

Epic Boss System

Complete boss battle framework for Unreal Engine 5.6 — 5.7

Create multi-phase boss fights with 11 combat systems in minutes.

v3.0.0

UE 5.6 / 5.7

C++ + Blueprint

8000+ Lines

Win64

Features



Multi-Phase Combat

1–3 phases with independent HP, per-phase attack config, cutscene transitions, and healing/invulnerability windows.



5 Attack Types

Ranged projectiles, melee with knockback, explosive barrels, AOE ground attacks, and dash/charge — plus Mixed and Custom modes.



Shield & Vulnerability

Health/Phase/Timed shields with regeneration, invulnerable boss with stagger-to-break system, damage multiplier windows.



Enrage Timer

MMO-style berserk timer with gradual damage/speed scaling, configurable instant-kill wipe mechanic.



Boss Intro Sequence

4 styles (NamePlate, CameraFocus, FullCutscape), name+title display, disable player input, invulnerability during intro.



Attack Patterns

Sequence-based combos (3x ranged → dash → melee), weighted random selection, distance/cooldown requirements.



Loot Table

Weighted drops with rarity system, guaranteed items, phase-specific loot, physics scatter on spawn.



Aggro / Threat

Multiplayer tank/DPS mechanics: threat decay, damage-to-threat conversion, taunt ability, auto-target switching with hysteresis.



Arena System

Configurable boundaries (Box/Sphere/Cylinder), push-back force, boundary damage, auto close/open on fight start/defeat.



Boss HUD Widget

Full C++ UUserWidget: health bar, shield bar, phase indicators, enrage timer, vulnerability indicator, smooth interpolation, 11 BP customization events.



Minion Spawning

Per-phase minion config, initial + continuous spawning, max alive limit, custom spawn points, kill-on-phase-end option.



External Health

Integrate with YOUR health system — report health changes from GAS, damage components, or any custom system.

Installation

1. Copy `EpicBossSystem` folder to `YourProject/Plugins/`
2. Restart Unreal Editor
3. Go to **Edit** → **Plugins**
4. Search "Epic Boss" and enable it
5. Restart editor when prompted

Dependencies: Niagara (built-in), LevelSequence (built-in). Both are auto-enabled.

Beginner's Guide — Your First Boss Fight in 5 Minutes

Don't panic! This plugin has 200+ settings, but you only need to touch **5 things** to get a working boss fight. Everything else has smart defaults. Start simple, add complexity later.

Step 1 — Create Boss Blueprint

1. **Content Browser** → **Right Click** → **Blueprint Class**
2. Search for `BossCharacter` as parent class
3. Name it `BP_MyBoss`, open it
4. Add a **Skeletal Mesh** (any character mesh — even the UE mannequin works for testing)
5. That's it — `BossComponent` is already attached automatically

Step 2 — Set 5 Essential Properties

Select the **Boss Component** in your BP. In Details Panel, you only need these:

CATEGORY	PROPERTY	WHAT TO SET	WHY
Boss Info	<code>Boss Name</code>	"My First Boss"	Shows on HUD health bar
Boss Phases	<code>Number of Phases</code>	1 (start simple!)	More phases = harder to debug at first
Boss Phases	<code>Phase Configs</code> → <code>[0]</code> → <code>Health Points</code>	500	Boss HP for phase 1
Boss Combat	<code>Projectile Class</code>	Leave empty for now	Boss works without attacks — add later
Boss Arena	<code>Arena Radius</code>	2000	Fight area size (0 = no arena)

Everything else uses good defaults. Don't touch the other 200 settings yet.

Step 3 — Place in Level and Test

1. Drag `BP_MyBoss` into your level
2. Press **Play**
3. Walk near the boss — the fight starts automatically (boss has trigger by default)
4. Boss HP bar appears on screen

Step 4 — Deal Damage to Boss

The boss needs to receive damage from your weapon. Two options:

If your weapon uses UE's damage system (`Apply Damage` node), the boss receives it automatically.

