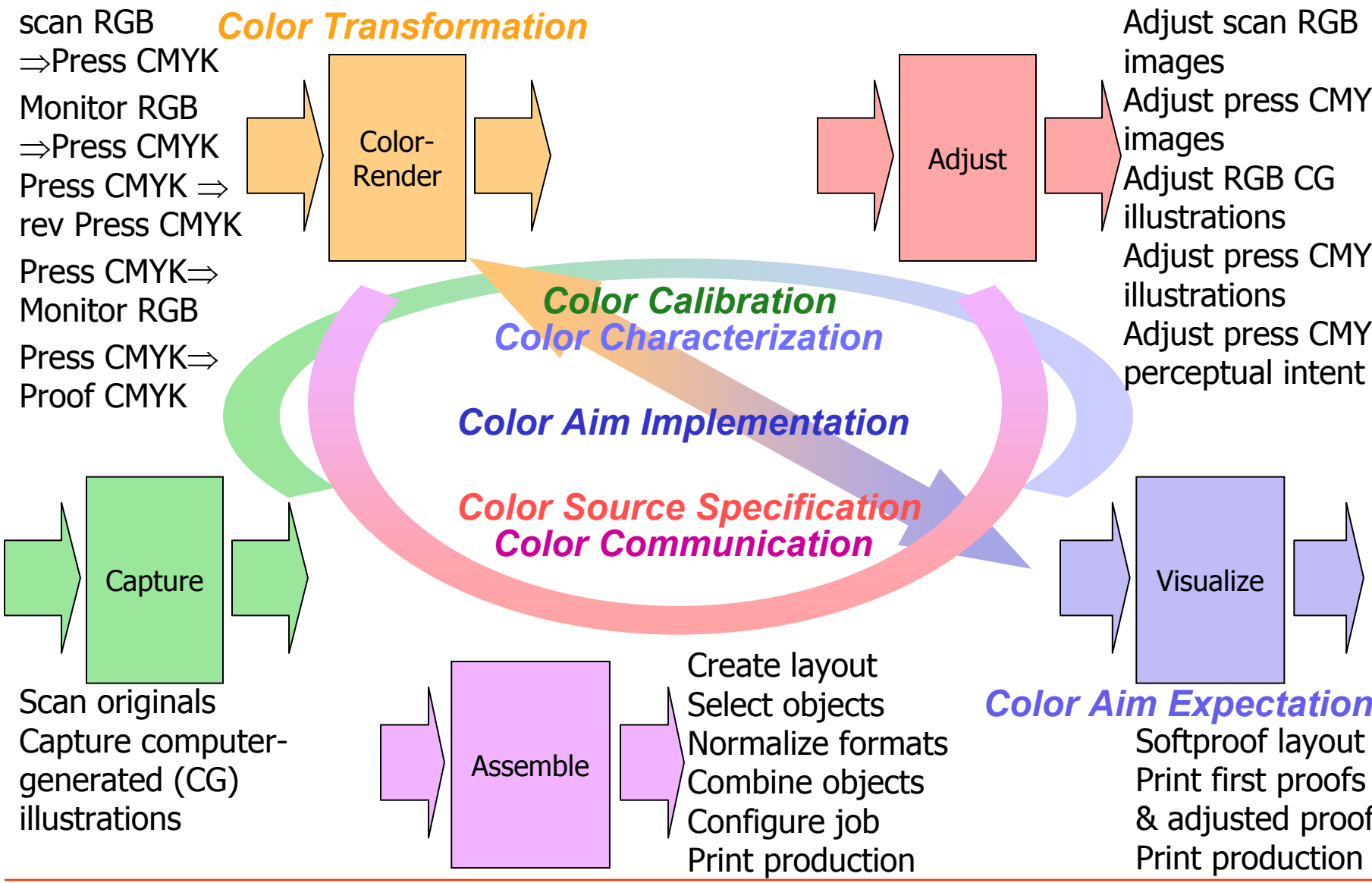
A Commercial Print Workflow – Design RGB



