The background of the image is a complex, abstract composition. It features numerous glowing, translucent spheres of various sizes scattered throughout the space. These spheres emit a soft light, some with a blue hue and others with a red or orange glow, creating a sense of depth and motion. Interspersed among the spheres are numerous thin, dark, intersecting lines that form a grid-like structure. In the foreground, there are larger, more prominent geometric shapes, possibly representing architectural elements or abstract forms. One such shape in the lower-left corner is a large, multi-colored polygon with shades of red, orange, and yellow. Another shape in the upper-right corner is a large, light-colored polygon with a grid pattern. The overall effect is one of a futuristic, digital, or scientific environment.

# FOUNDRY.

imagination engineered



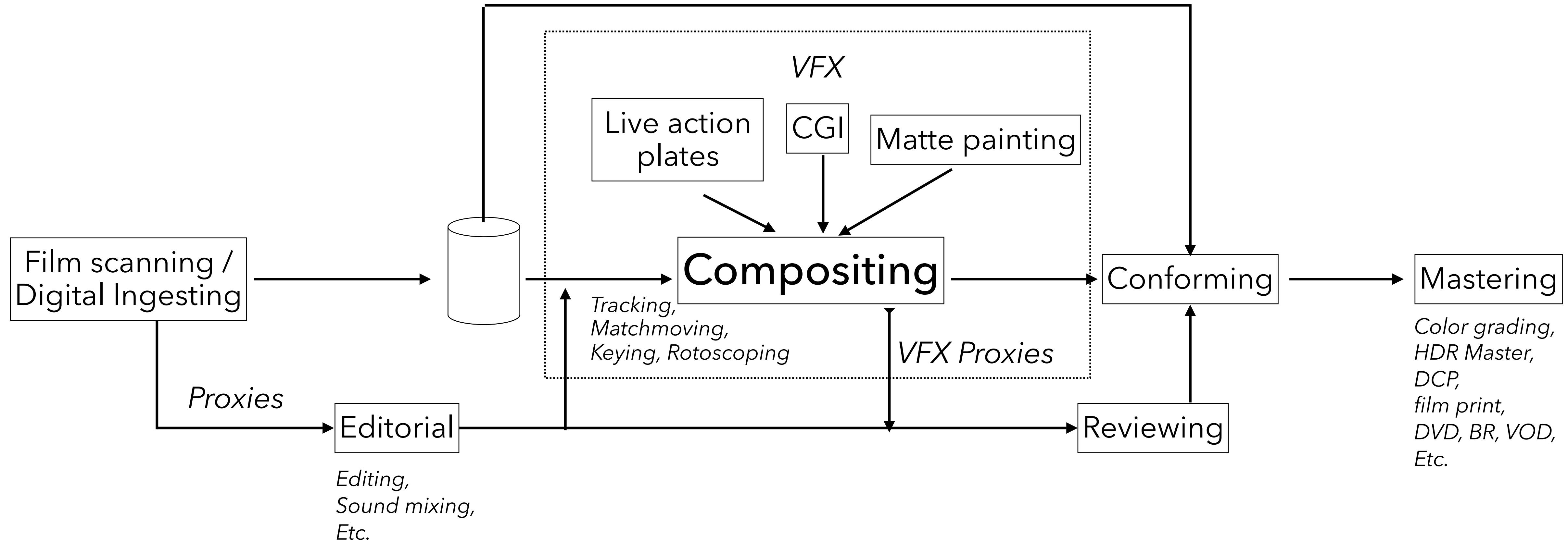
# Introduction to NUKE.

Guillaume Gales  
*Research Engineer*

[guillaume.gales@foundry.com](mailto:guillaume.gales@foundry.com)

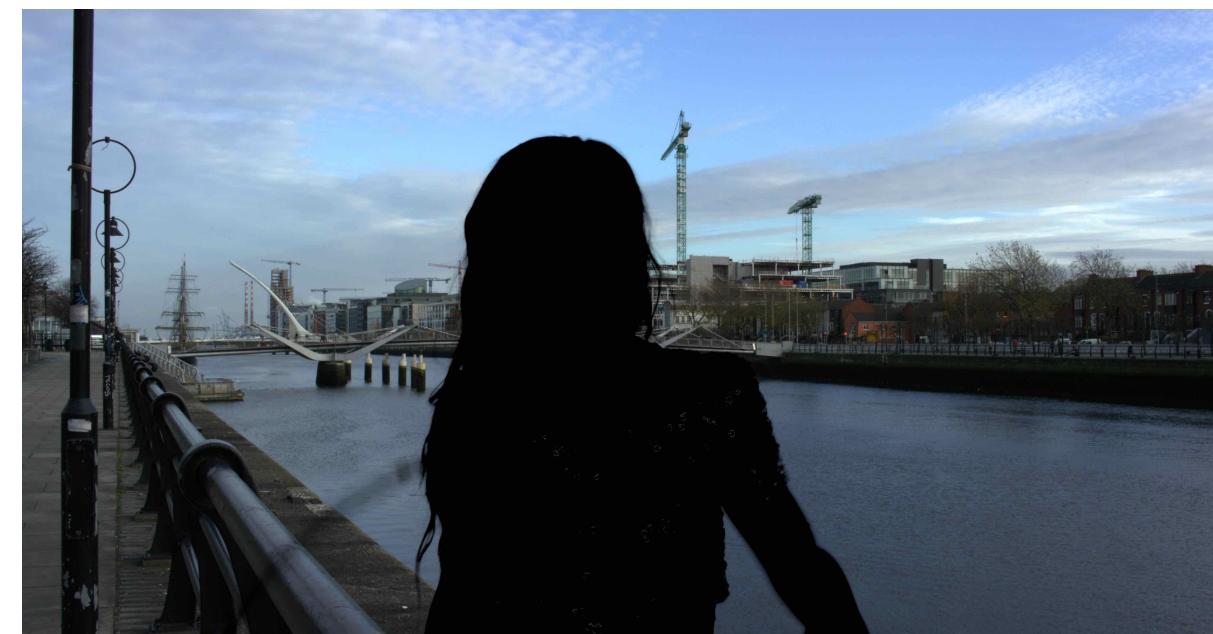
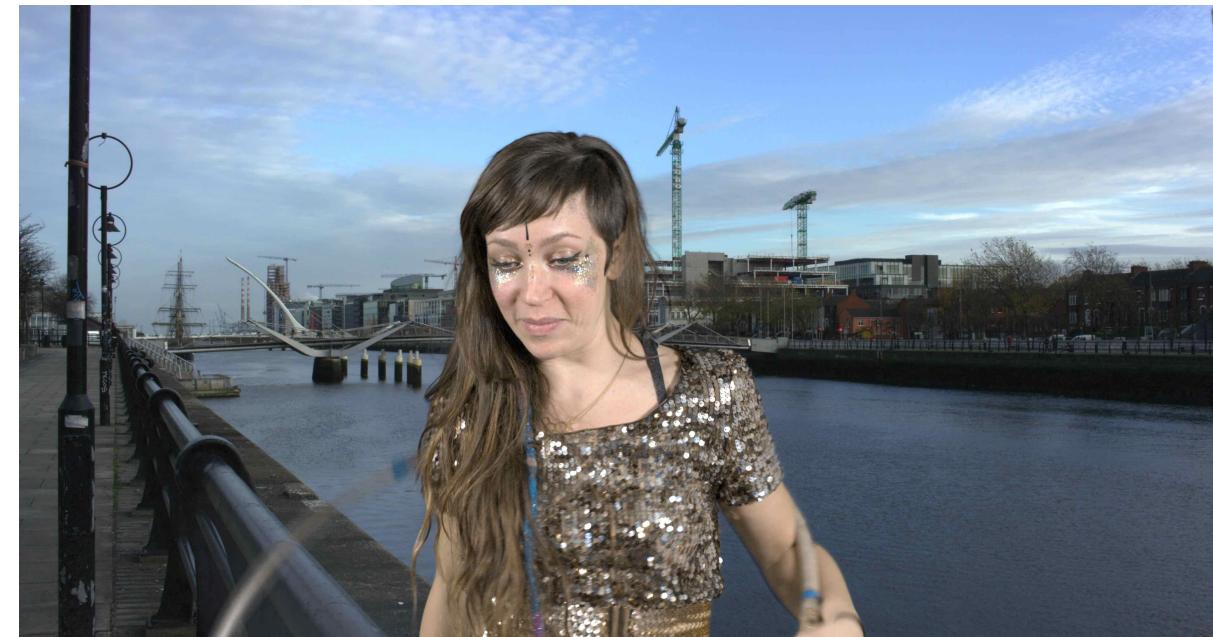
# What is Compositing?

- NUKE is the industry standard digital compositing software
- Compositing consists in putting together different elements (CGI, matte paintings, etc.) into a single shot.



# Compositing Example

- Compositing two plates foreground (F) and background (B):  $C = \alpha F + (1 - \alpha)B$ 
  - where alpha is the foreground opacity



F.