If you use a custom system, in your weapon hit logic call:

```
// Get BossComponent from hit actor
BossComponent→TakeBossDamage(50.0f);
```

Step 5 — Done! Now Expand

You have a working boss. Now add features one at a time:

🔥 "I WANT THE BOSS TO ATTACK"

1. Create Blueprint → parent: `ExplosiveBarrelProjectile` → name: `BP_BossProjectile`
2. Add a mesh to it (sphere, barrel, any projectile shape)
3. Back in Boss Component → **Projectile Class** = `BP_BossProjectile`
4. Boss now shoots at you automatically

🗡️ "I WANT MELEE ATTACKS"

In Boss Component → **Phase Configs** → **[0]** → **Attack Type** = `Melee`. Configure **Melee Damage** (50), **Melee Range** (200), **Melee Cooldown** (2s). Boss will punch you when close.

🔄 "I WANT MULTIPLE PHASES"

Set **Number of Phases** = 3. Expand **Phase Configs** → each phase has its own HP, attack type, speed, and damage multiplier. Phase 1 could be ranged (easy), Phase 2 mixed (medium), Phase 3 melee (intense).

🎬 "I WANT A CINEMATIC INTRO"

In Boss Component → **Boss|Intro** category:

- `Intro Style` = `CameraFocus` (camera zooms to boss)
- `Boss Title` = "Lord of Darkness" (shows under boss name)
- `Name Plate Duration` = 3 seconds
- `Disable Player Input` = true (player can't move during intro — **this is why your character freezes when boss spawns! Set to false if you don't want that**)

"I WANT THE BOSS TO HAVE SHIELDS"

In **Boss|Shield**: enable shield, set HP, set which attacks break it.

"I WANT AN ENRAGE TIMER"

In **Boss|Enrage**: set `Enrage Time` (e.g., 120 seconds). Boss goes berserk if not killed in time.

"I WANT MINIONS"

In **Boss|Minions**: enable minions, set minion class, spawn count, interval.

"I WANT AN ARENA BARRIER"

In **Boss|Arena**: set `Arena Radius`. Players can't leave the fight area. Arena auto-opens when boss dies.

Pro tip: Enable `Show Debug Logs` in **Boss|Debug** to see everything the boss does in the Output Log. Extremely helpful for understanding what's happening.

Previous Quick Start (for reference)

► Click to expand Option A / Option B setup

```
// In Details Panel:
Boss Name: "Dragon Lord"
Number Of Phases: 3
Phase Configs:
  [0] HealthPoints: 500, AttackType: Ranged
  [1] HealthPoints: 750, AttackType: Mixed, SpeedMultiplier: 1.5
  [2] HealthPoints: 1000, AttackType: Melee, DamageMultiplier: 2.0
Projectile Class: BP_BossProjectile
```

Multi-Phase Combat

Bosses support 1–3 phases, each with independent configuration via `FBossPhaseConfig` array. When HP drops to 0 in a phase, the boss transitions to the next phase with optional cutscene, healing, and invulnerability.

PER-PHASE CONFIG

PROPERTY	TYPE	DESCRIPTION
<code>HealthPoints</code>	float	HP for this phase
<code>AttackType</code>	<code>EBossAttackType</code>	Ranged, Melee, Mixed, BarrelThrow, Custom
<code>FireRateMultiplier</code>	float	Attack speed multiplier (1.0 = base)
<code>SpeedMultiplier</code>	float	Movement speed multiplier
<code>DamageMultiplier</code>	float	Outgoing damage multiplier
<code>ProjectileCount</code>	int32	Number of simultaneous projectiles
<code>MeleeConfig</code>	<code>FBossMeleeConfig</code>	Melee damage, range, knockback, cooldown
<code>MinionConfig</code>	<code>FBossMinionConfig</code>	Minion class, spawn rate, max alive
<code>PhaseCutscene</code>	<code>ULevelSequence*</code>	Optional cutscene on phase start
<code>PhaseTransitionMontage</code>	<code>UAnimMontage*</code>	Animation during phase transition

Combat System

The boss supports 5 base attack types configured per-phase:

TYPE	DESCRIPTION
Ranged	Fires projectiles at player (configurable class, speed, spread)
Melee	Close-range attack with configurable damage, range, knockback force
BarrelThrow	Throws explosive barrels with physics-based trajectory and AOE explosion
Mixed	Auto-switches between Ranged and Melee based on <code>MeleeSwitchDistance</code>
Custom	No built-in behavior — handle in Blueprint via <code>BP_OnMeleeAttackStarted</code>

External Health System

Set `HealthMode = External` to integrate with your own health system (GAS, custom damage component, etc.).

```
// In your damage handler:
BossComponent→ReportHealthChanged(CurrentHP, MaxHP);
// When boss dies:
BossComponent→ReportBossDefeated();
```

Phase transitions fire automatically based on reported health percentage!

