* **Remember:** Pose-to-Pose Animation Pose to pose animation is planned out and charted. Key drawings are done at intervals throughout the scene.
* **Hierarchical grouping of objects**: the Scene Graph -> Scene Graph — a tree-like data structure which describes each object's position, orientation, 3D model, etc. Scene graphs are versatile, and used in 3D tools and game engines. There's basically no 3D scene you can't describe with a scene graph.
* **Types of Character Animation** • Key-frame animation – Pose based specification by animator • Motion capture – Recording motion during mocap session • Procedural / simulation – Automatically generated via rules and equations • Combinations – e.g. mocap + simulation for realistic motion
* **Key-framing (manual**) From Learning Maya 2.0 • Requires a highly skilled user • Poorly suited for interactive applications • High quality / high expense • Limited applicability of animations
* **Motion Capture (recorded**) • Is digitally recording specific 3D movements of a person / object) and translating them into computer-animated images. It uses markers, sensors, many cameras. • Time-consuming setup and clean-up • Reasonable quality / reasonable price (Pixar does not use cause of this)
* **Procedural Animation** - Simulation • Generates motion of objects using numerical simulation methods methods( rules, physics, equations) methods. Physics = gravity, forces, springs. • Often used in hair, cloths, 2ndary objects (but sometimes full characters) • Also called physically based animation.
* **Character Animation** -> Two Components: – Skeleton motion (rigging , bones animayion ) – Skin movement (skinning, assigning weights)
* **Modeling Hierarchy** - topology of a skeleton is an open directed graph, or tree (also called a hierarchy). One joint is selected as the root and the other joints are connected up in hierarchical fashion.
* **Bones vs Joints ->** Joints are the building blocks of skeletons and their points of articulation. -> Joints let you transform a skeleton when posing and animating a bound model. -> Bones have length and are visual cues that illustrate the relationships between joints.
* **Degrees of Freedom Specifies** which of the joint’s local axes the joint can rotate about during posing. Click X, Y, or Z. The default setting allows the joint to rotate about all three of its local axes during IK posing.
* **Morphed based Animation** as called point set interpolation or blend shapes Hybrid of per-vertex mesh deformation, morph target and rigid hierarchical system
* **Skeleton Animation (skinned mesh)** -> •Each vertex is linked to one or more bones. •Updating skeleton pose provides updating mesh pose. 1. Build skeleton system (Rigging) 2. Bind object to skeleton (Skinning) 3. Animating (FK & IK)
* **Skining** - The process of binding deformable mesh to a skeleton -> Colored triangles are attached to 1 bone -> Part of the skinning process is called weights or weight painting: Weighting or Painting the weights – is assigning the proper influence that each joint has on your skinned mesh and is a vital part of the skinning process
* **Forward Kinematics**: • In animation, a method for posing a joint chain by rotating each joint individually. It is the default method in hierarchical Key frame animation of a rigged character (does not requite extra processing with IK solvers).
* **Inverse Kinematics**: -> a method of posing a joint chain by specifying the location in space you want the joint chain to reach. - Inverse kinematics (IK) is a method of animating that reverses the direction of the chain manipulation. Rather than work from the root of the tree, it works from the leaves. – IK requires IK handles and IK solvers, special tools for posing and animating
* **Morph-based Animation (point set interpolation)** -> “Blendshapes” -> Interpolates to another set of vertex positions "Blendshapes" in Maya. You are blending between shapes. -> Requires fixed topology ● same vertex count ● same vertex IDs
* **Visual Effects** Visual effects is the term used to describe any imagery created, altered, or enhanced for a film or other moving media that cannot be accomplished during live-action shooting.
* **Kinds of Particle** Systems / Dynamics . • Particle system is defined by a collection of particles that evolves over time. • Particles • Contain a set of points in global space with attributes • Rigid bodies • Geometry with fixed topology and points • Soft bodies • Geometry with points updated by a particle object
* **Basic Model of Particle Systems** 1) Born: New particles are generated 2) Live: Each new particle is assigned its individual attributes. 3) Die: Any particles that have existed past their prescribed lifetime are extinguished. 4) Conitinue: The remaining particles are moved and transformed according to their dynamic attributes. -> • Emitter is the source of the particles • location in 3D space determines origin
* **The Blobby Surface** render type displays particles as metaballs. Metaballs are spheres that blend together to form surfaces.