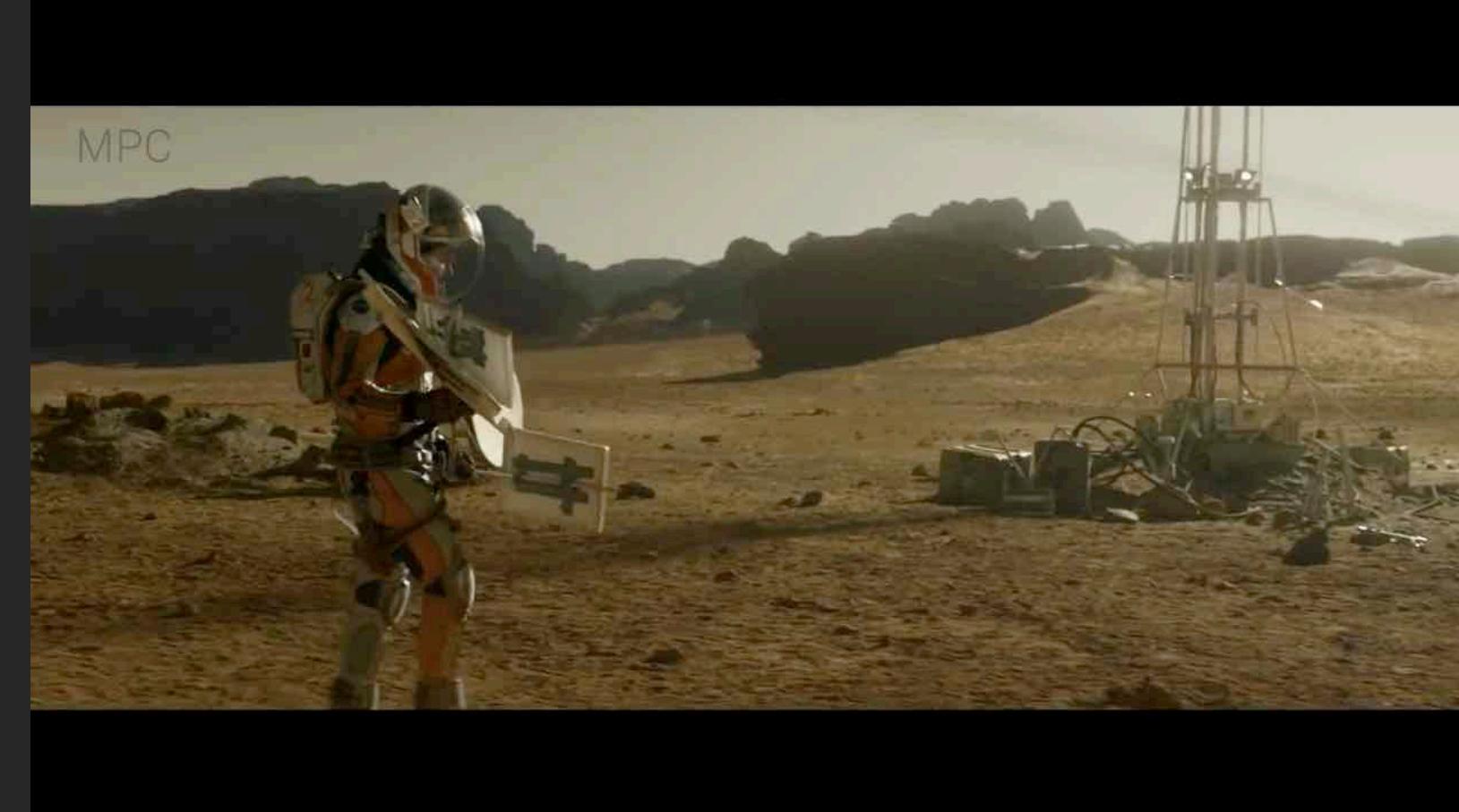


<https://www.youtube.com/watch?v=SabPuZ5coKk>

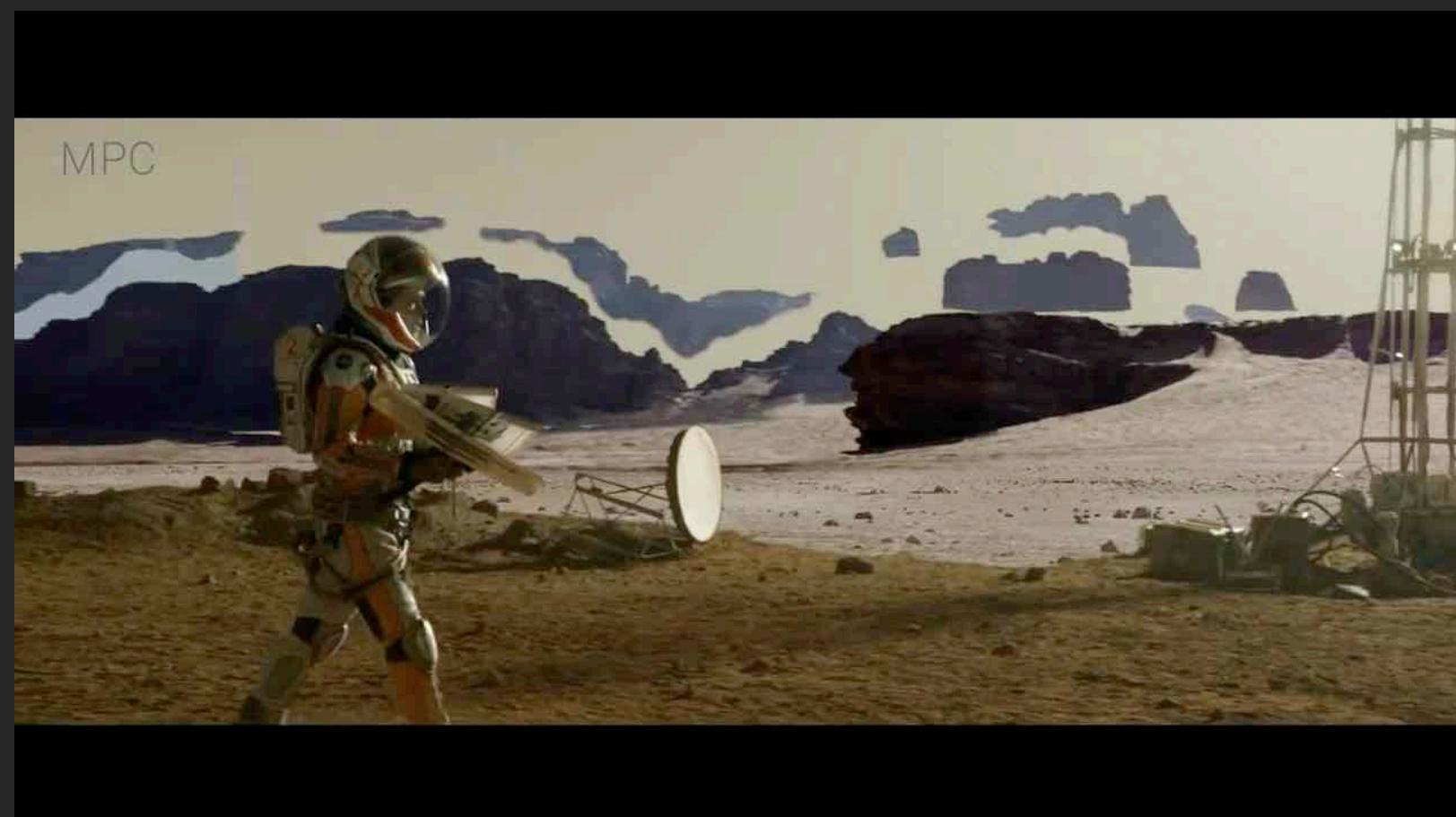
F.



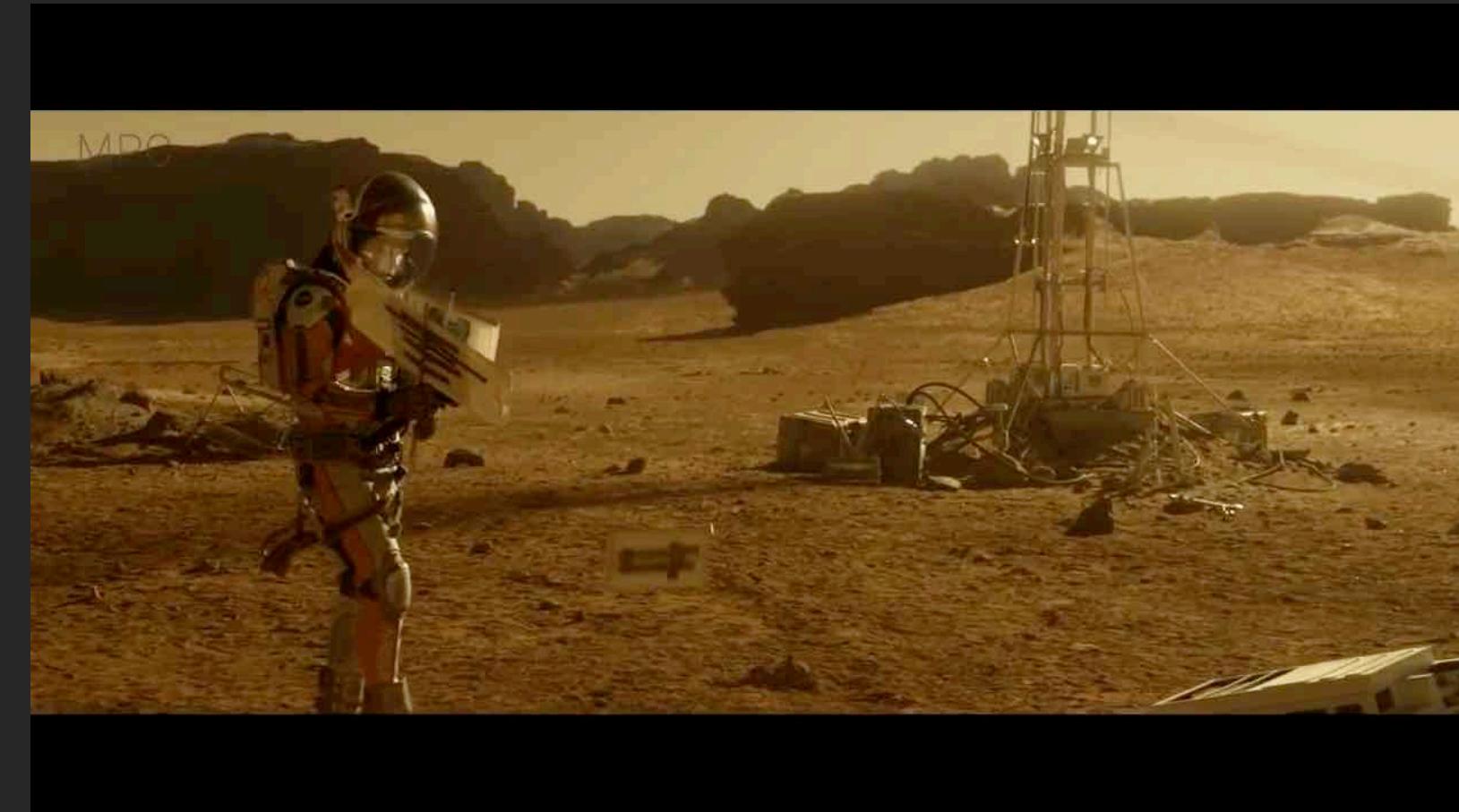
*Keying,  
Rotoscoping  
> alpha mattes*