Cutscene Transitions

Each phase can have an optional `ULevelSequence` cutscene that plays during phase transitions. Player/boss invulnerability is automatic during cutscenes. Players can skip via `SkipCurrentCutscene()`.

Minion Spawning

Per-phase minion configuration via `FBossMinionConfig`:

PROPERTY	DESCRIPTION
<code>MinionClass</code>	Actor class to spawn
<code>InitialSpawnCount</code>	How many to spawn when phase starts
<code>bContinuousSpawn</code>	Keep spawning at interval
<code>SpawnInterval</code>	Seconds between continuous spawns
<code>MaxAliveMinions</code>	Cap on simultaneous alive minions
<code>bKillMinionsOnPhaseEnd</code>	Auto-kill surviving minions on phase transition

Movement & AI

5 movement patterns via `EBossMovementType` :

TYPE	DESCRIPTION
<code>Stationary</code>	Boss stays at spawn location
<code>Circular</code>	Orbits around start position
<code>PatrolPoints</code>	Follows patrol waypoints
<code>RandomTeleport</code>	Teleports to random positions
<code>KeepDistance</code>	Maintains configured distance from player



Arena System v3.0

Configurable arena boundaries that lock players in during the boss fight. Arena auto-closes on fight start and auto-opens on defeat.

PROPERTY	DESCRIPTION
ArenaShape	Box, Sphere, or Cylinder
ArenaRadius	Radius / half-extent X
PushForce	Force pushing players back inside
BoundaryDamagePerSecond	DPS for players outside arena
bCloseOnFightStart	Auto-close when fight begins
bOpenOnDefeat	Auto-open when boss dies

```
// Manual control:
BossComponent→SetArenaClosed(true);
bool bInside = BossComponent→IsInsideArena(PlayerLocation);
```

AOE / Ground Attacks v3.0

5 AOE shapes with warning indicators, delayed damage, and knockback:

SHAPE	DESCRIPTION
Circle	Standard circular AOE around a point
Line	Straight-line beam (boss forward direction)
Cone	Frontal cone with configurable angle
Rectangle	Rectangular zone (width × length)
Cross	X-shaped cross pattern

```
// Trigger AOE at boss location:
BossComponent→PerformAOEAttack();
// Trigger at specific location:
BossComponent→PerformAOEAttackAtLocation(TargetLocation);
```

AOE has a `WarningDelay` — visual indicator appears first, damage happens after delay. Bind to `OnBossAOEWarning` to show custom indicators.

Dash / Charge Attack v3.0

Boss wind-up → charge → impact → recovery vulnerability cycle:

PROPERTY	DESCRIPTION
<code>DashSpeed</code>	Units per second during charge
<code>WindUpTime</code>	Delay before charge starts (animation time)
<code>MaxDashDistance</code>	Maximum distance boss will travel
<code>DashDamage</code>	Damage on direct contact
<code>ImpactDamage</code> / <code>ImpactRadius</code>	AOE explosion at end of dash
<code>RecoveryTime</code>	Vulnerability window after dash ends

```
BossComponent→PerformDashAttack(); // Dash toward player
BossComponent→PerformDashToLocation(Target); // Dash to specific point
```

Shield / Armor System v3.0

3 shield types configured per-phase via `PhaseShieldConfigs` array:

TYPE	DESCRIPTION
HealthShield	Shield has HP. Absorbs damage until broken. Regenerates after cooldown.
PhaseShield	Blocks all damage until manually broken (e.g. destroy shield generators).
TimedShield	Active for X seconds, then breaks automatically.

```

BossComponent→ActivateShield();
float overflow = BossComponent→DamageShield(100.0f);
BossComponent→BreakShield();

```

Shield automatically absorbs damage in `TakeDamage()` — you don't need to call `DamageShield` manually unless building custom damage pipeline.

Vulnerability Windows v3.0

Boss is invulnerable by default. Players must deal enough hits to **stagger** the boss, opening a damage window.

PROPERTY	DESCRIPTION
bInvulnerableByDefault	Boss takes no damage until vulnerable
StaggerThreshold	Hits needed to stagger boss
StaggerDuration	Seconds boss remains vulnerable after stagger
VulnerabilityDamageMultiplier	Damage multiplier during vulnerable state (e.g. 2.0x)

```

// Manual control:
BossComponent→MakeBossVulnerable(5.0f); // 5 second window
BossComponent→MakeBossInvulnerable();

```

Enrage / Berserk Timer v3.0

MMO-style berserk timer. When time runs out, boss enrages with massive stat boost or instant-kill wipe mechanic.

