

Start here



Draugr Characters | Version 1.0 | Buyer guide revision 20

What you receive

The Collection contains Juggernaut, Gravebinder and Runecaller. Individual products contain only the named character. Each includes an editable full-skeleton FK Control Rig, a skeletal mesh, custom skeleton, four mesh LOD levels, PBR textures, editable animated essence materials, a Physics Asset, an overview map with included animation demonstrations and portable geometry FBX source files.

Retargeted Epic Third Person template example animations are included, with a labeled overview station for every included clip. The purchased RPG animations shown in the promotional videos are a different set and are not included. The animated eye/body essence is an included material effect.

Weapons, animation FBX source files, gameplay controllers, AI, combat/spell systems and cloth simulation are not included.

First successful test

1. Extract the complete project folder. Open its .uproject file in Unreal Engine 5.8. Allow shader preparation to finish.
2. The startup overview map contains the supplied character or characters. Press Play to start all labeled animation stations. Each station loops one included clip. Find it under Content > BISDraugrCollection > Maps. Collection uses L_Overview; individual products use L_<Character>_Overview.
3. Find the skeletal mesh under BISDraugrCollection/Characters. Bruiser is the internal Juggernaut folder, Healer is Gravebinder and Mage is Runecaller. Double-click SK_Draugr_<Role> to inspect the mesh and its four LOD levels.
4. Check the material slots and open the assigned essence material instance. Juggernaut uses deep red, Gravebinder green and Runecaller blue. Enable real-time viewport updates to see the material motion. Press Play in the overview map to watch every included clip at its labeled station. Open individual animation assets to inspect a specific motion.

What the demonstration means

Display footage can show these character models moving with separately licensed animation content. It demonstrates appearance under those motions; it is not a promise that the motion, weapons, scenery or gameplay systems are included. Refer to the included-content list above.

Install and migrate

Move the content into your project

In Unreal, right-click the BISDraugrCollection content folder and choose Asset Actions > Migrate. Review the dependency list and select your project Content folder. Keep the pack folder intact. Do not relocate individual .uasset files in Explorer.

Environment and dependencies

Unreal Engine project format: 5.8; validated in Unreal Engine 5.8.3 on Windows with DX12/SM6. Earlier versions and other platforms are not verified. FBX geometry and PBR textures can be imported into compatible tools, but Unreal materials, animated essence, Control Rigs and Physics Assets do not transfer through FBX. Other-engine integration is not tested.

No purchased animation pack or private studio preview plugin is required to open the supplied content. Third-party animations used in display footage are not supplied or required. Buyers provide their own animation content and retain responsibility for its licensing.

Carry over the physics settings

After migration, apply the documented physics substepping settings in your destination project. Asset migration does not copy Config/DefaultEngine.ini.

Pose and reset the rig

Pose with the included Control Rig

1. Enable Unreal Engine's built-in Control Rig plugin in the destination project before migrating. The supplied project already enables it. No external paid plugin is needed.
2. Open BISDraugrCollection/ControlRigs/CR_Juggernaut, CR_Gravebinder or CR_Runecaller, matching your purchased character. The rig preview mesh is assigned. Choose Forwards Solve in the rig editor.
3. Select a control in the Rig Hierarchy. Each bone has a matching <BoneName>_CTRL control. Rotate a shoulder or finger control to pose that joint; translate the root control to move the entire hierarchy. Parent controls carry their children.
4. Undo temporary pose adjustments, or reset the edited controls current/local values to translation 0, rotation 0 and scale 1. Leave Initial Transform and Offset Transform unchanged; they define the resting setup. Reset every edited control to restore the complete reference pose.
5. To author motion, create your own Level Sequence, add a Skeletal Mesh Actor using the matching mesh, and add the matching Control Rig track to its binding. Key the controls in Sequencer. The sequence you create belongs to your project. The separate Animations folder supplies 26 retargeted Epic example clips per character; these are distinct from motions you author with the Control Rig.

Control layout and solve directions

The rig contains 119 transform controls covering the imported container/root, body, limbs, hands, fingers and remaining skeleton joints. Names preserve the supplied skeleton naming for traceability. Left controls are blue, right controls red, and center controls amber where bone names identify side.

Forwards Solve reads the controls and applies their global transforms to the matching bones. Backwards Solve reads the current bone pose and sets the controls to that pose, supporting a buyer-owned animation baking workflow. Use the correct solve direction; changing a bone manually during Forwards Solve will be overwritten by its control.

This is a complete FK posing rig. It does not include limb IK, foot locking, automatic ground contact, facial blendshapes, a locomotion state machine or weapon-fitting logic. Buyers must author and check those behaviors when their project needs them.

Keep the Control Rigs folder with the character assets when migrating. The rig preview references its character mesh and depends on its skeleton. No purchased RPG animations or private preview controller are dependencies.

Find a Control Rig

Paste the relevant object path into Unreal asset search or use the Content Browser folder path above.