*Add atmospheric  
layers*

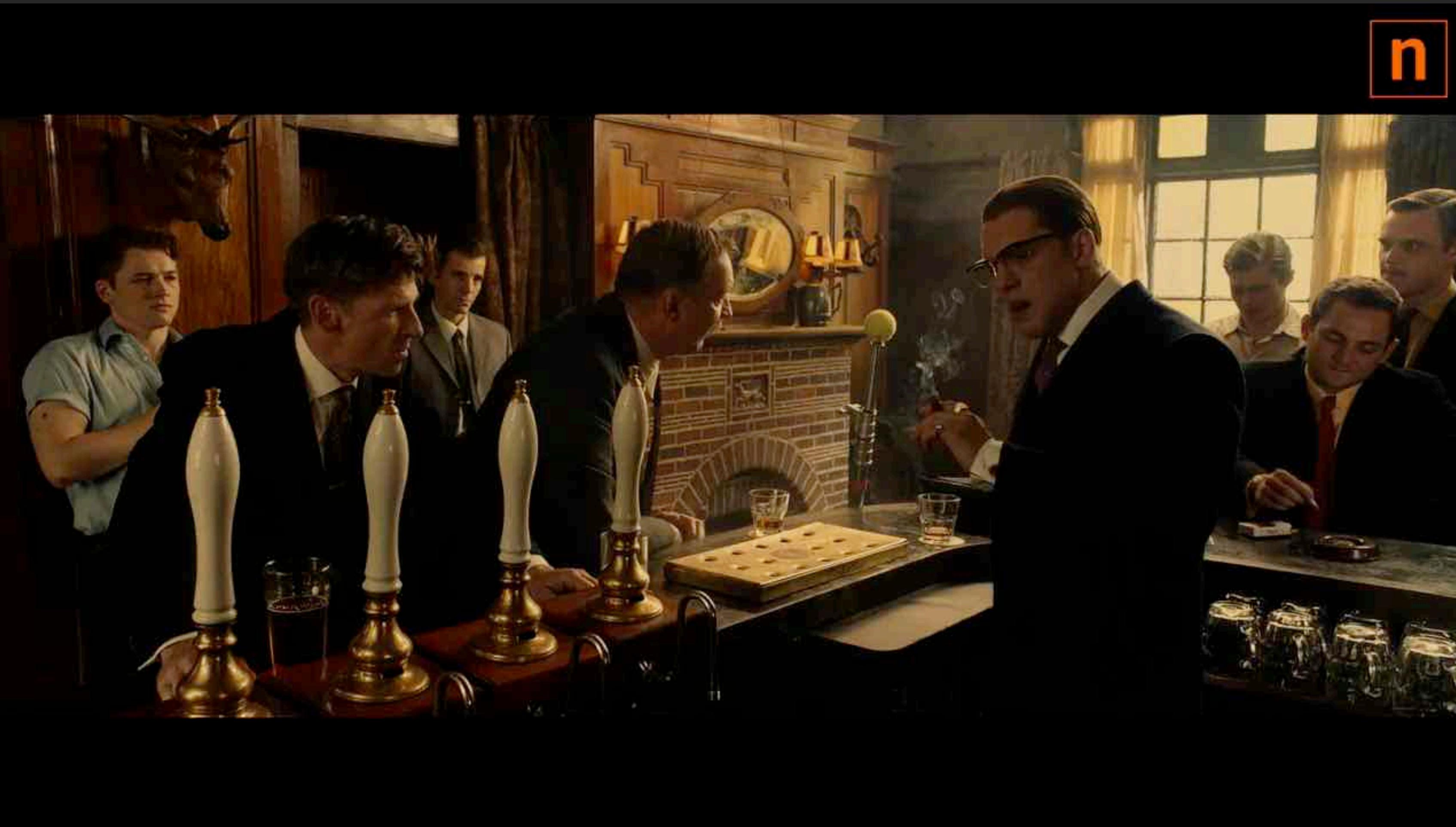


*Use mattes to  
integrate painted  
background  
elements*



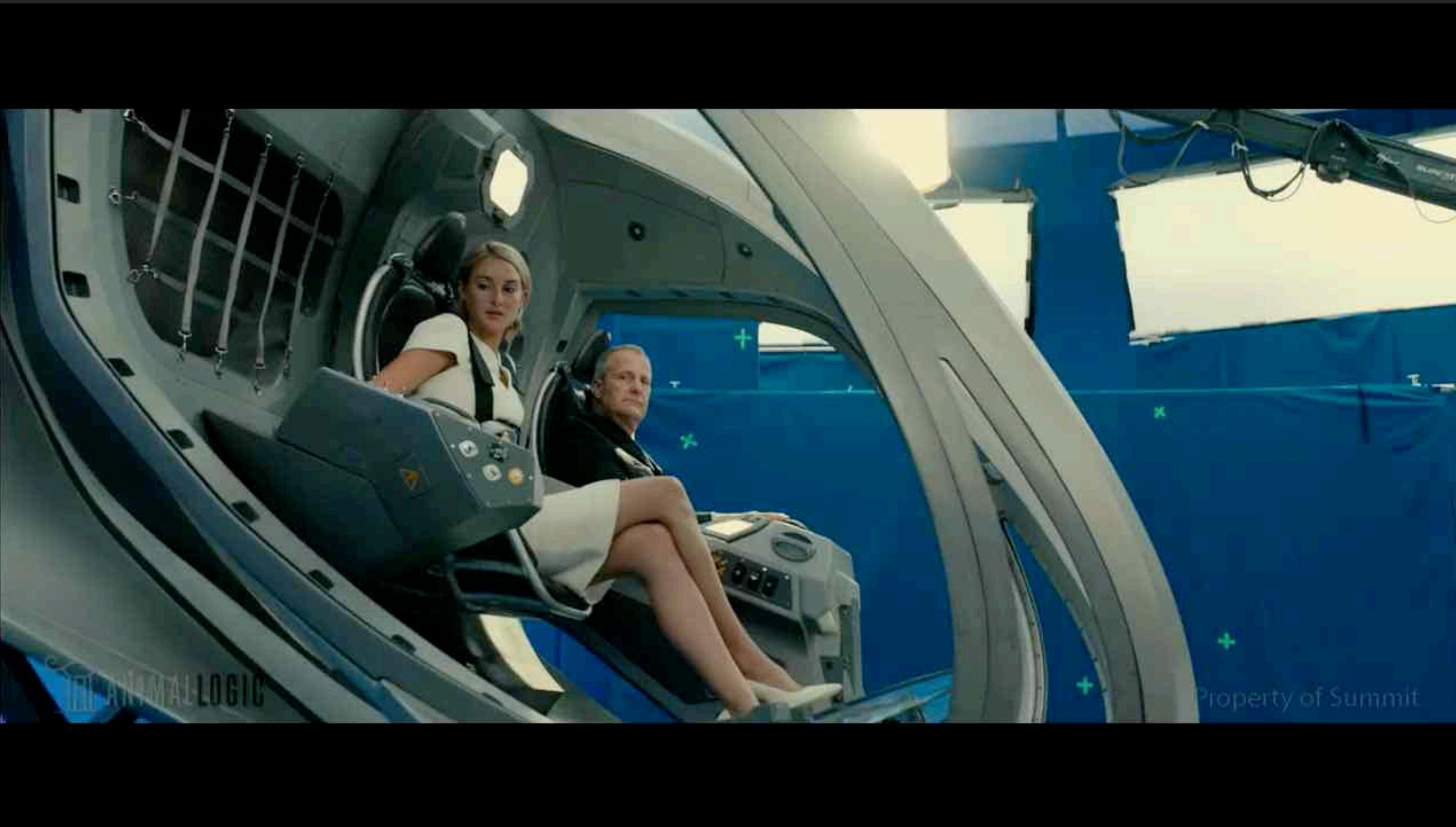
*Colour correction to  
simulate Mars lighting*

F.



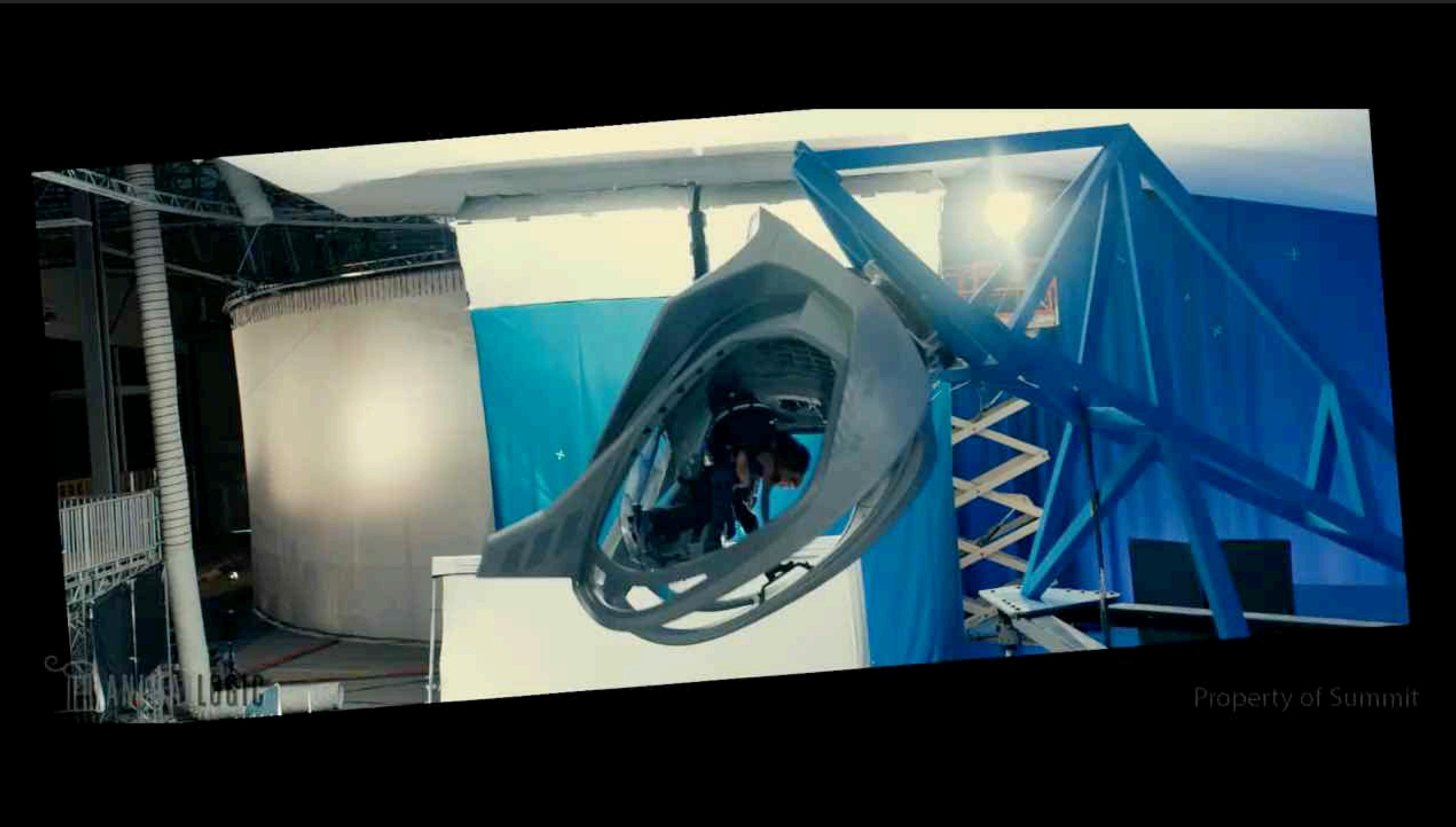
<https://www.youtube.com/watch?v=SabPuZ5coKk>