PROPERTY	DESCRIPTION
<code>EnrageTime</code>	Seconds until enrage
<code>EnrageDamageMultiplier</code>	Damage multiplier after enrage (e.g. 5.0x)
<code>EnrageSpeedMultiplier</code>	Movement speed multiplier
<code>bInstantKillOnEnrage</code>	Kill all players after delay
<code>bGradualScaling</code>	Stats scale gradually as timer runs down

Boss Intro Sequence v3.0

4 intro styles:

STYLE	DESCRIPTION
<code>None</code>	Fight starts immediately
<code>NamePlate</code>	HUD shows boss name + title (e.g. "DRAGON LORD — The Eternal Flame")
<code>CameraFocus</code>	Camera zooms on boss + name plate
<code>FullCutscene</code>	Plays ULevelSequence cutscene before fight starts

Player input is optionally disabled during intro. Boss is invulnerable until intro finishes. Skippable via `SkipBossIntro()`.

Attack Patterns / Combos v3.0

Define sequence-based attack patterns with weighted random selection.

```
// Example: "Heavy Combo" pattern
Pattern: "HeavyCombo"
Sequence:
  [0] Ranged x3, Delay: 0.5s
  [1] Wait, Delay: 1.0s
  [2] Dash x1
  [3] AOE x1
  [4] Melee x2, Delay: 0.3s
Weight: 10
MinDistance: 300
Cooldown: 8.0s
```

Each step supports: Ranged, Melee, AOE, Dash, Summon, Shield, Custom, Wait

Loot Table v3.0

Configure drops with rarity system, drop chances, and physics scatter:

PROPERTY	DESCRIPTION
ItemClass	Actor class to spawn as loot
DropChance	Probability (0.0–1.0)
MinQuantity / MaxQuantity	Random quantity range
bGuaranteed	Always drops (ignores chance)
Rarity	Common, Uncommon, Rare, Epic, Legendary
PhaseRequirement	Only drops if boss defeated in specific phase

Aggro / Threat System v3.0

Multiplayer threat management. Boss targets the player with highest threat.

```
// Add threat when player deals damage:
BossComponent→AddThreat(PlayerActor, DamageDealt);

// Tank taunts boss for 10 seconds:
BossComponent→Taunt(TankActor, 10.0f);

// Get current target:
AActor* Target = BossComponent→GetCurrentAggroTarget();
```

Threat decays over time. Boss only switches targets when threat difference exceeds

`ThreatSwitchThreshold` (hysteresis).

Boss HUD Widget v3.0

Auto-creating C++ UUserWidget with full layout built in code. Override with Blueprint subclass for visual customization.

Built-in elements: Boss name, title text, health progress bar (smooth interpolation), shield progress bar, phase indicator pips (diamond shapes), enrage countdown timer, vulnerability flash indicator.

```
// Auto-create (enabled by default):
bAutoCreateHUD = true;

// Use custom Widget BP subclass:
HUDWidgetClass = BP_MyBossHUD;

// Manual control:
BossComponent→CreateHUD();
BossComponent→DestroyHUD();
```

Override any of the 11 `BP_On*` events in your Widget Blueprint to customize appearance:

`BP_OnHealthChanged`, `BP_OnPhaseChanged`, `BP_OnShieldChanged`,

BP_OnVulnerabilityChanged , BP_OnEnrageTimerUpdated , BP_OnBossDefeated ,
BP_OnShowIntroNamePlate , BP_OnHUDInitialized , BP_OnHealthBarTick

API — Enums

Core Enums (BossComponent.h)

ENUM	VALUES
EBossProjectileType	Boulder, ExplosiveBarrel, AcidSpit, FireBall, IceSpike, Custom
EBossMovementType	Stationary, Circular, PatrolPoints, RandomTeleport, KeepDistance
EBossAttackType	Ranged, Melee, BarrelThrow, Mixed, Custom
EBossHealthMode	Standalone, External

V3.0 Enums (BossAbilityTypes.h)