/Game/BISDraugrCollection/ControlRigs/CR_Juggernaut

/Game/BISDraugrCollection/ControlRigs/CR_Gravebinder

/Game/BISDraugrCollection/ControlRigs/CR_Runecaller

Exact skeleton names and parents

Read the complete custom skeleton inventory

Skeleton inventory

Custom skeleton identity

Each character has 119 bones and its own Skeleton asset. These are custom humanoid skeletons. Shared Epic-style bone names do not make their reference transforms or hierarchy interchangeable with Manny, Quinn or another Epic skeleton. Retarget animations and validate deformation in your project.

The top grouping bone is RIG_Juggernaut, RIG_Gravebinder or RIG_Runecaller. Beneath it, the names and parent relationships in the table are shared. No skeleton bones or reference transforms were changed in this correction.

The included Control Rig exposes every bone through its matching _CTRL control. It supplies FK posing and Forwards/Backwards Solve. IK-named helper bones are present; limb IK, foot locking and retargeter assets are not included. Retargeted Epic template example animations are included. See Included animations for the exact list.

Custom auxiliary bones

The following retained source-rig bones provide pelvis, twist/share, torso, face and toe hierarchy. Their names are stable skeleton identifiers, not separate included gameplay features. Their presence does not promise facial animation or independent cloth simulation.

cc_base_pelvis, cc_base_l_forearmtwist01, cc_base_l_forearmtwist02, cc_base_l_elbowsharebone, cc_base_l_upperarmtwist01, cc_base_l_upperarmtwist02, cc_base_facialbone, cc_base_jawroot, cc_base_tongue01, cc_base_tongue02, cc_base_tongue03, cc_base_teeth02, cc_base_r_eye, cc_base_l_eye, cc_base_upperjaw, cc_base_teeth01, cc_base_r_upperarmtwist01, cc_base_r_upperarmtwist02, cc_base_r_forearmtwist01, cc_base_r_forearmtwist02, cc_base_r_elbowsharebone, cc_base_l_ribstwist, cc_base_l_breast, cc_base_r_ribstwist, cc_base_r_breast, cc_base_l_toebasesharebone, cc_base_l_pinkytoe1, cc_base_l_ringtoe1, cc_base_l_midtoe1, cc_base_l_indextoe1, cc_base_l_bigtoe1, cc_base_l_calftwist01, cc_base_l_calftwist02, cc_base_l_kneesharebone, cc_base_l_thightwist01, cc_base_l_thightwist02, cc_base_r_thightwist01, cc_base_r_thightwist02, cc_base_r_pinkytoe1, cc_base_r_bigtoe1, cc_base_r_indextoe1, cc_base_r_midtoe1, cc_base_r_ringtoe1, cc_base_r_toebasesharebone, cc_base_r_kneesharebone, cc_base_r_calftwist01, cc_base_r_calftwist02

Complete hierarchy

Download the exact names and parents . The first row changes to the matching character grouping name.

Bone	Parent
RIG_Juggernaut	(none)
root	RIG_Juggernaut
pelvis	root
cc_base_pelvis	pelvis
spine_01	pelvis
spine_02	spine_01
spine_03	spine_02
spine_04	spine_03
spine_05	spine_04
clavicle_l	spine_05
upperarm_l	clavicle_l
lowerarm_l	upperarm_l
cc_base_l_forearmtwist01	lowerarm_l

Bone	Parent
cc_base_l_forearmtwist02	cc_base_l_forearmtwist01
cc_base_l_elbowsharebone	lowerarm_l
hand_l	lowerarm_l
middle_metacarpal_l	hand_l
middle_01_l	middle_metacarpal_l
middle_02_l	middle_01_l
middle_03_l	middle_02_l
index_metacarpal_l	hand_l
index_01_l	index_metacarpal_l
index_02_l	index_01_l
index_03_l	index_02_l
ring_metacarpal_l	hand_l
ring_01_l	ring_metacarpal_l
ring_02_l	ring_01_l
ring_03_l	ring_02_l
pinky_metacarpal_l	hand_l
pinky_01_l	pinky_metacarpal_l
pinky_02_l	pinky_01_l
pinky_03_l	pinky_02_l
thumb_01_l	hand_l
thumb_02_l	thumb_01_l
thumb_03_l	thumb_02_l
cc_base_l_upperarmtwist01	upperarm_l
cc_base_l_upperarmtwist02	cc_base_l_upperarmtwist01
neck_01	spine_05
neck_02	neck_01
head	neck_02
cc_base_facialbone	head
cc_base_jawroot	cc_base_facialbone
cc_base_tongue01	cc_base_jawroot
cc_base_tongue02	cc_base_tongue01
cc_base_tongue03	cc_base_tongue02
cc_base_teeth02	cc_base_jawroot
cc_base_r_eye	cc_base_facialbone
cc_base_l_eye	cc_base_facialbone
cc_base_upperjaw	cc_base_facialbone
cc_base_teeth01	cc_base_upperjaw