F.



<https://www.youtube.com/watch?v=SabPuZ5coKk>

F.



Property of Summit

<https://www.youtube.com/watch?v=SabPuZ5coKk>

F.



<https://www.youtube.com/watch?v=SabPuZ5coKk>

F.



<https://www.youtube.com/watch?v=SabPuZ5coKk>



# Introduction to NUKE

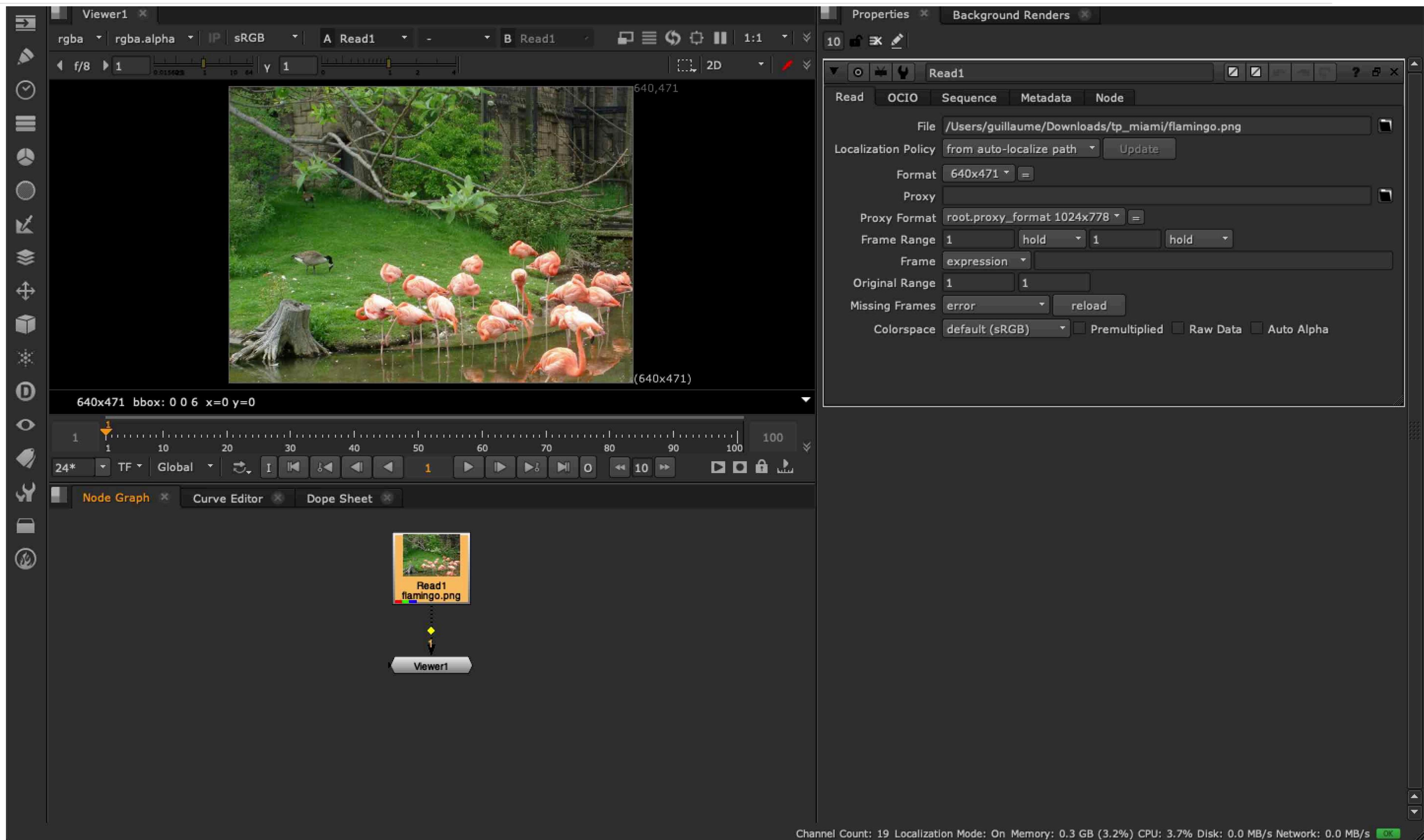
---

- Node-based application
  - Each node performs one operation (e.g. Math operations, Convolutions, etc.)
  - **Nodes combined together make graphs of flexible, complex and powerful image processing pipelines**
  - this flexibility also allows prototyping video processing algorithms
- <https://www.foundry.com/products/nuke/tutorials>
- <https://help.thefoundry.co.uk/nuke/11.0/>



# Introduction to NUKE

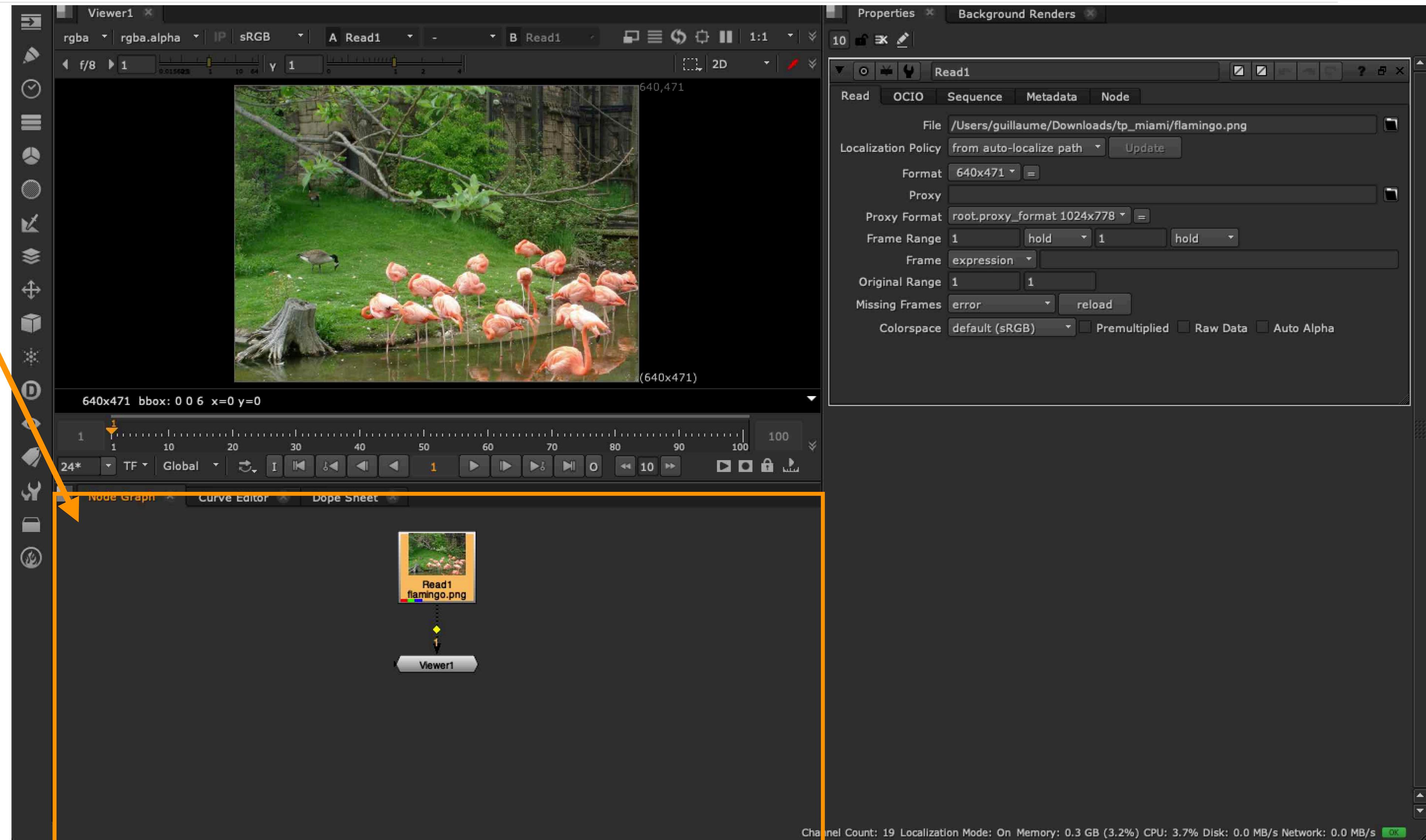
- NUKE interface
- Node Graph panel
  - Read node
- Viewer
  - RGBA channels
  - Color sampler
  - Gamma / Gain
- Properties panel