ENUM	VALUES
EBossAOEShape	Circle, Line, Cone, Cross, Rectangle
EBossShieldType	HealthShield, PhaseShield, TimedShield
EBossIntroStyle	None, NamePlate, CameraFocus, FullCutscene
EBossSequenceAction	Ranged, Melee, AOE, Dash, Summon, Shield, Custom, Wait
EBossLootRarity	Common, Uncommon, Rare, Epic, Legendary
EBossArenaShape	Box, Sphere, Cylinder

API — Structs

STRUCT	USED IN	DESCRIPTION
<code>FBossPhaseConfig</code>	Core	Per-phase: HP, attack type, multipliers, minions, cutscene
<code>FBossMeleeConfig</code>	Core	Melee: damage, range, radius, cooldown, knockback, montage, sound
<code>FBossMinionConfig</code>	Core	Minions: class, count, interval, max alive, spawn points
<code>FBossAOEConfig</code>	v3.0	AOE: shape, radius, damage, warning delay, knockback, VFX
<code>FBossDashConfig</code>	v3.0	Dash: speed, distance, damage, wind-up, recovery, impact AOE
<code>FBossShieldConfig</code>	v3.0	Shield: type, HP, duration, regen, damage reduction, VFX
<code>FBossVulnerabilityConfig</code>	v3.0	Vulnerability: invulnerable default, stagger threshold/duration
<code>FBossEnrageConfig</code>	v3.0	Enrage: timer, multipliers, instant kill, gradual scaling
<code>FBossIntroConfig</code>	v3.0	Intro: style, title, duration, disable input, cutscene ref
<code>FBossAttackPattern</code>	v3.0	Pattern: sequence entries, weight, distance, cooldown
<code>FBossSequenceEntry</code>	v3.0	Single step: action type, delay, repeat count
<code>FBossLootConfig</code>	v3.0	Loot master config: table, spawn radius, launch, VFX
<code>FBossLootEntry</code>	v3.0	Single loot: item class, chance, quantity, rarity, phase req
<code>FBossAggroConfig</code>	v3.0	Aggro: decay rate, threat multiplier, switch threshold
<code>FBossAggroEntry</code>	v3.0	Runtime: player ref, threat level, total damage

FBossArenaConfig

v3.0

Arena: shape, radius, push force, boundary damage, VFX

API — Functions

Core Combat

CALLABLE

`void StartBossFight()`

Start the boss fight. Initializes HP, starts intro (if configured), creates HUD, closes arena.

CALLABLE

`void StopBossFight()`

Stop the boss fight. Cleans up timers, destroys minions.

CALLABLE

`void TakeDamage(float DamageAmount)`

Deal damage to boss (Standalone mode). Auto-handles shield absorption, vulnerability check, stagger counting, and phase transitions.

CALLABLE

`void ForcePhase(int32 Phase)`

Force boss to enter specific phase.

CALLABLE

`void PerformMeleeAttack()`

Trigger melee attack manually.

PURE

`float GetHealthPercent() const`

Returns current HP percentage (0.0 – 1.0).

PURE

`int32 GetCurrentPhase() const`

Returns current phase number (1-based).

PURE

`bool IsBossFightActive() const`

Is the boss fight currently active?

External Health

CALLABLE

`void ReportHealthChanged(float CurrentHP, float MaxHP)`

Report HP from external system. Triggers phase transitions based on percentage.

CALLABLE

`void ReportBossDefeated()`

Report boss death from external system.

Cutscenes & Minions

CALLABLE

`void SkipCurrentCutscene()`

Skip currently playing cutscene.

CALLABLE

`void SpawnMinions(int32 Count)`

Spawn minions using current phase config.

CALLABLE

```
void KillAllMinions()
```

Destroy all alive minions immediately.

CALLABLE

```
void ReportMinionDeath(AActor* Minion)
```

Report minion death (call from minion's OnDestroyed).

V3.0 — Arena

V3.0 CALLABLE

```
void SetArenaClosed(bool bClosed)
```

Manually open/close the arena.

V3.0 PURE

```
bool IsInsideArena(FVector Location) const
```

Check if a location is inside the arena boundaries.

V3.0 — AOE & Dash

V3.0 CALLABLE

```
void PerformAOEAttack()
```

AOE attack at boss position using AOEConfig.

V3.0 CALLABLE

```
void PerformAOEAttackAtLocation(FVector Location)
```

AOE attack at specific world location.

V3.0 CALLABLE

```
void PerformDashAttack()
```

Dash toward current player/aggro target.