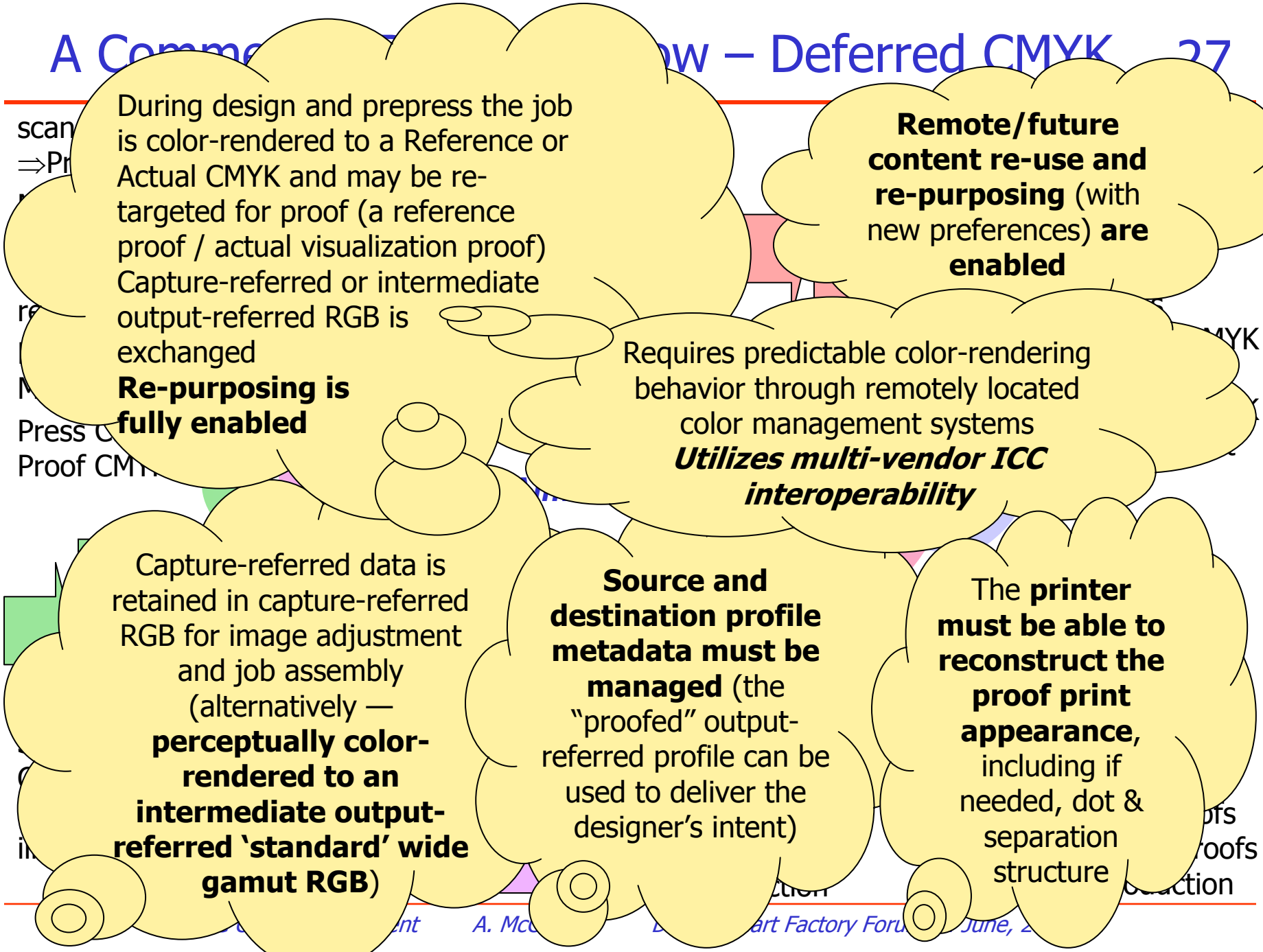
A Commercial Print Workflow – Design RGB