# Introduction to NUKE

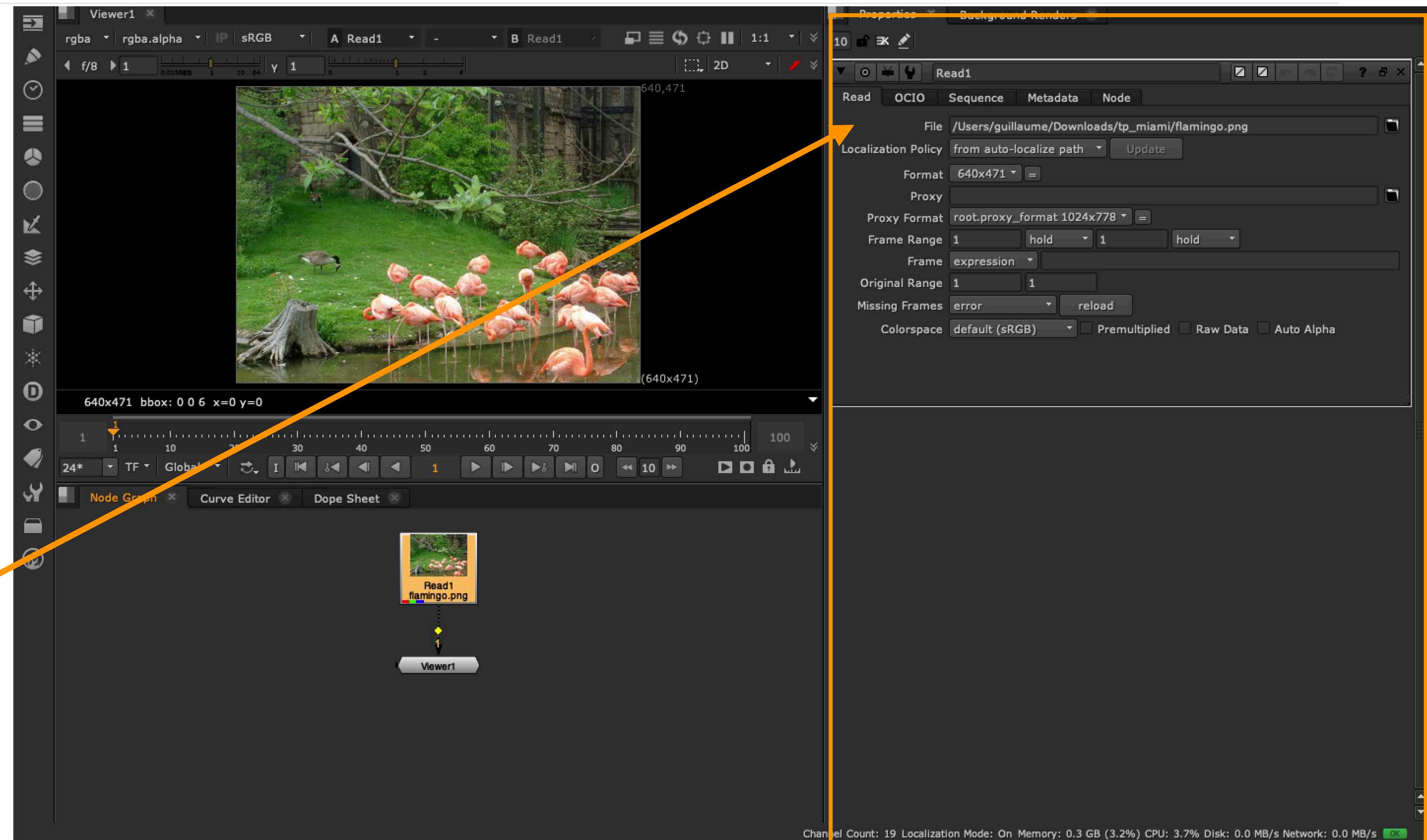
- NUKE interface
- Node Graph panel
  - Read node
- Viewer
  - RGBA channels
  - Color sampler
  - Gamma / Gain
- Properties panel





# Introduction to NUKE

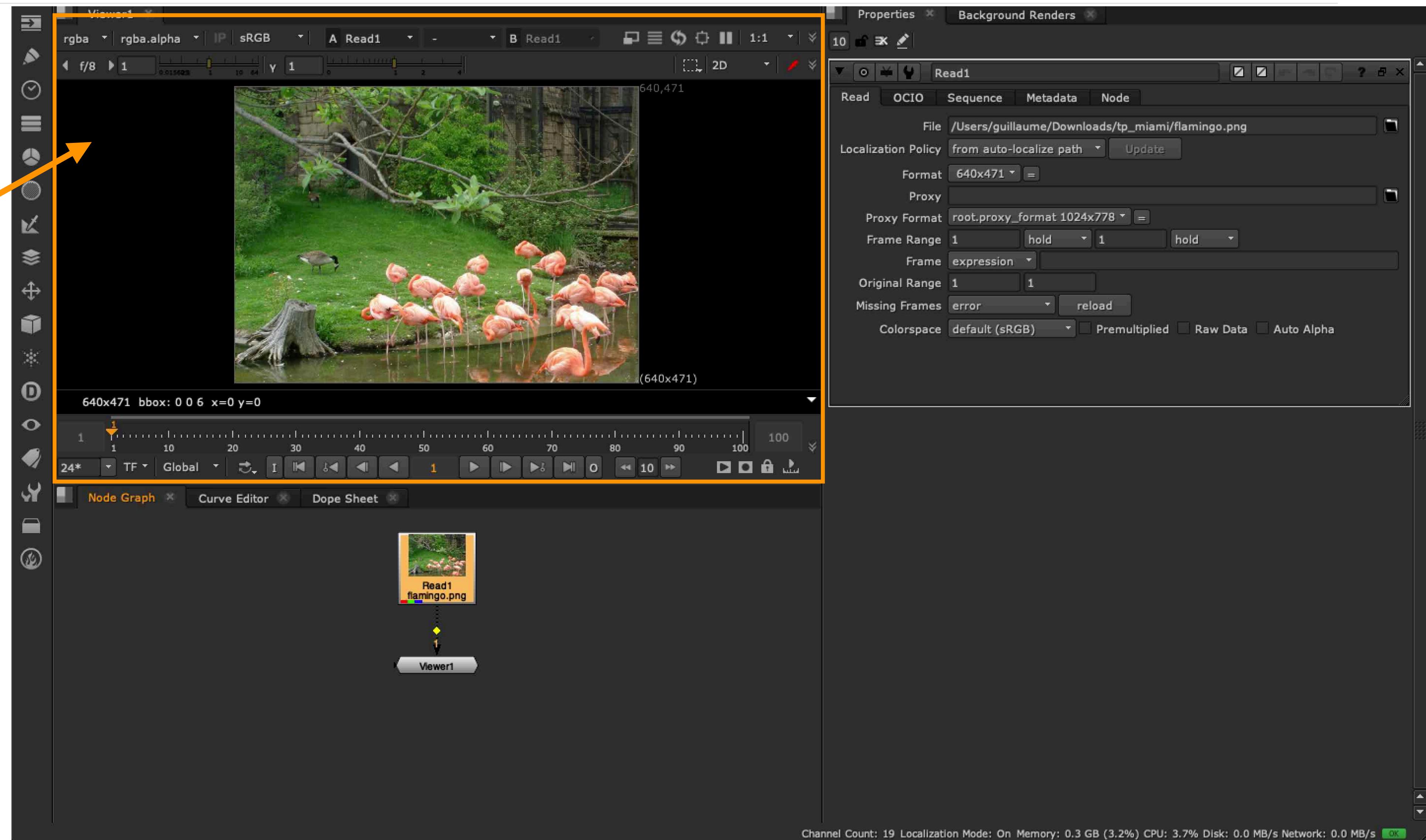
- NUKE interface
- Node Graph panel
  - Read node
- Viewer
  - RGBA channels
  - Color sampler
  - Gamma / Gain
- Properties panel





# Introduction to NUKE

- NUKE interface
- Node Graph panel
  - Read node
- Viewer
  - RGBA channels
  - Color sampler
  - Gamma / Gain
- Properties panel





# Introduction to NUKE

---

- *Expression Node* for pixel-wise operations
  - Applies the given mathematical expressions to each pixel
  - Expressions uses a C-like syntax
  - Reserved variables for each channel:
    - r: red
    - g: green
    - b: blue
    - a: alpha
  - Other reserved variables:
    - x: pixel x position (origin is at the bottom left corner)
    - y: pixel y position
    - width: image width
    - height: image height

# Introduction to NUKE

- Expression Node
  - Example 1: Colour inversion



Expressions

Expression2

Expression Node

channels: **rgb**   red   green   blue   none   =  
= 1-r  
channels: **rgb**   red   green   blue   none   =  
= 1-g  
channels: **rgb**   red   green   blue   none   =  
= 1-b  
channels: **none**   rgba.alpha   =  
=

mask: none   =   inject   invert   fringe  
(un)premult by: none   =   invert  
mix luminance: **0**   0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1  
mix: **1**   0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1

# Introduction to NUKE

- Expression Node
  - Example 2: Colour to luminance  $L = 0.2126R + 0.7152G + 0.0722B$