* **Flocking** • Also called Schooling or swarming or herding • Craig Reynolds • Three simple rules • Separation, Alignment, Cohesion •a realistic form of group behavior similar to flocks of birds, schools of fish, or swarms of bees.
* **Emergent behavior** • Behaviors that aren’t explicitly programmed into individual agent rules
* **Three Rules (Flocking**) 1. Separation: steer to avoid crowding local flockmates 2. Alignment: steer toward the average heading of local flockmates 3. Cohesion: steer to move toward the average position of local flockmates
* **Finite-State Machine** or **FSM**? A finite-state machine, or FSM for short, is a model of computation based on a hypothetical machine made of one or more states. Only a single state can be active at the same time, so the machine must transition from one state to another in order to perform different actions.
* **3 studies** – presenting today at IVA Intelligent virtual agents Male, female, androgenious
* **Walk cycle 4 key poses** - repeat for each side :Contact, Down, Passing, Up
* **Motion Capture (recorded**) • Use Markers/sensors placed on subject • Time-consuming clean-up • A mocap suit can stretch by up to 10%
* **Rotoscoping Rotoscoping** is an animation technique that animators use to trace over motion picture footage, frame by frame, to produce realistic action.
* **Motion Capture vs. Keyframing**-> mocap : Every Frame, very dense, hard to edit; keyframing: only key frames, sparse and editable, can stretch & edit
* **Retargeting** -> you can retarget animation data from the joints of a skeleton (source) to the joints of another skeleton (target)
* **While the terms are interchangeable**, ’motion capture’ implies recording only the movement of head and body but no facial or finger data, whereas ’performance capture’ includes detailed capture of hands, fingers and face. Full performance capture further includes emotion and voice.
* **Mocap issues** -> Issues: data inaccuracy, expense € Optical: $90,000 – $210,000 high res cameras, occlusion, no outdoors, more and more markers, cameras € Magnetic: $100,000 – $120,000 heavier sensors, body wires, 10x accurate < optical, smaller workspaces, sensitive to metal, EMI, flooring € Mechanical: $20,000 – $30,000 US limited movement, heavy and cumbersome, must know rigid body at design time
* **Limitations of MOCAP** € Only realistic motion captured € WYSIWYG (what you see is what you get) – Can’t add more expression – Continually need to recapture motion
* **main problem with motion** capture associated with characters has to do with mass distribution, weight, and exaggeration.
* **Three Layers of Immersive Design** -> virtual reality means designing three layers of sensory stimuli: • The Environment • Directional Cues, • User Initiated Events and Feedback
* **A close-up of a list

  Description automatically generatedUser testing** • Place illusion • Plausibility illusion • Embodiment illusion
* **Audio:** What Sound can do in VR? • Aids immersion • Deepen presence • Act as navigation guidance –direct user attention
* **Spatial audio?** Track head movements and maintain relative sound position to the user
* **Definition of a Great Game** : great game is a series of interesting and meaningful choices made by the player in pursuit of a (clear and compelling?) goal
* **Interesting and Meaningful Choices** • Meaningful choices are perceived by the player as having significant consequences • Although they may not have actual consequences
* **Many games are chains of convexities** -> convexity of choices • Starts with a single choice, widens to many choices, returns to a single choice
* **Gameplay Trumps Story** • If you have a conflict between gameplay or story, first look for a compromise that favors both • Failing that, make sure that the gameplay is good at expense of story
* **1. Spatial Reasoning (Physical**) : Reasoning about objects in 3D space and how they might interact (includes your own body, hand-eye coordination).
* **2. Pattern Recognition (Mental**) : Recognizing patterns in organized sets of data, remembering chains of linked events that are significant.
* **3. Social** : Practicing interpersonal communication skills, competing/cooperating with others or modeling dynamics of social situations.
* **What is a game?** • Structure/dynamics we find in (some) games: • Objectives, goals • Outcomes, (specifically variable results) • Uncertainty • Rules and Structure • Stories
* **3 components** improve a players perception of a game: 1. Multiple clear achievable goals. 2. The illusion of choice. 3. Clear punishments and rewards.
* **Methodologies: Code and Fix :** • Little or no planning
* **Methodologies: Waterfall:** • Very well-defined steps in development • Lots of planning ahead of time • Doesn't react well to changes
* **Methodologies: Iterative:** • Multiple development cycles during a single project • Allows for planning but also for changes
* **Methodologies: Agile** • Very short iterations ( 2-3 weeks) • Iterate based on feedback of what was learned so far