Bone	Parent
clavicle_r	spine_05
upperarm_r	clavicle_r
cc_base_r_upperarmtwist01	upperarm_r
cc_base_r_upperarmtwist02	cc_base_r_upperarmtwist01
lowerarm_r	upperarm_r
cc_base_r_forearmtwist01	lowerarm_r
cc_base_r_forearmtwist02	cc_base_r_forearmtwist01
cc_base_r_elbowsharebone	lowerarm_r
hand_r	lowerarm_r
middle_metacarpal_r	hand_r
middle_01_r	middle_metacarpal_r
middle_02_r	middle_01_r
middle_03_r	middle_02_r
ring_metacarpal_r	hand_r
ring_01_r	ring_metacarpal_r
ring_02_r	ring_01_r
ring_03_r	ring_02_r
thumb_01_r	hand_r
thumb_02_r	thumb_01_r
thumb_03_r	thumb_02_r
index_metacarpal_r	hand_r
index_01_r	index_metacarpal_r
index_02_r	index_01_r
index_03_r	index_02_r
pinky_metacarpal_r	hand_r
pinky_01_r	pinky_metacarpal_r
pinky_02_r	pinky_01_r
pinky_03_r	pinky_02_r
cc_base_l_ribstwist	spine_04
cc_base_l_breast	cc_base_l_ribstwist
cc_base_r_ribstwist	spine_04
cc_base_r_breast	cc_base_r_ribstwist
thigh_l	pelvis
calf_l	thigh_l
foot_l	calf_l
cc_base_l_toebasesharebone	foot_l
ball_l	foot_l

Bone	Parent
cc_base_l_pinkytoe1	ball_l
cc_base_l_ringtoe1	ball_l
cc_base_l_midtoe1	ball_l
cc_base_l_indextoe1	ball_l
cc_base_l_bigtoe1	ball_l
cc_base_l_calftwist01	calf_l
cc_base_l_calftwist02	cc_base_l_calftwist01
cc_base_l_kneesharebone	calf_l
cc_base_l_thightwist01	thigh_l
cc_base_l_thightwist02	cc_base_l_thightwist01
thigh_r	pelvis
cc_base_r_thightwist01	thigh_r
cc_base_r_thightwist02	cc_base_r_thightwist01
calf_r	thigh_r
foot_r	calf_r
ball_r	foot_r
cc_base_r_pinkytoe1	ball_r
cc_base_r_bigtoe1	ball_r
cc_base_r_indextoe1	ball_r
cc_base_r_midtoe1	ball_r
cc_base_r_ringtoe1	ball_r
cc_base_r_toebasesharebone	foot_r
cc_base_r_kneesharebone	calf_r
cc_base_r_calftwist01	calf_r
cc_base_r_calftwist02	cc_base_r_calftwist01
ik_foot_root	root
ik_foot_r	ik_foot_root
ik_foot_l	ik_foot_root
ik_hand_root	root
ik_hand_gun	ik_hand_root
ik_hand_r	ik_hand_gun
ik_hand_l	ik_hand_gun

Included animations and retargeting

Review every included animation

1. Open the supplied project in Unreal Engine 5.8. The startup overview map contains a labeled station for each included clip: 26 in a single-character package and 78 in the Collection.
2. Press Play and click the viewport to take control of the inspection camera. Each station loops its assigned clip independently. Use WASD to fly, move the mouse to look, and Q/E to descend or ascend. Press Escape to exit Play.
3. Read the station label to identify the character and exact animation. Walk and jog clips demonstrate motion in place; a stationary actor location does not mean its animation has stopped. For jumping and attack examples, let the full short loop finish.
4. For close inspection, open the matching A_ animation in Content/BISDraugrCollection/Animations. Use its timeline to pause or scrub. Each character uses its own custom Skeleton and its own retargeted examples.
5. The AC_ composite is also retained as a combined inspection playlist. The overview stations use the individual clips so every example can be reviewed without waiting for the playlist.

The inspection camera and stations are demonstrations. These assets do not include a character locomotion controller, combat system, weapons, or spell effects.

Origin and included scope

These motions are retargeted derivatives of Epic Games' installed Unreal Engine 5.8 Third Person template examples. They are included for demonstration and example use, not sold as an original Black Iron Spine animation library. See the Unreal Engine EULA for Epic's terms.

The characters retain their custom skeletons. Retargeted examples do not make these skeletons directly interchangeable with Manny or Quinn. The included clips do not supply weapons, AI, combat abilities, caster spells, input, possession switching or a Character Blueprint. The B/N controls and camera zoom shown in the private promotional preview are not included.

Promotional videos show separately purchased RPG animations. Those specific motions are not included. The included Epic template examples are listed below.

