

ASSETS AND MODELS

FORMATS, MODELING, ASSETS

TOMÁŠ POLÁŠEK IPOLASEK@FIT.VUTBR.CZ

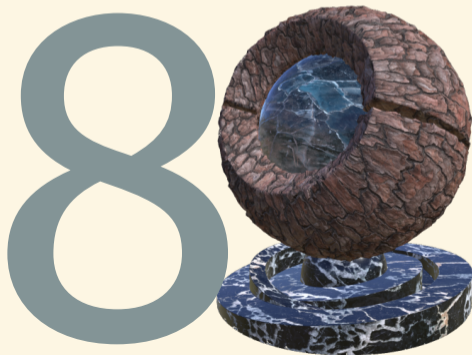
BRNO UNIVERSITY OF TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

DCGM, CPhoto@FIT

FACULTY OF FINE ARTS

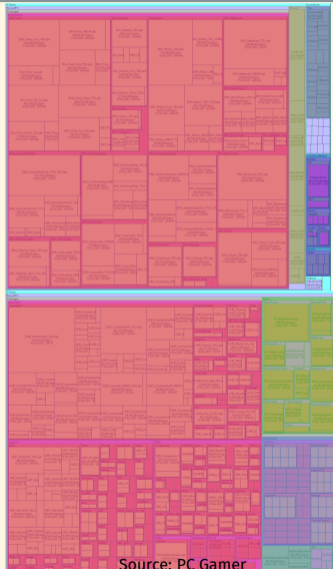
GAME MEDIA STUDIO



GAME ASSET MANAGEMENT

GAME ASSETS AND MANAGEMENT

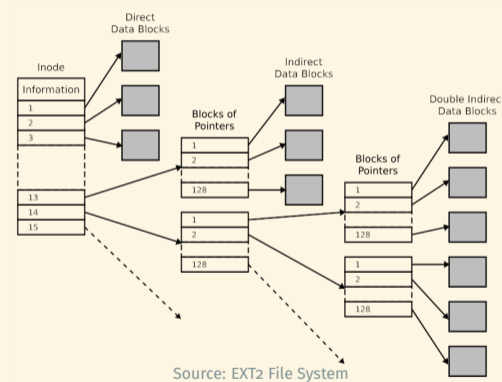
- Game → Multimedia Experience
- Assets & Resources
 - ▶ Visual: Materials, Models, Animations
 - ▶ Aural: Audio, Music, Effects
 - ▶ Structural: Levels, Worlds, Relations
 - ▶ Programmatic: Scripts, Shaders, Libraries
 - ▶ Specific: Collisions, Physics, Layouts
- Problems:
 - ▶ Varied Formats
 - ▶ Optimal Representation
 - ▶ Storage × Runtime
 - ▶ Limited Memory
 - ▶ Load Scheduling
 - ▶ Asset Streaming
- → Asset Manager



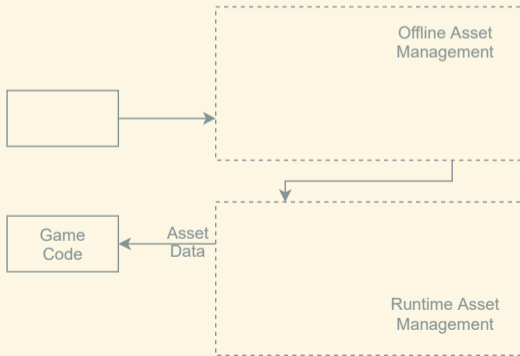
Source: PC Gamer

FILE SYSTEM

- Goal: Access Files
- File Path
- Operations
 - ▶ Path Manipulation
 - ▶ File I/O
 - ▶ Mapping
 - ▶ Streaming
- Wrapper API
- Synchronous & Asynchronous IO
- Virtual File System

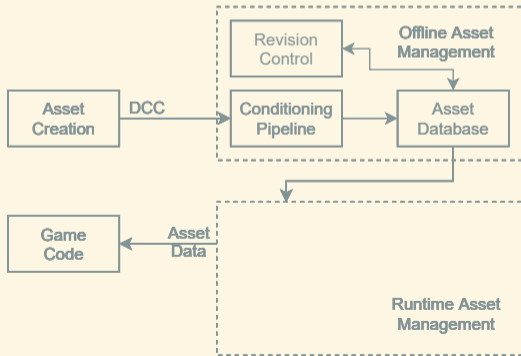


- Asset Management Layer [2]
- Two Components
 - **Offline**
 - **Runtime**
- File System Wrapper
 - Identification \rightsquigarrow “Paths”
 - Asset Data \rightsquigarrow “File I/O”
 - Composites \rightsquigarrow “Mapping”
 - Loading \rightsquigarrow “Streaming”
- Platform Independence