A Commercial Print Workflow – Deferred CMYK

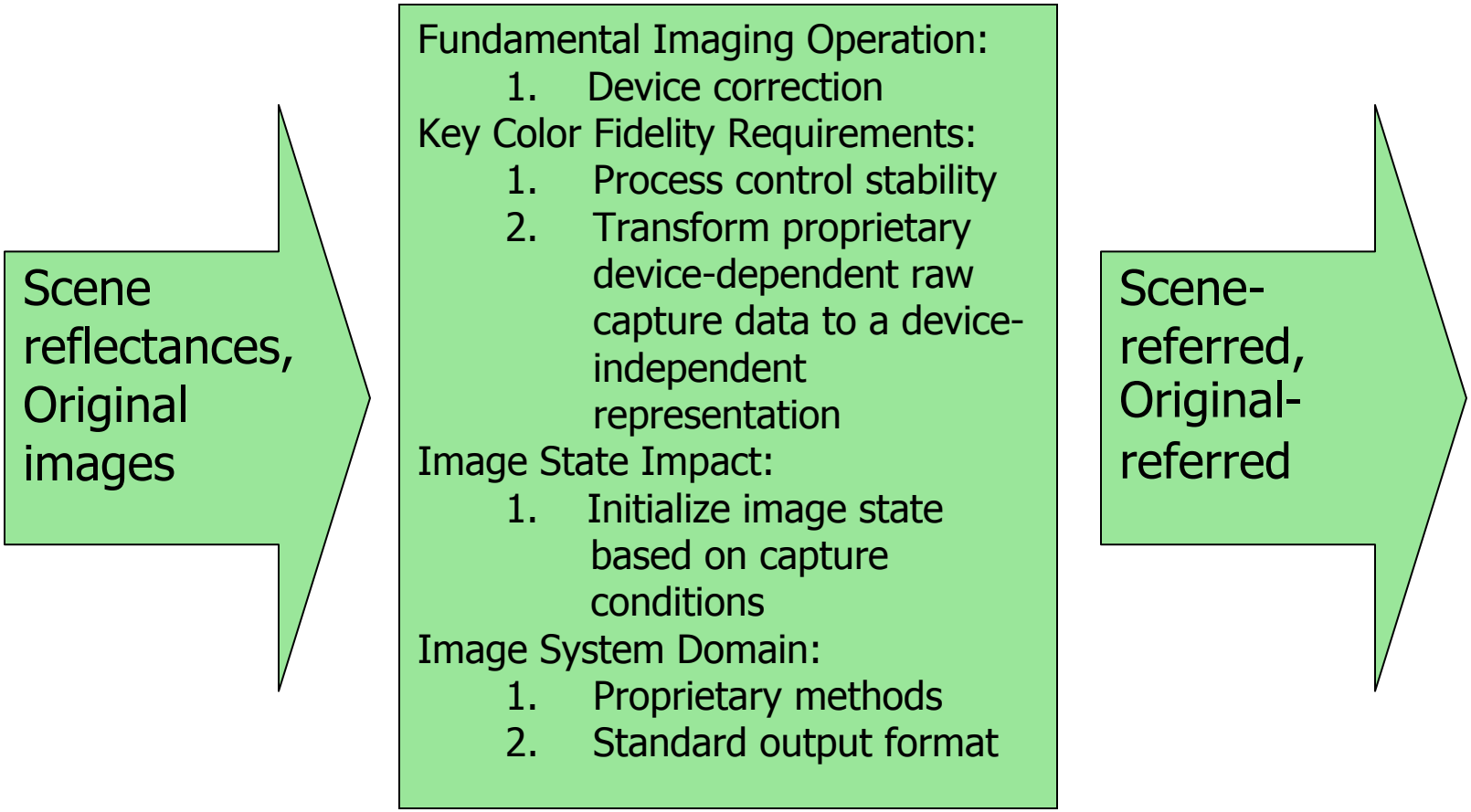


A Comme… …w – Deferred CMYK



Detail on Workflow Primitives…

The Capture Workflow Primitive



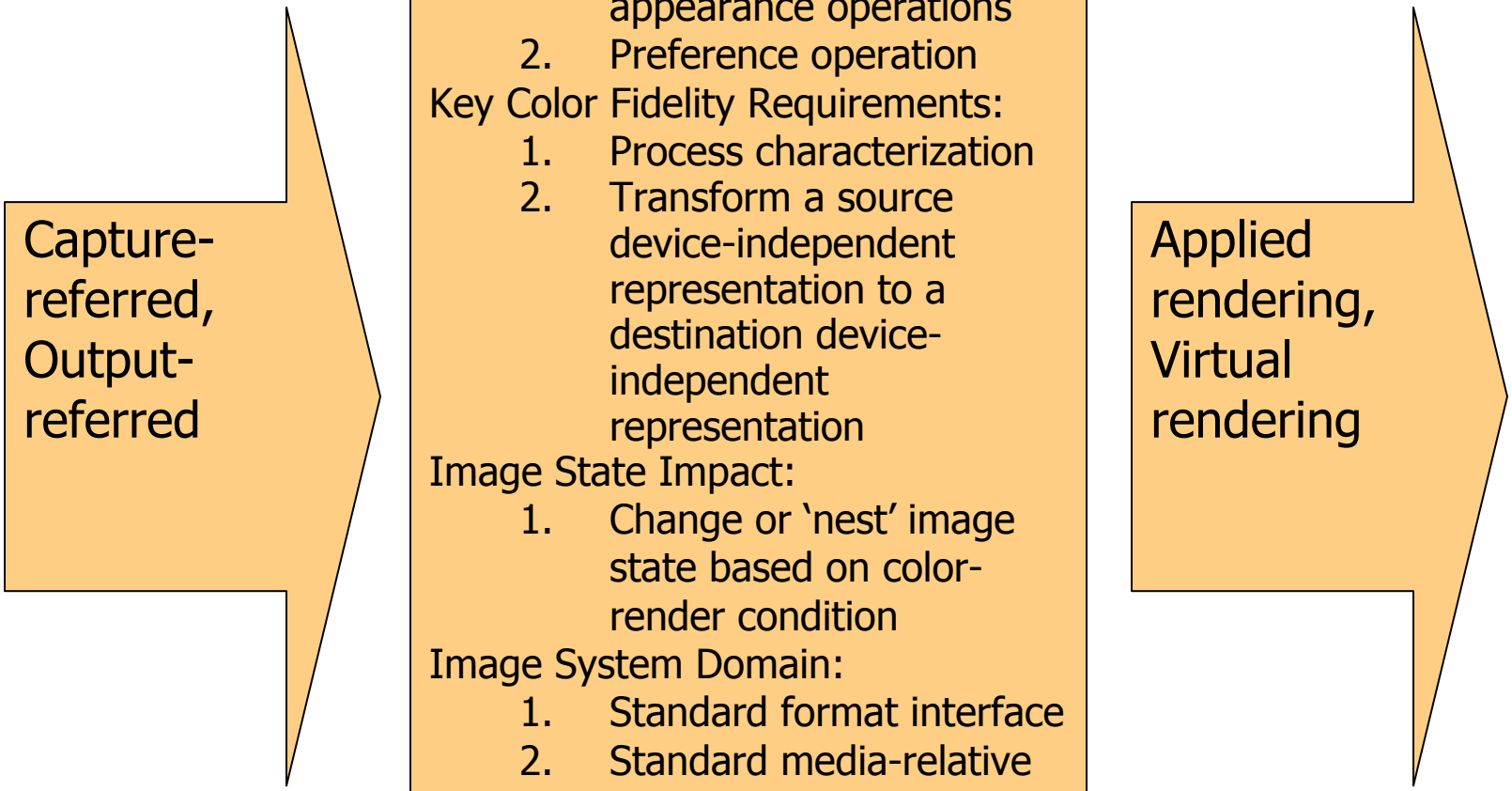
The Capture Workflow Primitive

- Capture Scene
  - Image state is initialized to scene-referred
  - Capture scene-referred images from a digital camera or from a scan of a negative
  - Process to correct for capture device, e.g., gray balance and intensity adjustment
  - Data is not color-rendered to any visualization, i.e., an ICC profile is associated but not applied
  - Scene capture refers to capture of the spectral radiances of a view of the natural world and may also include capture of a computer simulation of such spectral radiances
  - The Capture Scene primitive produces scene-referred images - each with an associated input profile - characterizing the capture condition (negative scanner, digital camera)

The Capture Workflow Primitive

- Capture Original
  - Image state is initialized to original-referred
  - Capture reflection or transparency color-rendered images, scan from reflection hardcopy (e.g., artwork), or scan from positive transparency; or capture color-rendered computer generated image
  - Process to correct for capture device, e.g., gray balance and intensity
  - The original-referred image data contains color-rendering to the output gamut of the process that created the original hardcopy
  - This case includes color-rendered computer graphics images: image data created through computer graphics application that is rendered to an image look and is original-referred to the creation station
  - Computer generated color-rendered images may be rendered to the gamut of the creation display, or rendered to a print softcopy
  - The Capture Original primitive produces original-referred images - each with an associated input profile - characterizing the capture condition (e.g., reflection scanner, positive transparency scanner, computer graphics workstation monitor) - the profiles are not applied

The Color-Render (CMM) Workflow Primitive



The Color-Render (CMM) Workflow Primitive

- Color-Render CHANGES IMAGE STATE
- Color-Render includes profile based gamut mapping and manual adjustments to correct gamut mapping