Juggernaut animation inventory

26 animation sequences. Durations below are seconds.

Asset	Duration
A_Juggernaut_MM_Attack_01	1.00
A_Juggernaut_MM_Attack_02	1.00
A_Juggernaut_MM_Attack_03	1.67
A_Juggernaut_MM_ChargedAttack	1.83
A_Juggernaut_MF_Unarmed_Jog_Bwd	1.50
A_Juggernaut_MF_Unarmed_Jog_Bwd_Left	1.50
A_Juggernaut_MF_Unarmed_Jog_Bwd_Right	1.50
A_Juggernaut_MF_Unarmed_Jog_Fwd	1.77
A_Juggernaut_MF_Unarmed_Jog_Fwd_Left	1.77
A_Juggernaut_MF_Unarmed_Jog_Fwd_Right	1.77
A_Juggernaut_MF_Unarmed_Jog_Left	1.53

Asset	Duration
A_Juggernaut_MF_Unarmed_Jog_Right	1.73
A_Juggernaut_MM_Dash	0.97
A_Juggernaut_MM_Fall_Loop	3.00
A_Juggernaut_MM_Jump	0.87
A_Juggernaut_MM_Land	0.87
A_Juggernaut_MM_WallJump	0.47
A_Juggernaut_MM_Idle	7.57
A_Juggernaut_MF_Unarmed_Walk_Bwd	2.13
A_Juggernaut_MF_Unarmed_Walk_Bwd_Left	2.13
A_Juggernaut_MF_Unarmed_Walk_Bwd_Right	2.13
A_Juggernaut_MF_Unarmed_Walk_Fwd	1.50
A_Juggernaut_MF_Unarmed_Walk_Fwd_Left	1.50
A_Juggernaut_MF_Unarmed_Walk_Fwd_Right	1.50
A_Juggernaut_MF_Unarmed_Walk_Left	1.53
A_Juggernaut_MF_Unarmed_Walk_Right	1.53

Gravebinder animation inventory

26 animation sequences. Durations below are seconds.

Asset	Duration
A_Gravebinder_MM_Attack_01	1.00
A_Gravebinder_MM_Attack_02	1.00
A_Gravebinder_MM_Attack_03	1.67
A_Gravebinder_MM_ChargedAttack	1.83
A_Gravebinder_MF_Unarmed_Jog_Bwd	1.50
A_Gravebinder_MF_Unarmed_Jog_Bwd_Left	1.50
A_Gravebinder_MF_Unarmed_Jog_Bwd_Right	1.50
A_Gravebinder_MF_Unarmed_Jog_Fwd	1.77
A_Gravebinder_MF_Unarmed_Jog_Fwd_Left	1.77
A_Gravebinder_MF_Unarmed_Jog_Fwd_Right	1.77
A_Gravebinder_MF_Unarmed_Jog_Left	1.53
A_Gravebinder_MF_Unarmed_Jog_Right	1.73
A_Gravebinder_MM_Dash	0.97
A_Gravebinder_MM_Fall_Loop	3.00
A_Gravebinder_MM_Jump	0.87
A_Gravebinder_MM_Land	0.87
A_Gravebinder_MM_WallJump	0.47
A_Gravebinder_MM_Idle	7.57

Asset	Duration
A_Gravebinder_MF_Unarmed_Walk_Bwd	2.13
A_Gravebinder_MF_Unarmed_Walk_Bwd_Left	2.13
A_Gravebinder_MF_Unarmed_Walk_Bwd_Right	2.13
A_Gravebinder_MF_Unarmed_Walk_Fwd	1.50
A_Gravebinder_MF_Unarmed_Walk_Fwd_Left	1.50
A_Gravebinder_MF_Unarmed_Walk_Fwd_Right	1.50
A_Gravebinder_MF_Unarmed_Walk_Left	1.53
A_Gravebinder_MF_Unarmed_Walk_Right	1.53

Runecaller animation inventory

26 animation sequences. Durations below are seconds.

Asset	Duration
A_Runecaller_MM_Attack_01	1.00
A_Runecaller_MM_Attack_02	1.00
A_Runecaller_MM_Attack_03	1.67
A_Runecaller_MM_ChargedAttack	1.83
A_Runecaller_MF_Unarmed_Jog_Bwd	1.50
A_Runecaller_MF_Unarmed_Jog_Bwd_Left	1.50
A_Runecaller_MF_Unarmed_Jog_Bwd_Right	1.50
A_Runecaller_MF_Unarmed_Jog_Fwd	1.77
A_Runecaller_MF_Unarmed_Jog_Fwd_Left	1.77
A_Runecaller_MF_Unarmed_Jog_Fwd_Right	1.77
A_Runecaller_MF_Unarmed_Jog_Left	1.53
A_Runecaller_MF_Unarmed_Jog_Right	1.73
A_Runecaller_MM_Dash	0.97
A_Runecaller_MM_Fall_Loop	3.00
A_Runecaller_MM_Jump	0.87
A_Runecaller_MM_Land	0.87
A_Runecaller_MM_WallJump	0.47
A_Runecaller_MM_Idle	7.57
A_Runecaller_MF_Unarmed_Walk_Bwd	2.13
A_Runecaller_MF_Unarmed_Walk_Bwd_Left	2.13
A_Runecaller_MF_Unarmed_Walk_Bwd_Right	2.13
A_Runecaller_MF_Unarmed_Walk_Fwd	1.50
A_Runecaller_MF_Unarmed_Walk_Fwd_Left	1.50
A_Runecaller_MF_Unarmed_Walk_Fwd_Right	1.50
A_Runecaller_MF_Unarmed_Walk_Left	1.53