V3.0 **CALLABLE**

```
void PerformDashToLocation(FVector Target)
```

Dash toward specific world position.

V3.0 — Shield

V3.0 **CALLABLE**

```
void ActivateShield()
```

Activate shield for current phase (uses PhaseShieldConfigs).

V3.0 **CALLABLE**

```
void BreakShield()
```

Force-break the shield.

V3.0 **CALLABLE**

```
float DamageShield(float DamageAmount)
```

Deal damage to shield. Returns overflow damage that passes through.

V3.0 — Vulnerability & Enrage

V3.0 **CALLABLE**

```
void MakeBossVulnerable(float Duration)
```

Open vulnerability window for specified duration.

V3.0 **CALLABLE**

```
void ForceEnrage()
```

Trigger enrage immediately.

V3.0 **PURE**

```
float GetEnrageTimeRemaining() const
```

Seconds remaining until enrage.

V3.0 — Intro, Patterns, Loot, Aggro

V3.0 **CALLABLE**

```
void PlayBossIntro()
```

Play intro sequence (auto-called by StartBossFight if configured).

V3.0 **CALLABLE**

```
void SkipBossIntro()
```

Skip intro and start fight immediately.

V3.0 **CALLABLE**

```
void ForceAttackPattern(FName PatternName)
```

Force boss to execute specific named pattern.

V3.0 **CALLABLE**

```
TArray<AActor*> SpawnLoot()
```

Spawn loot drops manually (auto-called on defeat if enabled).

V3.0 **CALLABLE**

```
void AddThreat(AActor* Player, float ThreatAmount)
```

Add threat value for player in aggro table.

V3.0 **CALLABLE**

```
void Taunt(AActor* Player, float Duration)
```

Force boss to target specific player for duration.

V3.0 — HUD

V3.0 **CALLABLE**

```
UBossHUDWidget* CreateHUD()
```

Create and display boss HUD widget.

V3.0 **CALLABLE**

```
void DestroyHUD()
```

Remove boss HUD from viewport.

Events & Delegates

All delegates are `DYNAMIC_MULTICAST` — bindable from both C++ and Blueprint.

Core Events (v2.0)

DELEGATE	PARAMETERS
OnBossHealthChanged	float HealthPercent, float CurrentHealth, float MaxHealth
OnBossPhaseChanged	int32 NewPhase, int32 MaxPhases
OnBossFightStarted	FText BossName
OnBossDefeated	FText BossName
OnBossMeleeAttack	(none)
OnBossMeleeHit	AActor* HitActor, float Damage
OnBossRequestDamage	AActor* DamagedActor, float DamageAmount
OnBossCutsceneStarted	int32 Phase
OnBossCutsceneFinished	int32 Phase
OnMinionSpawned	AActor* Minion
OnMinionDied	AActor* Minion
OnAllMinionsDead	(none)
OnMinionWaveSpawned	int32 Count

V3.0 Events

DELEGATE	PARAMETERS
OnBossAOEWarning	FVector Location
OnBossAOEDamage	FVector Location, int32 PlayersHit
OnBossDashStarted	(none)
OnBossDashHit	AActor* HitActor
OnBossDashEnded	(none)
OnBossShieldActivated	(none)
OnBossShieldBroken	(none)
OnBossShieldChanged	float CurrentShield, float MaxShield
OnBossVulnerabilityChanged	bool blsVulnerable
OnBossStaggered	(none)
OnBossEnraged	(none)
OnBossEnrageWarning	float SecondsRemaining
OnBossIntroStarted	(none)
OnBossIntroFinished	(none)
OnBossPatternStarted	FName PatternName
OnBossSequenceAction	EBossSequenceAction Action, int32 StepIndex
OnBossAggroTargetChanged	AActor* NewTarget, AActor* PreviousTarget
OnBossLootSpawned	TArray<AActor*> SpawnedLoot
OnBossArenaStateChanged	bool blsClosed

Blueprint Implementable Events

Override these in Blueprint to add custom behavior:

EVENT	WHEN
BP_OnBossFightStarted	Fight begins (after intro)
BP_OnBossDefeated	Boss HP reaches 0 in final phase
BP_OnPhaseChanged	Phase transition completes
BP_OnMeleeAttackStarted	Melee attack initiated
BP_OnMeleeHit	Melee attack connects with player
BP_OnCutsceneStarted / Finished	Phase cutscene start/end
BP_OnMinionSpawned / Died	Individual minion spawn/death
BP_OnAllMinionsDead	All tracked minions are dead
BP_DealDamageToActor	Boss requests damage on actor (override for custom damage)
BP_OnAOEWarning / Damage	AOE warning/damage phase
BP_OnDashStarted / Ended	Dash charge cycle
BP_OnShieldActivated / Broken	Shield state change
BP_OnBossVulnerable	Vulnerability window opens
BP_OnBossStaggered	Boss staggered from hits
BP_OnBossEnraged	Enrage timer expires
BP_OnIntroStarted / Finished	Intro sequence lifecycle
BP_OnSequenceActionCustom	Custom action in attack pattern

BossCharacter AI

`ABossCharacter` is an optional pre-built Character with BossComponent + AI navigation. It provides:

- **5 AI states:** Idle, Attacking, Repositioning, Retreating, Circling
 - **NavMesh integration:** walks to patrol points, avoids walls
 - **Arena awareness:** stays within arena boundaries
 - **Auto-start fight:** optional auto-trigger with configurable delay
 - **Debug visualization:** draw ranges, states, arena bounds
-

Changelog

v3.0.0 — February 2026 — "The Grand Update"

11 new systems, 2 new source files, 8000+ lines of C++

- **Arena System** — Configurable boundaries (Box/Sphere/Cylinder), push-back, boundary DPS
- **AOE/Ground Attacks** — 5 shapes, warning delay, decal indicators, knockback
- **Dash/Charge Attack** — Wind-up → charge → impact → recovery vulnerability
- **Shield/Armor System** — 3 types (Health/Phase/Timed), regeneration, damage reduction
- **Vulnerability Windows** — Invulnerable by default, stagger system, damage multiplier
- **Enrage/Berserk Timer** — Gradual scaling, instant-kill wipe mechanic
- **Boss Intro Sequence** — 4 styles, name plate, camera focus, full cutscene
- **Attack Patterns/Combos** — Sequence-based, weighted random, cooldowns
- **Loot Table** — Rarity system, drop chances, phase-specific, physics scatter
- **Aggro/Threat System** — Threat decay, hysteresis, taunt, multiplayer support
- **Boss HUD Widget** — C++ UUserWidget with 11 BP customization events
- **Bug fixes:** GetWorld() null safety (16 fixes), static variable isolation, bracket balancing, montage cleanup
- New files: BossAbilityTypes.h, BossHUDWidget.h/.cpp
- Dependencies: +Niagara, +UMG

v2.0.0 — January 2026

- FBossPhaseConfig system replacing legacy arrays
- 5 attack types: Ranged, Melee, BarrelThrow, Mixed, Custom
- External Health System (GAS integration)
- Melee combat with knockback
- Minion spawning system
- ULevelSequence cutscene support
- BossCharacter AI with 5 movement patterns
- 14 delegates, 20+ Blueprint functions

v1.0.0 — December 2025

- Initial release
- Multi-phase boss system
- Projectile attacks
- Basic movement patterns
- 4 events, 4 Blueprint functions

FAQ

Can I use this without BossCharacter?

Yes! `UBossComponent` is an ActorComponent — attach it to any Character or Pawn. BossCharacter is an optional convenience class.

Does External Health mode work with GAS?

Yes. Set `HealthMode = External`, then call `ReportHealthChanged()` from your GAS attribute callbacks. Phase transitions fire automatically.

Can I mix v3.0 systems selectively?

Absolutely. Every v3.0 system has an enable flag. Arena: `bEnableArena`, Enrage: `bEnableEnrage`, Aggro: `bEnableAggro`, Loot: `bEnableLoot`, etc. Disabled systems have zero performance cost.

Is multiplayer supported?

The Aggro/Threat system is designed for multiplayer. Arena boundary enforcement works for all player controllers. However, boss state replication (health sync, phase sync) requires your own networking layer.

How do I customize the HUD?

Create a Blueprint subclass of `UBossHUDWidget`, set it as `HUDWidgetClass` on `BossComponent`, then override any of the 11 `BP_0n*` events for custom visuals.

Support

Documentation	adrenalinegames.pl/epicbosssystem
Marketplace	Fab Store
Engine	Unreal Engine 5.6, 5.7
Platform	Win64
Dependencies	Niagara (built-in), LevelSequence (built-in)
Language	C++ with full Blueprint exposure