- Color-Render includes gamut to gamut transform, e.g., CMYK to CMYK, sRGB to monitor RGB transform, within the same meta-image-state
- Color-Render includes appearance (viewing environment) adaptation and preferential aspects of the perceptual intent
- Color-Render may occur ONE or N TIMES
  - Following Capture
  - Before Adjust Image/Object
  - Following Adjust Image/Object
  - As a substep of Assemble Object/Entity
  - After Assemble Object/Entity
  - As a substep of Visualization
  - Using Visualization (manual color-rendering)

The Color-Render (CMM) Workflow Primitive

- Gamut mapping and appearance mapping are carried out in combination between a source profile, a destination profile, and a color management module (CMM)
- Both gamut mapping and appearance adaptation are built into the perceptual transforms of source and destination profiles
- Chromatic adaptation is built into media-relative colorimetric intent transforms
- A CMM handles clipping to the destination gamut boundary as required for the media-relative colorimetric intent
- A CMM should handle certain gamut mapping cases
  - E.g., for perceptual, media-relative colorimetric, and saturation intents, white should NEVER be interpolated
    - 255r,255g,255b should always convert to non-image/non-ink values (255,255,255 for RGB and 0,0,0,0 for CMYK)

The Color-Render (CMM) Workflow Primitive

- 'Virtual color-rendering' can be used to assign a visualization for later use

  ⇒ Producing images with an associated (but not applied) Visualization Condition (ICC Profile), e.g., "Virtual CMYK"

  Examples
  - SR Data + Input Profile ⇒ SR Data + Input Profile + Output Profile(s)**
  - OR Data + Input Profile ⇒ OR Data + Input Profile + Output Profile(s)**
  - ROR Data + Source Profile ⇒ ROR Data + Source Profile + Output Profile(s)**
  - AOR Data + Source Profile ⇒ AOR Data + Source Profile + Output Profile(s)**

    **optional additional Output Profile to handle proofing, image setter, etc. The 2nd or additional output profile would be chained following the input and primary output profiles when the image data is processed for the proofer, imagesetter, etc.

- 'Applied color-rendering' can be used to process image data through a source profile and one or more destination profiles

  ⇒ Producing images realized in a Visualization Condition (output-referred state)