OFFLINE ASSET MANAGEMENT

- Goal: Pre-Process Everything
- DCC → Raw Data
- Conditioning → Assets
- Indexing → Database
- Revision Control



ASSET CREATION PIPELINE

- Digital Content Creation
- Imparts Style
- Tool Workflow → Pipeline
- Inter-Tool Compatibility
- Workflow Automation
- Programming Pipeline



ASSET CONDITIONING

■ Goal: Ease Asset Utilization

- ▶ Data Compression
- ▶ MipMap Generation
- ▶ Retopologizing

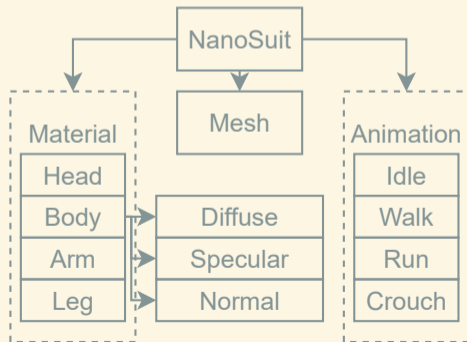
■ Raw DCC Data

■ Prescription Pipeline

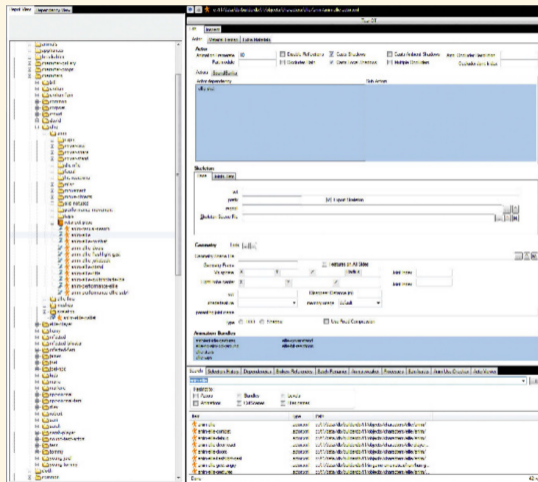
1. Data Exporter
2. Resource Compiler
3. Asset Linker

■ Dependency Graph

■ Specialized Formats



- Configuration, Store, Connections
- Management & Modification
- Configuration Metadata
- Cross-References & Integrity
- Revision History



Source: Naughty Dog [2]

■ Varied Approaches

- ▶ Packaged × Loose Files
- ▶ Text × Binary Metadata
- ▶ Offline × Runtime
- ▶ Physical × Virtual FS
- ▶ Asset Compression
- ▶ Flat × Overlay Hierarchy
- ▶ Custom × General Formats

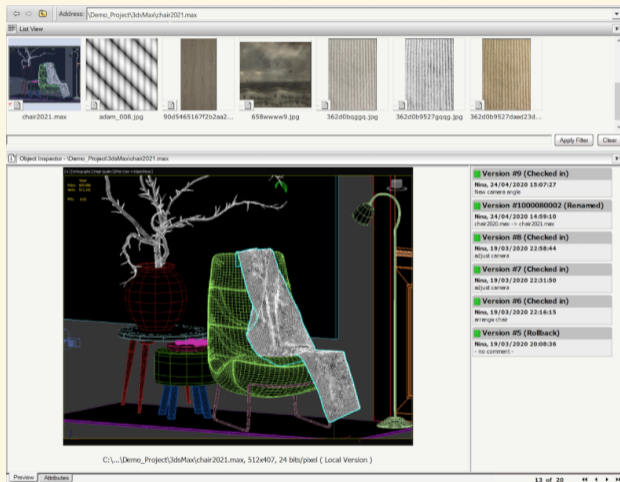
■ Examples:

- ▶ Unity
- ▶ Unreal Engine 4
- ▶ Naughty Dog Engine [2]
- ▶ Overwatch [1]
- ▶ OGRE

```
fileFormatVersion: 2
guid: c2a3d4c9951f4dc4482c20e67c651897
ModelReporter:
  serializedVersion: 21202
  internalIDToNameTable: []
  externalObjects:
    - first:
        type: UnityEngine:Material
        assembly: UnityEngine.CoreModule
        name: Ceiling
        second: (fileID: 2180000, guid: e64feac767ea7e14ba813147acb9b2e9)
    materials:
      - *
    animations:
      - *
    meshes:
      - *
    assetBundleName:
    assetBundleVariant:
  arm_diff.png PNG File 532 KB
  arm_diff.png.meta META File 3 KB
  arm_showroom_diff.png PNG File 846 KB
  arm_showroom_diff.png.meta META File 3 KB
  arm_showroom_spec.png PNG File 1,629 KB
  arm_showroom_spec.png.meta META File 3 KB
  body_diff.png PNG File 596 KB
  body_diff.png.meta META File 3 KB
  body_showroom_diff.png PNG File 840 KB
  body_showroom_diff.png.meta META File 3 KB
  body_showroom_spec.png PNG File 1,726 KB
  body_showroom_spec.png.meta META File 3 KB
  gles_diff.png PNG File 1 KB
  gles_diff.png.meta META File 3 KB
  hand_diff.png PNG File 612 KB
  hand_diff.png.meta META File 3 KB
  hand_showroom_spec.png PNG File 1,670 KB
  hand_showroom_spec.png.meta META File 3 KB
  helmet_showroom_diff.png PNG File 730 KB
  helmet_showroom_diff.png.meta META File 3 KB
  helmet_showroom_spec.png PNG File 1,529 KB
  helmet_showroom_spec.png.meta META File 3 KB
  leg_diff.png PNG File 1,110 KB
  leg_diff.png.meta META File 3 KB
  leg_showroom_diff.png PNG File 1,339 KB
  leg_showroom_diff.png.meta META File 3 KB
  leg_showroom_spec.png PNG File 1,731 KB
  leg_showroom_spec.png.meta META File 3 KB
```

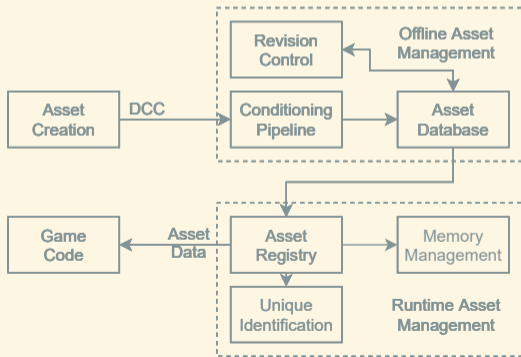
REVISION CONTROL

- History & Changes
- Varied Formats
- Large Files
- Version Control System
 - ▶ GIT + LFS
 - ▶ Plastic SCM
 - ▶ Perforce
 - ▶ AlienBrain



RUNTIME ASSET MANAGEMENT

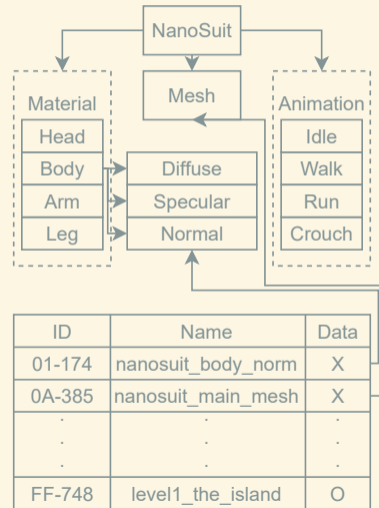
- Goal: Provide Access to Assets
- Register Assets
- Indexing & Identification
- Memory Management
- Principles:
 - ▶ One Copy
 - ▶ Lifetime & Garbage Collection
 - ▶ Composition & References
 - ▶ Streaming



- Unique Identifier → GUID
- Name Uniqueness
- Relocation Resistance
- Revision Consistency
- Various Solutions
 - ▶ Asset Path : “:/assets/level1/meshes/character.fbx”
 - ▶ Integer ID : AAF83-82F-ABC1ABD6-5374-8A52CCF62631
 - ▶ Hash : 3c643b93398b9dbb8boce094c106de2b

ASSET REGISTRY

- Goal: Catalogue Assets
- Dictionary or Database
- At Most Single Copy
- Asset Initialization