Define constants

Expression1

Expression Node

rc	= 0.2126
gc	= 0.7152
bc	= 0.0722
	=

channels **rgb** red green blue none =  
 $= rc*r + gc*g + bc*b$

channels **rgb** red green blue none =  
 $= rc*r + gc*g + bc*b$

channels **rgb** red green blue none =  
 $= rc*r + gc*g + bc*b$

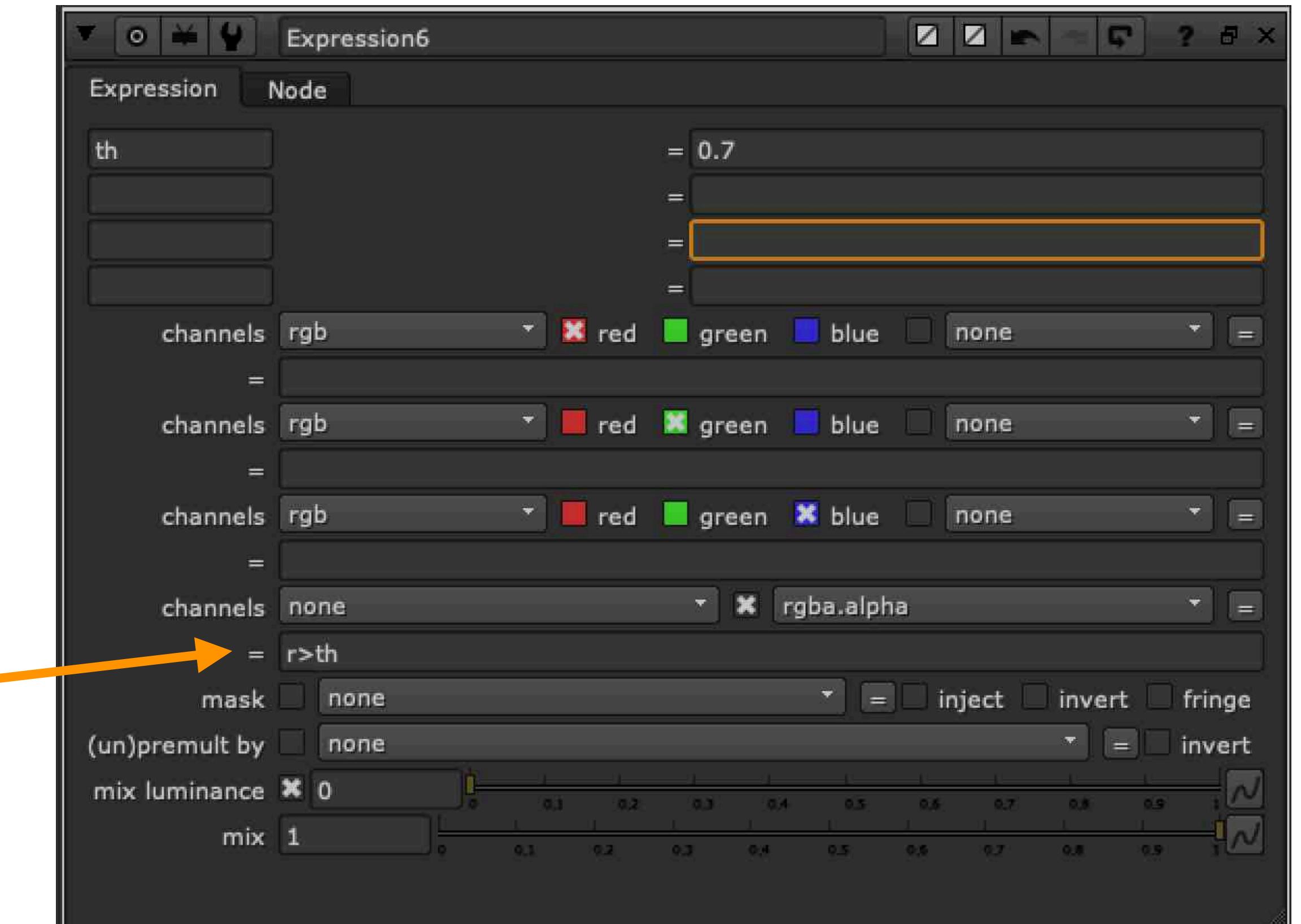
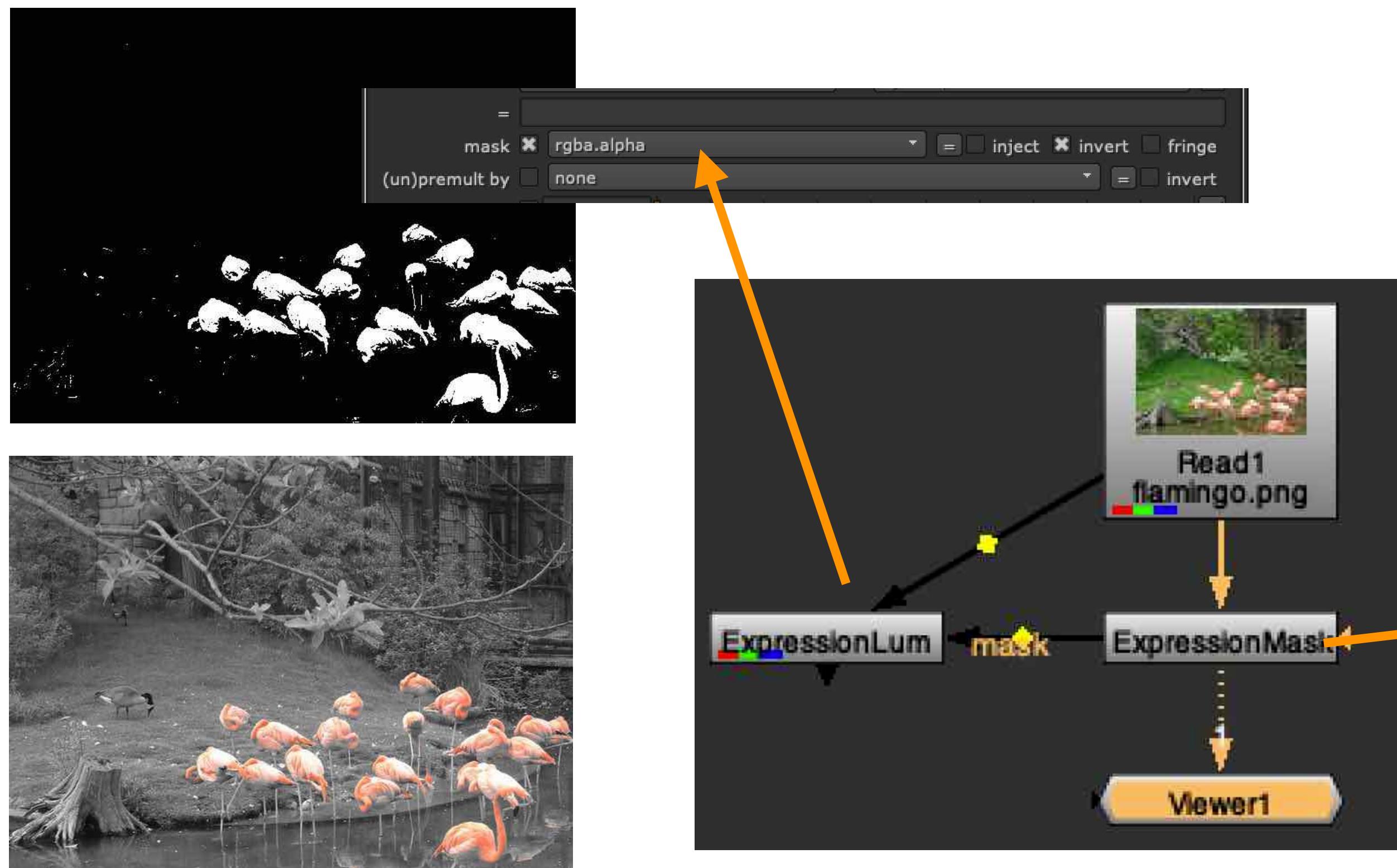
channels **none** rgba.alpha =  
 $=$

mask none = inject invert fringe  
(un)premult by none = invert

mix luminance **0** 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 =  
mix **1** 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 =

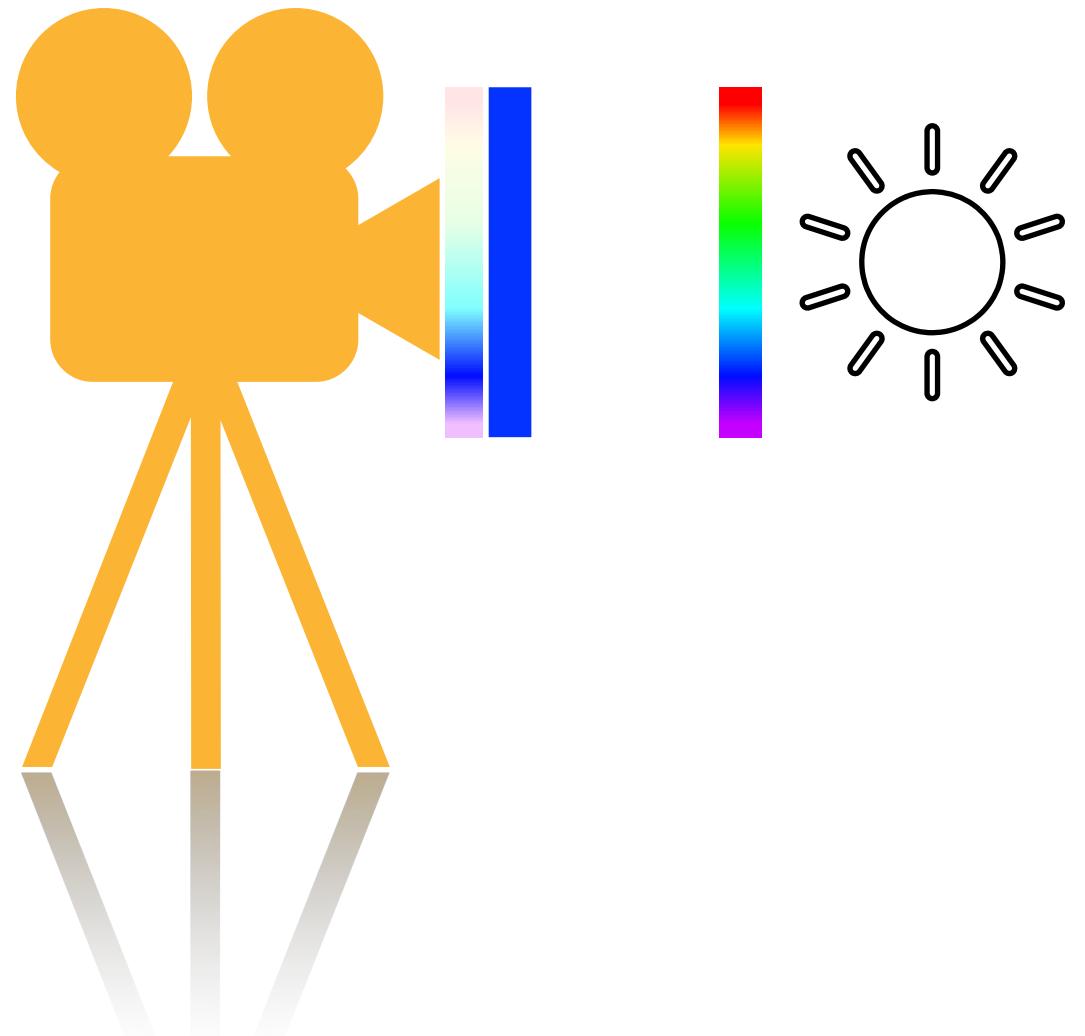
# Introduction to NUKE

- Expression Node with Mask input
  - **Example 3:** Apply desaturation to non-pink pixels
    - 1. Use the red channel to generate a mask for pink pixels
    - 2. Use the mask input of the expression node from the previous example to apply the effect to non-pink pixels only