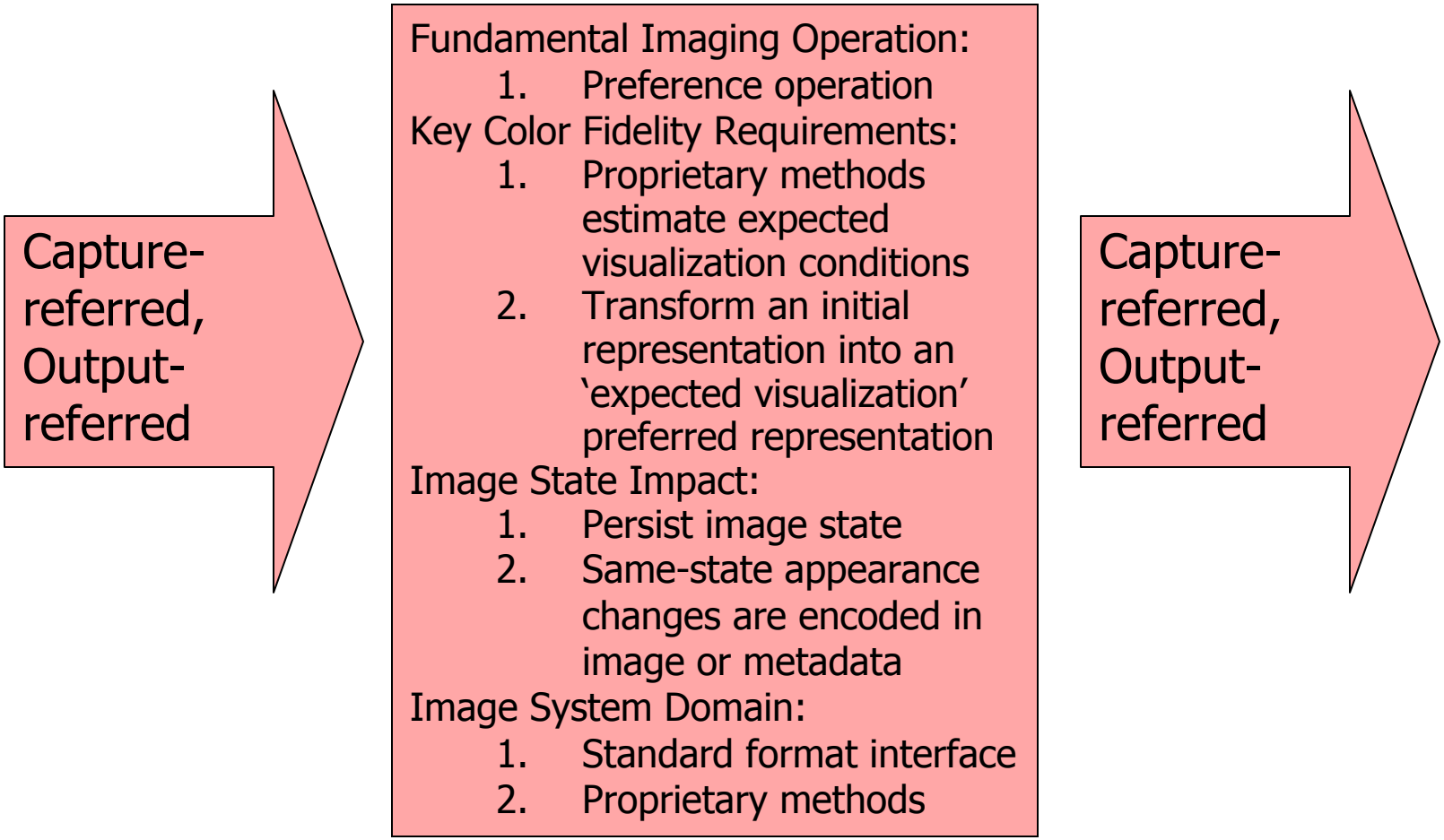
  Examples
  - SR Data + Input Profile ⇒ AOR or ROR Data + Source Profile
  - OR Data + Input Profile ⇒ ROR or AOR Data + Source Profile
  - ROR or AOR Data + Source Profile ⇒ a different ROR or AOR Data + Source Profile

The Color-Render (CMM) Workflow Primitive

- 'Virtual color-rendering' can be used to ...sign a visualization for ... use
  - ⇒ Produ... ...ges with an ... (bu... ...ed) Visual... ... Condition (ICC Pro... ...Virtu...
  - Examples
    - SR Data + In...
    - ...
    - AOR... ...s)**
      - **op...
      - The ... ...the
- 'Applied color-r... ...e ...proc... ...age data through a sou... profil... ...one ... ...e destina... ...n profiles
  - ⇒ Producing images realized ... ...ualization ...ition (output-refe...red state)
  - Examples
    - SR Data + Input Profile ⇒ AOR... ...ROR Data + Source Profile
    - OR Data + Input Profile ⇒ ROR... ...AOR Data + Source Profile
    - ROR or AOR Data + Source Profile ⇒ a different ROR or AOR Data + Source Profile

Note that in a Version 4 Input profile – the media-relative colorimetric rendering intent portrays a capture-referred image into the Colorimetric PCS – in this case a Color-Render transform is applied (chromatic adaptation, white point mapping)- but the image does not become "output-referred"
...so in this case managing the display image state is up to the destination profile.