Asset	Duration
A_Runecaller_MF_Unarmed_Walk_Right	1.53

Use your own animations

These are custom skeletons with 119 native bones, including the imported container/root. Direct compatibility with the Epic mannequin skeleton is not claimed. Each character has its own supplied skeleton.

Assign the supplied skeletal mesh to your Character Blueprint Mesh component. Set its relative transform to fit your capsule and facing direction. Use an Animation Blueprint built for the matching skeleton and animation clips that you own or are licensed to use.

Animations made for a different skeleton require retargeting before use. Test foot contacts, shoulder/hand deformation and weapon grips with your chosen motions. The included example clips are already retargeted to the corresponding Draugr skeleton. A gameplay Animation Blueprint and an IK Retargeter are not supplied.

Optional test animations from Mixamo

Mixamo: <https://www.mixamo.com/> . Adobe describes Mixamo as free with an Adobe ID, without a Creative Cloud subscription, and permits animations in personal and commercial projects under its terms. Download animations from Adobe yourself. None are bundled with these characters.

For a first test, choose a Mixamo library humanoid and download one simple idle and walk. Keep the source character and animation imports in a separate test folder using their source skeleton. Do not assign the Draugr skeleton to an unrelated FBX merely to force the import to succeed.

Use Unreal IK Rig and IK Retargeter to map source and destination body chains and align the retarget poses. These Draugr have custom 119-bone skeletons; the included FK Control Rig is a posing tool, not an IK Retargeter. Set up and validate retargeting for each purchased character.

Preview one clip first, correct facing direction, scale and feet, then export retargeted animations for the destination skeleton. Check shoulders, hands and fingers before retargeting a large set. Root motion availability depends on the source clip and skeleton; do not assume downloaded motions include a suitable root track.

This is an optional workflow outline, not a tested Mixamo compatibility preset or an included integration. Keep the supplied skeleton unchanged while testing. We do not recommend re-rigging the original character just to try an animation.

Adobe FAQ: <https://helpx.adobe.com/creative-cloud/faq/mixamo-faq.html> . Unreal retargeting guide: <https://dev.epicgames.com/documentation/en-us/unreal-engine/ik-rig-animation-retargeting-in-unreal-engine> . Consult the service terms applicable to your download.

Game integration

Using the characters in your game

Work in a backup or version-controlled copy of your game. Open the supplied project in Unreal Engine 5.8. Enable the built-in Control Rig plugin in your destination project if you plan to use the supplied rigs. Migrate Content/BISDraugrCollection with Asset Actions > Migrate into the destination Content directory so Unreal carries the dependencies with it.

All four products use the same BISDraugrCollection namespace. Singles contain only their named character. If you already imported a single and later import the collection, review replacement prompts and back up your customizations before allowing overlapping assets to be replaced. Keep game-specific Blueprints and duplicated material instances in your own game folder.

For a playable character, duplicate a working Character Blueprint in your game, preserving its input, capsule, camera and CharacterMovement setup. Replace its Mesh component skeletal mesh with the purchased character. The package does not supply those gameplay systems.

Fit the mesh inside the capsule: place the soles at the capsule bottom and align the character face with the Blueprint forward direction. Inspect front and side views. Do not assume the mannequin mesh rotation or vertical offset is correct for these characters. Test forward, sideways and backward movement before adding combat.

Set the Mesh animation mode to Use Animation Blueprint and choose an Animation Blueprint for this character skeleton. A mannequin Animation Blueprint cannot simply be assigned to a different skeleton. Retarget your animation assets and adapt the Blueprint, or build one using your compatible clips.

A basic locomotion setup commonly uses horizontal velocity length for speed and CharacterMovement Is Falling for airborne state. Use your own idle/walk/run blend, jump or fall states, and landing transition. Choose in-place or root-motion movement deliberately; do not drive the character twice. These are integration suggestions, not supplied or verified gameplay features.

Check feet at idle and through a full walk/run cycle. A constant height error can be mesh/capsule placement; feet that lift or slide during only part of a clip call for animation, retarget-pose or stride review. Do not offset the entire mesh to conceal a bad motion.