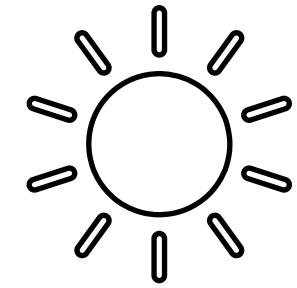
# Introduction to NUKE

- Expression Node 
- **Exercise 1a:** Day-for-night
  - This technique simulates night when shooting in bright daylight. It is achieved in-camera by placing a blue filter in front of the lens.
  - Use an expression node to emulate this effect in NUKE.



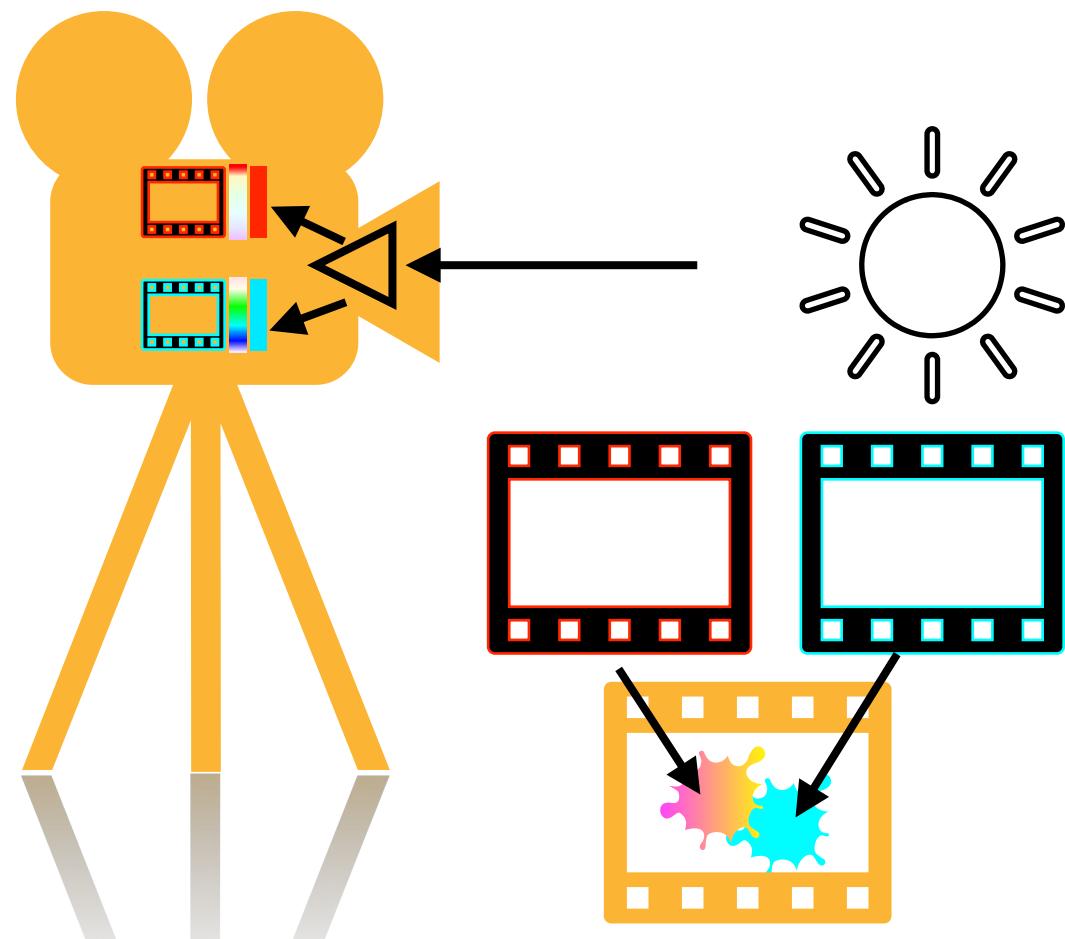
# Introduction to NUKE

- Expression Node
  - **Exercise 1b:** Day-for-night
    - Modify the previous exercise such that the effect is gradually applied from the top to the bottom
    - *Hint:* you may use one, or two expression nodes.



# Introduction to NUKE

- Expression Node
  - **Exercise 2:** Two-strip technicolor
    - This early technique was used to capture and reproduce colours. It was achieved using a prism to split the image in 2. A different coloured filter (red and blue-green) is placed in front of each side and a print is realised combining both strips of film with the corresponding dye.
    - Because only two “primary” colours are used, the resulting colour are not very natural.
    - Use an expression node to emulate this effect in NUKE.

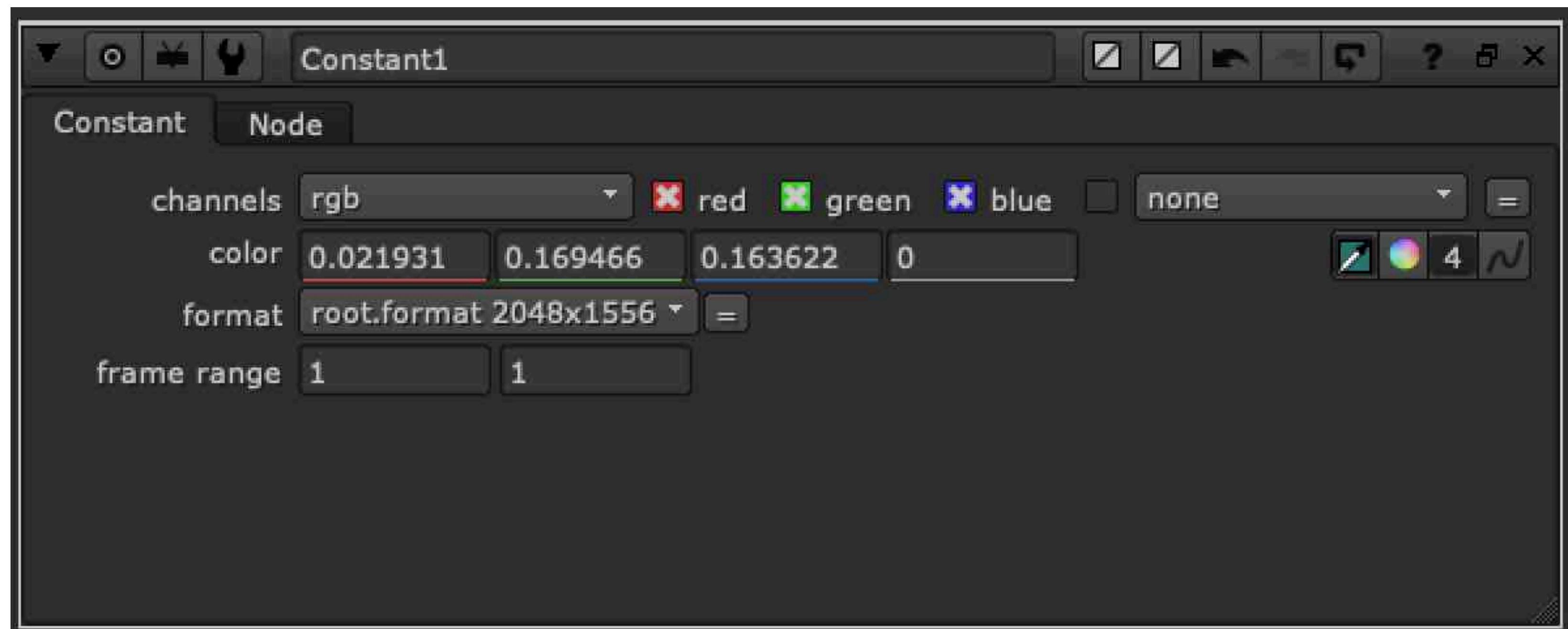


*Hint:* Print is a subtractive colour system  
but we can abstract it here.