The Adjust Image/Object Workflow Primitive



The Adjust Image/Object Workflow Primitive

- Adjust deals with preference and aesthetic changes, not changes required due to constraints imposed by any particular visualization mechanism
  - Adjust does NOT change image state
  - Adjust DOES change encoded values
  - Adjust does not change image or assembled object file formats
- Adjust includes local and/or global preference operations
- Adjust may occur ONE or N TIMES
  - Following Capture
  - Before Color-Render
  - Following Color-Render
  - As a substep of Assemble Object/Entity
  - As a substep of Visualization
  - Using Visualization
- Global preference changes may be accomplished by adjusting a profile

The Adjust Image/Object Workflow Primitive

- Directly Preference adjust image data
  - Any associated profile(s) are carried along but not changed
  - This method will support local changes as well as global changes in the image data
  - Optionally, adjust image data while viewing through the associated input/source profile, or the concatenated input/source and destination profiles
  - Adjust image data while viewing through the available profile(s) to get the desired real or virtual output appearance and to ensure that the preference adjustments are compatible with profile-based color-renderings

The Adjust Image/Object Workflow Primitive

- Preference adjust the perceptual intent of the input/source profile associated with an image
  - While viewing the (capture-referred or reference-output-referred) image through the input/source profile and, if a destination profile is associated, while also viewing it through the destination profile
  - The edits are applied to the perceptual intent transform of the input/source profile and the viewing is through the perceptual intent transform(s) of the associated profile(s)
  - This method allows only global changes using the current ICC paradigm
- This path requires an image container or file format that can support embedding or associating at least one profile with an image, so that the input/source profile containing the edits can be linked with the image
- If an output profile is also associated, then the image container must allow two linked profiles

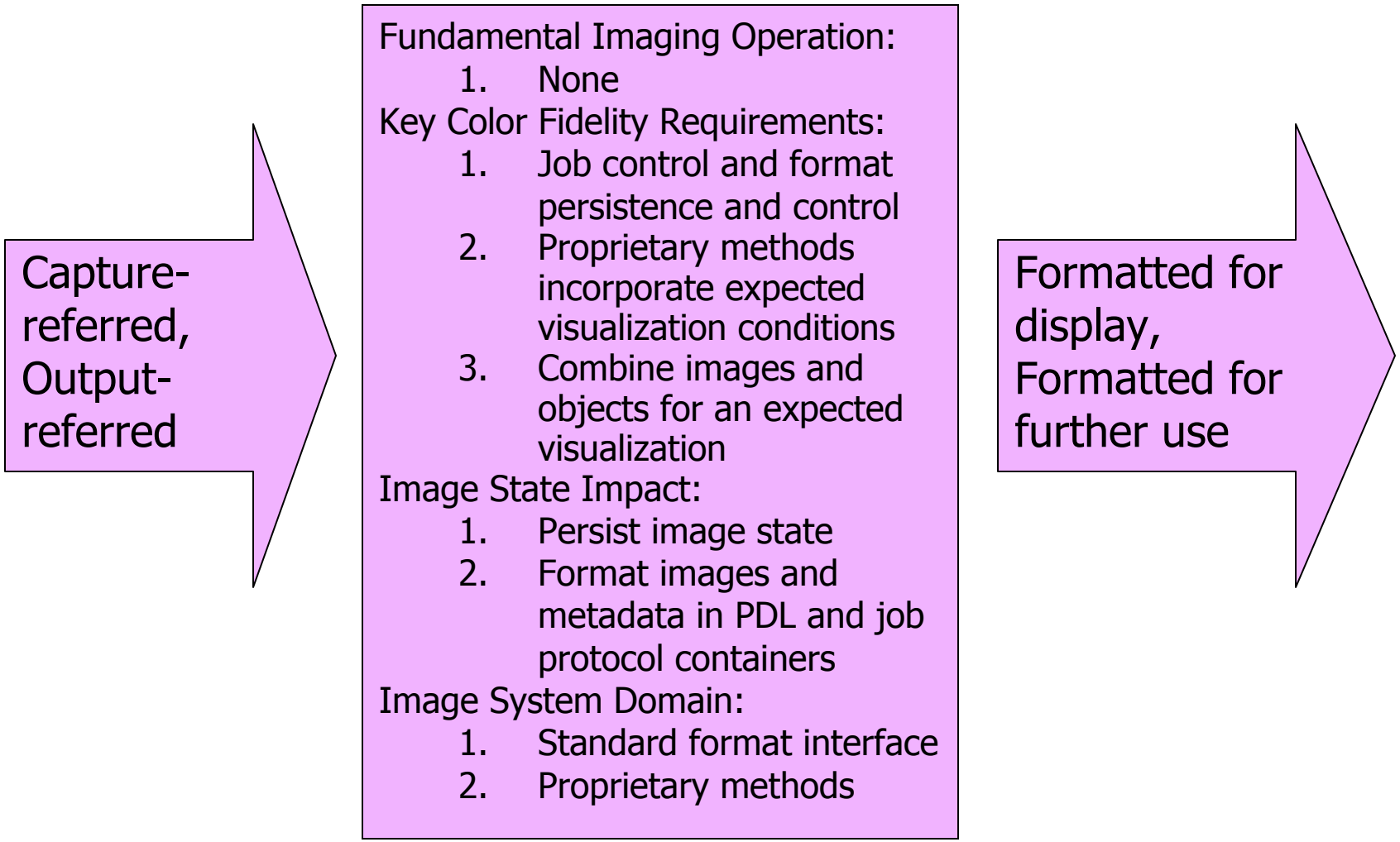
The Adjust Image/Object Workflow Primitive