ASSET REGISTRY DECISIONS

■ Loading Options

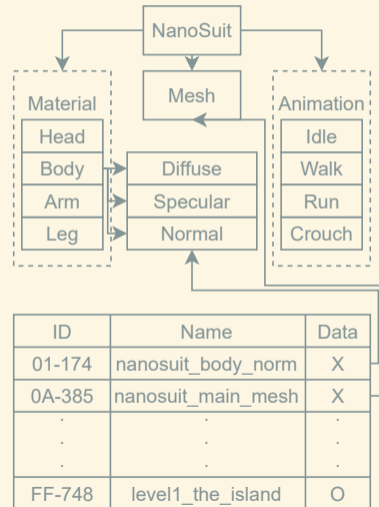
- ▶ Immediate Load
- ▶ Pre-Load Data
- ▶ Asynchronously

■ Managing Lifetime

- ▶ Preset Limit
- ▶ Lazy Loading
- ▶ Garbage Collector
- ▶ Memory Allocation

■ Referential Integrity

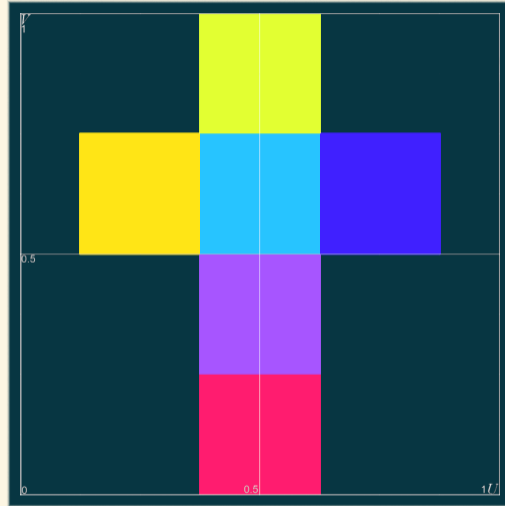
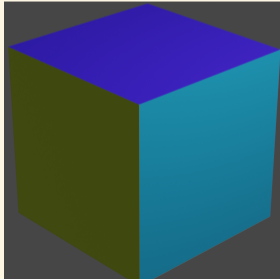
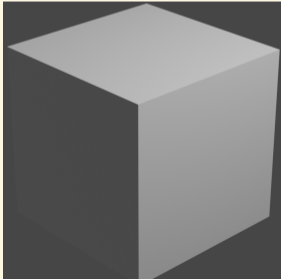
- ▶ Composite vs External
- ▶ Dependency graph
- ▶ GUID × Pointers



MODEL CREATION PIPELINE

UV MAPPING

- Texture → Model
- Unwrapping
 - Seams
 - Projection
- UV Coordinates



DETAIL MAPPING

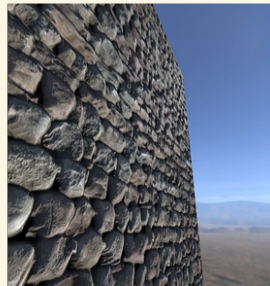
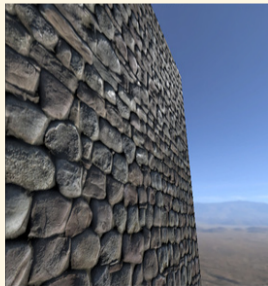
■ Geometry Complexity

- Detail
- Performance

■ Retopologizing

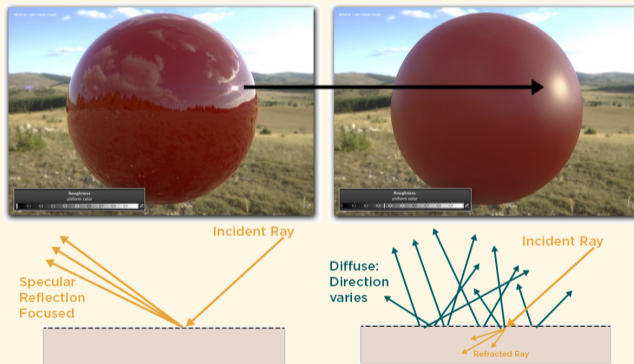
■ Mapping Details

- Normal Mapping
- Bump Mapping
- Parallax Mapping



PHYSICALLY BASED RENDERING

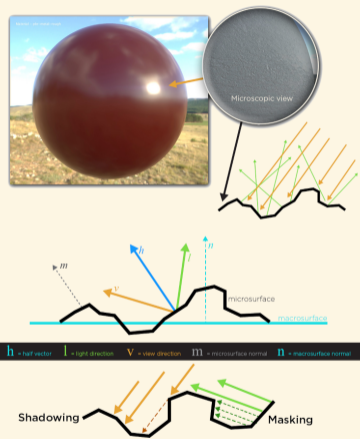
- Realistic Lighting & Shading
- Measured Materials
- Material Library
- Diffuse & Specular Reflection