Weapons, performance and shipping

Juggernaut is visually suited to a two-handed bruiser, Gravebinder to a staff-bearing healer/support caster, and Runecaller to a wand with a book or orb. These are intended archetypes, not supplied weapon meshes, attacks or abilities. Buyers can assign different roles.

When adding weapons, author sockets and fit the grip to your chosen weapon and motions. A two-handed weapon may need off-hand IK or additional animation editing in your game. The supplied FK rig does not automatically attach weapons or maintain grips.

Inspect all four LODs at actual gameplay camera distances. LOD0 is high detail. Profile your intended number of characters and adjust LOD thresholds and texture budgets for your hardware. No mobile or crowd-performance claim is made.

Use the supplied Physics Asset as a starting point for your collision and ragdoll behavior. Your game must manage capsule collision, simulation and recovery. Test your own floors and slopes; a ragdoll is not an authored death clip.

Before shipping your game, package and test your target platform, ensure required character assets are referenced for cooking, and confirm the licenses for every animation and weapon you added. Follow the product license supplied through your store purchase; this guide does not replace it.

Essence, LODs and physics

Customize the living essence

The eye, chest and damaged-face treatment uses animated material shading. Open the assigned instance under Materials/EssenceR3. Duplicate the instance before editing exposed color, intensity and swirl parameters. Keep its parent and referenced mask texture with it.

Inspect the result in your intended lighting and exposure. Bloom, emissive appearance and contrast depend on your project renderer settings. The Juggernaut face treatment is an intentional magical effect.

LODs and physics

The default ragdoll uses conservative joint limits and thigh collision shapes that cover the skinned skirt or robe. The lower body bends less freely to protect the garment from extreme folding. These settings affect physics simulation only; they do not restrict the included animations. Loosening limits or changing collision shapes requires renewed clipping tests.

There are four LOD levels per character. LOD0 is high detail; inspect lower LODs and choose screen-size transitions appropriate for your camera distance and character count. Profile in your game. No frame-rate or mobile-performance guarantee is made.

A Physics Asset is supplied. Your gameplay code owns ragdoll activation and collision. Stop character movement, choose an appropriate collision profile and enable skeletal-mesh physics when needed. Prevent the movement capsule from fighting the simulated body.

Test on your actual ground geometry. Check constraint stability, floor contact, clothing deformation and restoration if your game supports revival. A Physics Asset is not a death animation.

Texture folders

All Unreal textures are under BISDraugrCollection/Textures, with Juggernaut, Gravebinder and Runecaller subfolders. Each character folder contains its base color, normal, metallic, roughness and essence-mask textures. A single-character package contains only its matching subfolder. Unreal material references are updated with the move.

Physics settings when migrating

The supplied project enables physics substepping. Unreal's Migrate command copies assets and their dependencies, but does not copy project settings. Apply these settings in your destination project before testing ragdoll.

1. Open Edit > Project Settings > Engine > Physics and expand the advanced Framerate settings.
2. Enable Substepping. Set Max Substep Delta Time to 0.008333333 seconds and Max Substeps to 8.
3. Use the supplied Physics Asset with its matching mesh. Keep mesh scale at 1.0 for the initial test and disable conflicting capsule collision while simulating the body.
4. Test on your project's floor collision and check the complete character, including armor and robes. Integration settings, larger impulses and different collision geometry can change the result.

The shipped project uses these values in Config/DefaultEngine.ini under [/Script/Engine.PhysicsSettings]:
bSubstepping=True, MaxSubstepDeltaTime=0.008333333, MaxSubsteps=8. Automatic fall and impulse checks were run at 60 and 30 frames per second with these settings. These checks do not guarantee behavior for every game or force.

Inspect ragdoll movement

1. Open the matching PA_Draugr_Bruiser, PA_Draugr_Healer, or PA_Draugr_Mage Physics Asset and start Simulate.
2. In Unreal Engine 5.8, hold Ctrl and drag with the right mouse button over a collision body to grab and move it. Ctrl plus left mouse button applies an impulse. Clicking without Ctrl does not perform the simulation grab.
3. Lift the torso gently, move it sideways, release it, and let it settle. Repeat from an arm and leg. Stop Simulate to restore the reference pose.
4. Check the body and armor for excessive folding, penetration, stretching, or persistent jitter. Include these interactions in your own game's collision and damage testing.

The garment is skinned geometry; cloth simulation is not included. Physics tuning and your game's movement or damage logic are separate from the included animation demonstrations.

Technical inventory

Technical inventory

Textures per character: 4096 x 4096 base color and normal maps, 2048 x 2048 metallic and roughness maps, plus the character essence mask. One skeletal mesh, one skeleton, one Physics Asset, one essence parent material and one material instance per character.

Four skinned geometry FBXs (LOD0 through LOD3) are under SourceArt/<Character>/Geometry. Source PBR textures are in its Textures subfolder. Read SourceArt/<Character>/READ-ME.txt before editing. These files contain no character animation takes.