- Preference adjust an abstract profile associated with an image
  - While viewing the (capture-referred or reference-output-referred) image through the input (or) source profile and, if a destination profile is associated, while also viewing it through the destination profile
  - One advantage in using an abstract profile to capture edits, compared to using either a source/input or destination profile, is that then image dependence can be constrained to the abstract profile, rather than being merged with the device-centric source/input or destination profiles
  - This method allows only global changes using the current ICC paradigm
  - **This path requires an image container or file format that can support embedding or associating more than one profile with an image, so that at least 2 profiles (the input/source profile and the abstract profile) can be linked with the image**
  - **If an output profile is also associated, then the image container must allow three linked profiles**

The Adjust Image/Object Workflow Primitive

- Preference adjust the perceptual intent transform of the output profile associated with an image (before it is applied)
  - While viewing the (capture-referred or reference-output-referred) image through the input/source profile and output profile
  - The edits are applied to the output profile perceptual intent transform and the viewing is through the perceptual intent transform(s) of the associated profiles
  - This method allows only global changes using the current ICC paradigm
- This path requires an image container or file format that can support embedding or associating at least two profiles with an image, so that the input/source profile, and the output profile with the edits, can be linked with the image
- This method can be used to accomplish visualization-specific preference adjustments

The Assemble Object/Entity Workflow Primitive



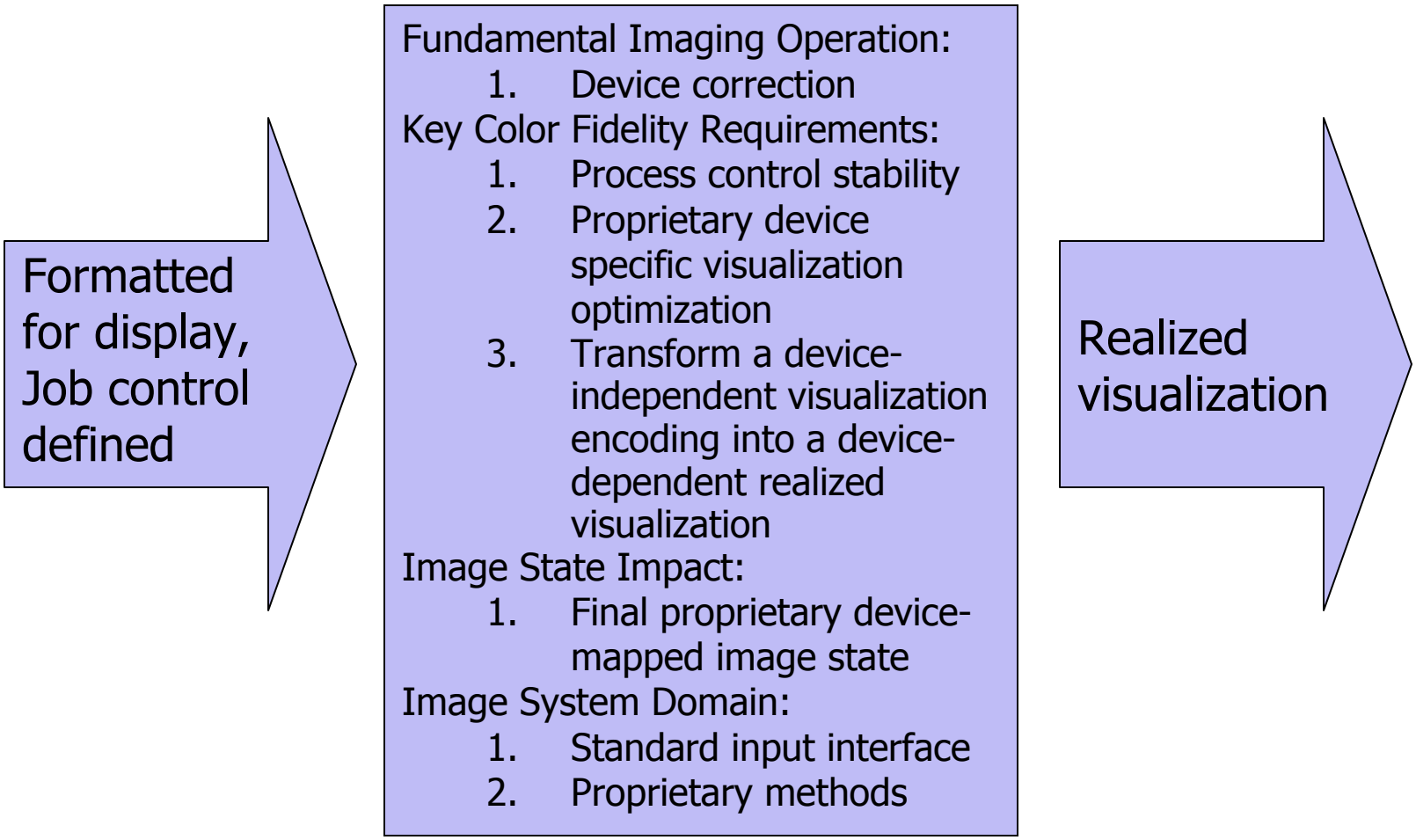
The Assemble Object/Entity Workflow Primitive 44