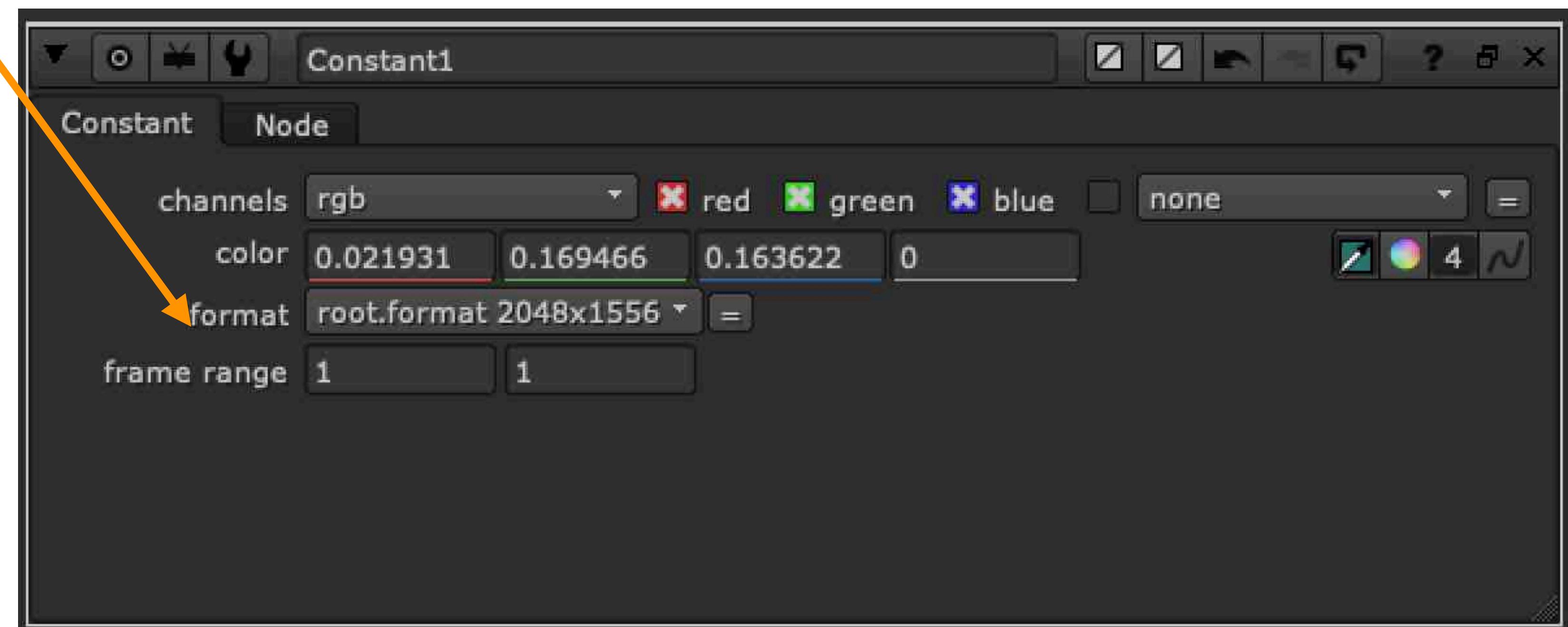
# Introduction to NUKE

- *Constant Node*
  - Allows to generate an image with a constant values for specified channels
    - Output format
    - Colour picker



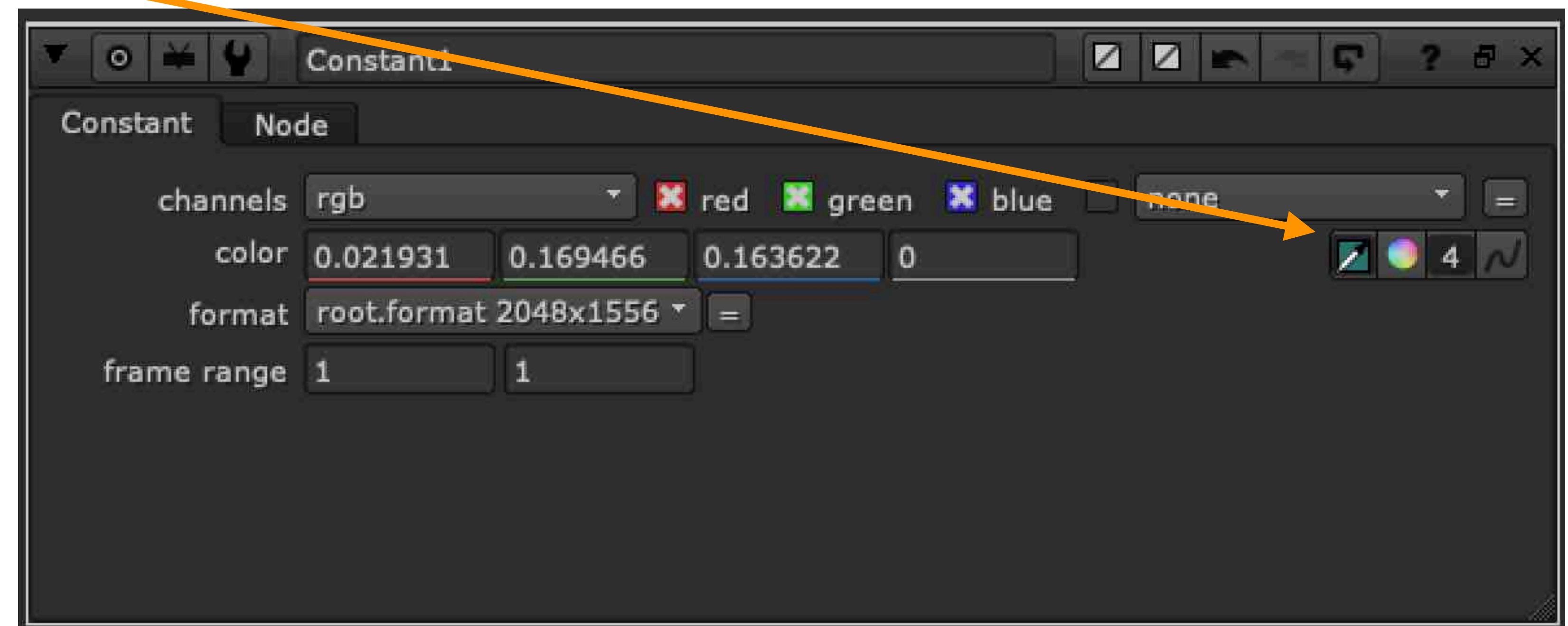
# Introduction to NUKE

- *Constant Node*
  - Allows to generate an image with a constant RGBA value
    - Output format
    - Colour picker



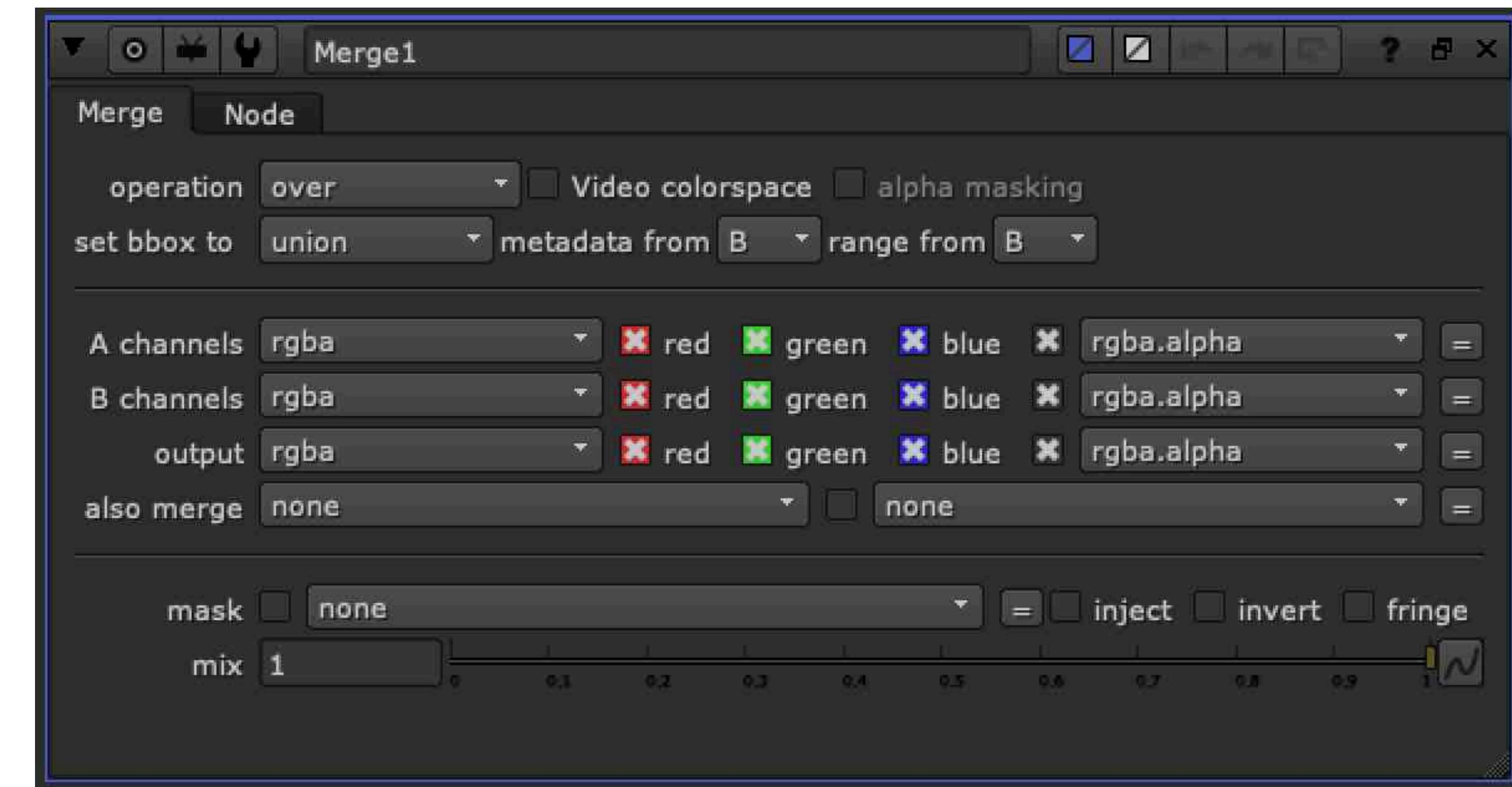
# Introduction to NUKE

- *Constant Node*
  - Allows to generate an image with a constant RGBA value
    - Output format
    - Colour picker



# Introduction to NUKE

- *Merge Node*
  - Combines two images A and B with the specified operation
  - Common operations:
    - Average  $(A+B)/2$
    - Difference  $\text{abs}(A-B)$
    - Minus  $A-B$
    - Multiply AB
    - Plus  $A+B$
    - ...
  - Hint: hover the mouse over the operation menu to display the formula for each operation





# Introduction to NUKE

---

- *Spatial operators*
  - Convolutions:
    - Using a matrix *Matrix*
    - Using an image of a kernel *Convolve*

# Introduction to NUKE

- *Edge detector*
  - **Exercise 3:** Propose a graph to build a simple edge detector 



# Introduction to NUKE

- *Edge detector*
  - **Exercise 4:** Propose a graph to emulate a shallow depth of field