Character	LOD0	LOD1	LOD2	LOD3
Juggernaut	188,174	94,078	47,034	23,510
Gravebinder	195,765	97,874	48,928	24,455
Runecaller	178,936	89,466	44,732	22,363

Exact skeleton names and parents

Read the complete custom skeleton inventory

Included demonstration animations

Each single-character product contains its character-specific Epic-derived example clips and one demonstration composite. The Collection contains all three sets. See the exact animation inventory and playback instructions . Geometry FBX sources remain animation-free; the included demonstrations are Unreal animation assets.

Edit source and restore

Editing source files and restoring content

Duplicate the project before editing geometry. Preserve bone names, hierarchy, bind pose and skin weights if you intend to retain compatibility with the supplied skeleton and Control Rig.

Test reimport of the matching character and LOD in the copied project. Check skeleton selection, material assignments, all four LODs, skinning, physics fit and essence masks. Changed topology or UVs may need new masks. Regenerating a skeleton can invalidate the rig; do not overwrite a working skeleton to force an import.

Restore a damaged asset from your backup or migrate a clean copy from the original download, preserving unrelated game changes. Keep customized material instances and gameplay Blueprints in your own game folder to make future updates easier.

The studio demonstration controls, including B possession switching, N animation cycling and camera zoom, are not included in the buyer project. Its overview map is for inspecting the supplied character assets.

Troubleshooting and FAQ

Control Rig review and troubleshooting

Open the matching CR asset under BISDraugrCollection/ControlRigs. Select Rig Hierarchy below the preview, rather than My Blueprint. Clear the search filter, then search for `_CTRL`. Select a control, not its similarly named bone. Use Forwards Solve for posing.

If no rotation gizmo appears, select a control and activate the Rotate tool in the viewport. Inspect the viewport control-visibility options and confirm the preview mesh is assigned. If a specific part has no control, record the asset and selected hierarchy item and send support a screenshot; do not delete or rebuild the skeleton.

To undo a temporary pose, undo the adjustments. To reset control values manually, use current/local translation 0, rotation 0 and scale 1. Leave Initial Transform and Offset Transform intact: they establish the reference position. Reset all edited controls to restore the whole pose.

FK means each joint is posed by rotating its control; moving the wrist does not automatically bend the elbow as an IK handle would. Do not diagnose that difference as a missing rig feature.

If the model stays in its reference pose during Play, confirm that a compatible animation or Animation Blueprint is actually assigned. Press Play in the overview scene to start the labeled Epic example animation stations, or open an included animation asset directly. The purchased RPG animations shown in the promotional videos are not included.

If essence appears still, enable real-time viewport updates. If it appears too bright or dark, check exposure, bloom and the assigned material instance. Preserve the essence parent and mask dependencies when migrating.

If materials or rigs are missing after copying files, migrate again from the intact project through Unreal. For a failed startup, retain Saved/Logs and note the last message and elapsed time before reporting the issue; startup percentages alone do not identify the cause.

Animation and demonstration FAQ

Animated essence means the eye/body material moves over time. Separately, the package includes 26 retargeted Epic template example clips per character; see the Included animations chapter for the exact inventory. Purchased RPG motions and class-specific spellcasting sets are not included.

Demonstration footage uses separately licensed animations. You do not need that particular RPG pack to open or pose the purchased characters. To animate them in a game, author your own clips or obtain suitable animations and retarget them.

Obtain external animation files directly from their provider and follow the license for your purchase or download. This character purchase grants no license to the third-party motions shown in demonstration footage.

Your store purchase license governs use of these characters. Third-party animation and weapon licenses apply separately. This guide adds no separate license restrictions or permissions.

Support: support.blackironspine@proton.me. Include the product name, Unreal version, relevant asset path, reproduction steps and error text. For deformation issues, include the source animation name, retarget setup and a screenshot.

Support

For help, email support.blackironspine@proton.me with the character product name, Unreal Engine version and a description of the issue. Include relevant error messages and the steps needed to reproduce it.

Character quick starts

Choose your purchased product

The Collection includes all three characters. A single-character purchase contains only that character. The paths below refer to the assets supplied in the matching product.

Collection

Open BISDraugrCollection.uproject in Unreal Engine 5.8. The startup map is /Game/BISDraugrCollection/Maps/L_Overview.

The collection contains all three characters. Read the Juggernaut, Gravebinder and Runecaller quick starts for exact asset paths, then Game-Integration-Guide.txt for game setup. Treat each character skeleton independently when retargeting. There are three meshes, three skeletons and three FK Control Rigs; retargeted Epic template example animations are included for each character.

Migrate the BISDraugrCollection folder through Unreal. Back up modifications before combining this collection with an existing single-character installation because the asset paths overlap.