- Assemble Object/Entity deals with image and/or object formats; selection, format, and linkage of metadata and job information
  - Assemble Object/Entity does not change image state, or image encoding values
  - Assemble Object/Entity creates and changes object combinations using page description language (PDL) containers
  - Assemble Object/Entity embeds or links images into PDL containers
- Assemble may occur ONE or N TIMES
  - Following Capture
  - Before Adjust Image/Object
  - Following Adjust Image/Object
  - Before Color-Render
  - Following Color-Render
  - Using Visualization

The Assemble Object/Entity Workflow Primitive 45

- Assemble Object/Entity includes consideration of the color-rendering condition of an assembled entity
  - I.e., do the image states (color-render conditions) of an assembled entity and its linked components meet the submission requirements of the targeted visualization path?
  - Image or object components can be pre-color-rendered to match the submission requirements of a targeted visualization path
- Assemble Object/Entity can produce combinations of
  - Job submission protocols, job tickets, PDL files, application files
    - Containing encapsulated, embedded, or referenced image and line art elements
      - Color-Rendered Images (AOR) + Source Profile(s)
      - Reference Color-Rendered Images (ROR) + Source Profile(s)
      - Capture-referred Images + Input Profiles
      - Color-Render Deferred Images + Input/Source Profile(s) + Output Profile(s)
    - ICC profiles can be linked to specific images or can be associated with an entire object or an entire entity

The Visualization Workflow Primitive



The Visualization Workflow Primitive

- Visualization performs a final visualization-condition-specific image state transition to create a realized visualization
  - Each visualization system imposes a set of 'submission condition requirements' on received data
  - When submission conditions are not met a job may fail or may produce unexpected results
- As an image, object, or entity is used
  - Visualizations can occur sequentially leading to a final visualization
  - Visualizations can occur in parallel, leading to multiple final visualizations
  - Intermediate visualizations can occur as required for
    - Adjust
    - Color-Render
    - Assemble
  - When the submission condition requirements of an intermediate visualization and a targeted final visualization are aligned - the job can be 'proofed'

Detail on Image Processing Operations…

Fundamental image processing operations-1

- Correction operations:
  - 'Corrections' are operations that are required based on device limitations or anomalies
  - 'Corrections' can be capture side image specific adjustments that prepare an image so that it can then be processed through gamut mapping operations, appearance operations, and/or preference operations in different ways for different output options
  - Capture correction operations can be image and capture method specific and should precede color-rendering operations
  - Correction operations can also occur at the final stage of output, within the output device (e.g., output device calibration)
  - Output correction operations are device specific, and are the responsibility of the output device

Fundamental image processing operations-2

- Appearance operations:
  - Appearance operations are input to output environment specific operations
  - Examples are
    - Lightening or dynamic range compression to compensate for illumination level and surround differences
    - Unsharp masking to maintain detail contrast
  - Depending on the particular situation, a preference or gamut mapping algorithm might operate prior to or following an appearance operation

Fundamental image processing operations-3

- Preference operations:
  - Preference operations are image specific and may be output gamut constrained
  - Preference issues deal with intentional alterations of appearance to increase the aesthetic value of an image
  - The preference may pertain to a particular visualization
    - Applied in an output-referred image state
  - The preference may pertain to all visualizations
    - Applied in a capture-referred image state
  - Examples
    - Sharpening and contrast boosting
    - Selecting a saturated green grass that you know is actually within a targeted output gamut $\Rightarrow$ output constrained preference adjustment
    - Preference operations may be combined with a color-rendering operation or may be distinct

Fundamental image processing operations-4

- Gamut mapping operations:
  - Gamut mapping operations are input to output specific
    - Mapping the co-ordinates of the elements of a source image to the co-ordinates of the corresponding elements of a reproduction
    - Compensating for differences in the source and output color gamut volume and shape
      - E.g., Perceptual PCS to visualization output gamut
    - Gamut mapping operations fit the results of appearance and preference operations into actual device/medium gamuts
      - Yeilds best results when the operation is image specific
    - The term 'gamut mapping' includes the simple case of clipping at the gamut boundary as must occur when going from a larger gamut to a smaller gamut even in the media-relative colorimetric Rendering Intent case

Five components of the Digital Smart Factory: Color management perspective

- **Smart Content**: The incorporation of intelligent capability within content files or applications to create value = ICC profiles and rendering intent selections.
- **Control**: The implementation of Computer Integrated Manufacturing for the purpose of automation, productivity, capability, and predictability = use of JDF 1.2 and ICC profiles in automated color management solutions.
- **Commerce**: The identification of needed protocols, standards, and practices that will assist the industry in transacting business orders, payments, and specifications = JDF 1.2.
- **Architecture**: The investigation and recognition of the required hardware and software platforms for creating an optimizing environment for integration and integrity = ICC workflow semantic model and color control architecture.
- **Infrastructure**: The identification of management principles, training, and skill sets required to create a digital smart factory environment = ICC color management in a JDF 1.2 environment.