Source: The PBR Guide

MICROFACET THEORY

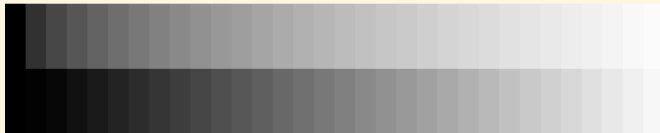
- Surface Irregularities
- Planar Details
- Shadowing & Masking
- Causes Diffusion



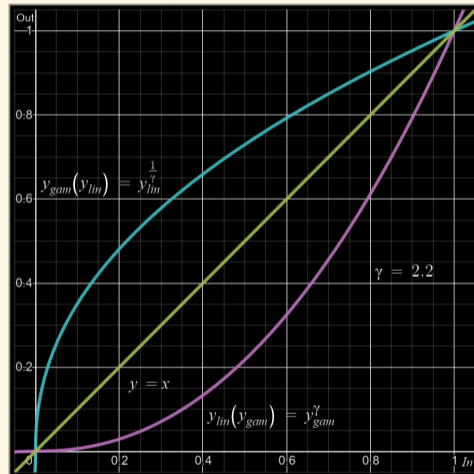
Source: The PBR Guide

COLOR

- Perceived Wavelengths
- Diffuse & Specular
- Psychophysics: $\Psi(I) = kI^\alpha$
- Non-Linear Response
- Gamma Correction [3]
 - ▶ Limited Precision
 - ▶ Encoding $v_{gam}(v_{lin}) = v_{lin}^{\frac{1}{\gamma}}$
 - ▶ Decoding $v_{lin}(v_{gam}) = v_{gam}^\gamma$
- Standard: sRGB

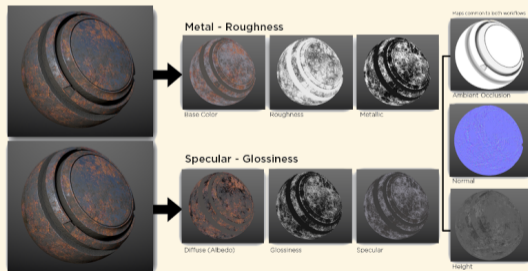


Source: John Novak : Gamma [3]

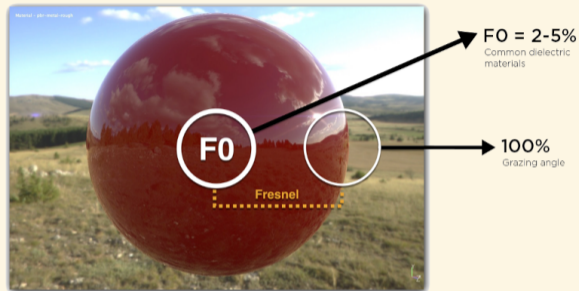


MATERIAL PROPERTIES

- Physical Plausibility
- BRDF and GGX [4]
- Fresnel Effect
- Conductors \times Dielectrics
- Metal-Rough vs Specular-Gloss