[Download the plain-text quick start](#)

Juggernaut

1. Open BISDraugrCollection.uproject in Unreal Engine 5.8. The single-product overview is /Game/BISDraugrCollection/Maps/L_Juggernaut_Overview. The collection uses L_Overview.
2. Mesh: /Game/BISDraugrCollection/Characters/Bruiser/SK_Draugr_Bruiser
3. Skeleton: /Game/BISDraugrCollection/Characters/Bruiser/SK_Draugr_Bruiser_Skeleton
4. Physics Asset: /Game/BISDraugrCollection/Characters/Bruiser/PA_Draugr_Bruiser
5. Posing rig: /Game/BISDraugrCollection/ControlRigs/CR_Juggernaut. Open Rig Hierarchy, select a _CTRL control and rotate it in Forwards Solve.
6. Essence color: deep red. Inspect the mesh's assigned material instance before changing it.
7. Intended theme: two-handed bruiser; sword supplied by buyer. Archetype is descriptive; no combat or weapon system is supplied.
8. For your game, follow Game-Integration-Guide.txt. Use a matching-skeleton Animation Blueprint and your own or separately licensed, correctly retargeted animation clips.

Included: custom skeleton, skeletal mesh with four LODs, Physics Asset, PBR textures, animated essence, full-skeleton FK Control Rig with 119 controls, overview map, retargeted Epic template example animations and geometry FBX source.

Excluded: purchased RPG animations shown in promotional videos, weapons, locomotion/controller Blueprints, AI, spells and cloth simulation.

Optional animation source: <https://www.mixamo.com/> — obtain animations directly and retarget; no tested direct compatibility is claimed.

Support: support.blackironspine@proton.me

Download the plain-text quick start

Gravebinder

1. Open BISDraugrCollection.uproject in Unreal Engine 5.8. The single-product overview is /Game/BISDraugrCollection/Maps/L_Gravebinder_Overview. The collection uses L_Overview.
2. Mesh: /Game/BISDraugrCollection/Characters/Healer/SK_Draugr_Healer
3. Skeleton: /Game/BISDraugrCollection/Characters/Healer/SK_Draugr_Healer_Skeleton
4. Physics Asset: /Game/BISDraugrCollection/Characters/Healer/PA_Draugr_Healer
5. Posing rig: /Game/BISDraugrCollection/ControlRigs/CR_Gravebinder. Open Rig Hierarchy, select a _CTRL control and rotate it in Forwards Solve.
6. Essence color: green. Inspect the mesh's assigned material instance before changing it.
7. Intended theme: healer/support caster; staff supplied by buyer. Archetype is descriptive; no combat or weapon system is supplied.
8. For your game, follow Game-Integration-Guide.txt. Use a matching-skeleton Animation Blueprint and your own or separately licensed, correctly retargeted animation clips.

Included: custom skeleton, skeletal mesh with four LODs, Physics Asset, PBR textures, animated essence, full-skeleton FK Control Rig with 119 controls, overview map, retargeted Epic template example animations and geometry FBX source.

Excluded: purchased RPG animations shown in promotional videos, weapons, locomotion/controller Blueprints, AI, spells and cloth simulation.

Optional animation source: <https://www.mixamo.com/> — obtain animations directly and retarget; no tested direct compatibility is claimed.

Support: support.blackironspine@proton.me

Download the plain-text quick start

Runecaller

1. Open BISDraugrCollection.uproject in Unreal Engine 5.8. The single-product overview is /Game/BISDraugrCollection/Maps/L_Runecaller_Overview. The collection uses L_Overview.
2. Mesh: /Game/BISDraugrCollection/Characters/Mage/SK_Draugr_Mage
3. Skeleton: /Game/BISDraugrCollection/Characters/Mage/SK_Draugr_Mage_Skeleton
4. Physics Asset: /Game/BISDraugrCollection/Characters/Mage/PA_Draugr_Mage
5. Posing rig: /Game/BISDraugrCollection/ControlRigs/CR_Runecaller. Open Rig Hierarchy, select a _CTRL control and rotate it in Forwards Solve.
6. Essence color: blue. Inspect the mesh's assigned material instance before changing it.
7. Intended theme: offensive caster; wand and off-hand item supplied by buyer. Archetype is descriptive; no combat or weapon system is supplied.
8. For your game, follow Game-Integration-Guide.txt. Use a matching-skeleton Animation Blueprint and your own or separately licensed, correctly retargeted animation clips.

Included: custom skeleton, skeletal mesh with four LODs, Physics Asset, PBR textures, animated essence, full-skeleton FK Control Rig with 119 controls, overview map, retargeted Epic template example animations and geometry FBX source.

Excluded: purchased RPG animations shown in promotional videos, weapons, locomotion/controller Blueprints, AI, spells and cloth simulation.

Optional animation source: <https://www.mixamo.com/> — obtain animations directly and retarget; no tested direct compatibility is claimed.

Support: support.blackironspine@proton.me

Download the plain-text quick start