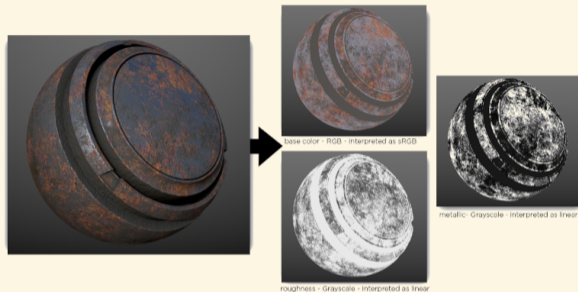
Source: The PBR Guide



Source: The PBR Guide

METALLIC ROUGHNESS WORKFLOW

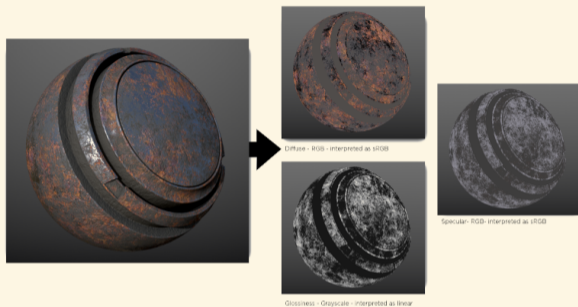
- Color, Metalness, Roughness
- Reference Chart
- Base Color
 - ▶ RGB Map, No AO
 - ▶ Diffuse (Di) & Reflectance (Me)
 - ▶ Brighter for Metals
 - ▶ Interpolation Problems
- Metalness
 - ▶ Grayscale Mask + Full Scale
 - ▶ Metal \rightarrow 1.0
- Roughness
 - ▶ Grayscale Map, Full Scale
 - ▶ Rough \rightarrow 1.0
- Pros & Cons:
 - ▶ Automatic Fo
 - ▶ Less Memory
 - ▶ Edge Artifacts



Source: The PBR Guide

SPECULAR GLOSSINESS WORKFLOW

- Diffuse, Specular, Glossiness
- Reference Chart
- Diffuse
 - ▶ RGB Map, No AO
 - ▶ Albedo, No Reflectance
 - ▶ Metal \rightarrow 0.0
- Specular
 - ▶ RGB Map, No AO
 - ▶ Fo (Di), Reflectance (Me)
 - ▶ Reflective \rightarrow (1.0, 1.0, 1.0)
- Glossiness
 - ▶ Grayscale Map
 - ▶ Rough \rightarrow 0.0
- Pros & Cons:
 - ▶ Weaker Edge Artifact
 - ▶ Explicit FO & Memory
 - ▶ Energy Conservation

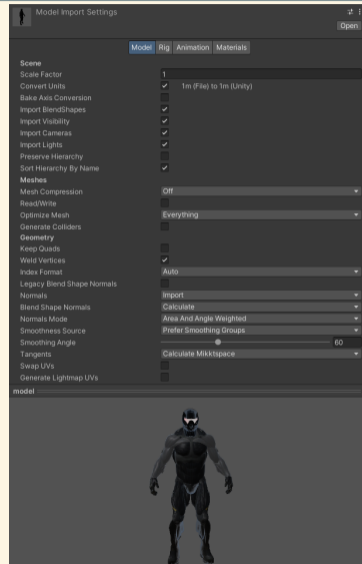


Source: The PBR Guide

UNITY ASSET MANAGEMENT

ASSETS IN UNITY

- Wide Format Support
 - ▶ Models, Images, Materials
 - ▶ Audio Files & Tracks
 - ▶ Code, Shaders
 - ▶ Scenes
- Importers: Builtin × Custom
- Database Metadata
- Implicit & Explicit Import
- Artist Workflow
- Programming Workflow
- AssetBundles & SubScenes



ADDITIONAL RESOURCES

- [Blog] Jarrod Hasenjager: Metal Material Study
- [Assets] Quixel Megascans Library
- [Book] The PBR Guide
- [Reference] Marmoset ToolBag PBR
- [John Novak] What every Coder Should Know About Gamma



Source: Unity: MegaCity Demo

Thanks For
Your Attention!

Gothic 1

REFERENCES I

- [1] DAVID CLYDE. ***THE DATA BUILDING PIPELINE OF OVERWATCH***. <https://www.gdcvault.com/play/1024019/The-Data-Building-Pipeline-of-2017>.
- [2] JASON GREGORY. ***GAME ENGINE ARCHITECTURE, SECOND EDITION***. 3rd. USA: A. K. Peters, Ltd., CRC Press, 2018. ISBN: 1351974288.
- [3] JOHN NOVAK. ***WHAT EVERY CODER SHOULD KNOW ABOUT GAMMA***. <http://blog.johannovak.net/2016/09/21/what-every-coder-should-know-about-gamma/>. 2016.
- [4] BRUCE WALTER ET AL. **“MICROFACET MODELS FOR REFRACTION THROUGH ROUGH SURFACES.”**. In: *Rendering techniques 2007* (2007), 18th.