

# Final IK Change Log

## [Version 0.4 - TBD](#)

- [AimIK](#)
- [FullBodyBipedIK](#)
- [Interaction System](#)
- [Common](#)
- [LimbIK](#)
- [Grounder](#)
- [BipedIK](#)
- [FABRIK](#)
- [Third Party Support](#)
- [Upgrade Guide](#)

## [Version 0.3 - 07.04.2014](#)

- [FullBodyBipedIK](#)
- [FABRIK](#)
- [LimbIK](#)
- [Grounder](#)
- [Common](#)

## [Version 0.22 - 13.03.2014](#)

- [FullBodyBipedIK](#)
- [AimIK](#)
- [FABRIKRoot](#)
- [Upgrade Guide](#)

## [Version 0.21 - 20.02.2014](#)

## [Version 0.2 - 19.02.2014](#)

- [FullBodyBipedIK](#)
- [LookAtIK](#)
- [AimIK](#)
- [Rotation Limits](#)
- [Common](#)
- [Documentation](#)
- [Upgrade Guide](#)

## [BETA \(0.1\) - 15.01.2014](#)

## Version 0.4 - TBD

### AimIK

1. Fixed error when Clamp Weight was 1
2. Added polePosition, poleWeight and poleTarget to the solver. This enables us to keep another axis of the Aim Transform oriented at a position in world space

### FullBodyBipedIK

1. About 20% performance improvement
2. Fixed initiation error when manually setting up the bone references in the Editor
3. Improved bend direction stabilization
4. Added FBlkChain.push and FBlkChain.pushParent
5. New custom editor
6. FBlkIK iterations can be now set to 0. In that case, full body effect is disabled and only trigonometric passes will be calculated.
7. Added effector target transforms. You can now assign them in the Editor or write `ik.leftHandEffector.target = transform`. FBlkIK will automatically set effector position and rotation to match the target transform's. It will overwrite `IKEffector.position = something`.
8. Added the Full Body FPS demo scene and scripts
9. Added the Pendulum demo scene to demonstrate how a character could be mapped to a ghost ragdoll with FBlkIK

### Interaction System

1. InteractionObject now also works with Legacy
2. Icons for all Interaction System components
3. Added Push and PushParent to InteractionObject weight curve types
4. All InteractionSystem methods now return a bool value notifying if the operation was actually carried out or not. So if `StartInteraction()` returns false, the interaction did not start (maybe because the effector was already in interaction).
5. Integrated InteractionLookAt to InteractionSystem to reduce the number of components. All used InteractionLookAt components need to be removed, Unity will give a warning if it finds any.
6. Added `InteractionObject.WeightCurve.Type.PoserWeight`. That will be used to determine weight of the hand posers.
7. Added picking up spherical objects to the Interaction Pickup2Handed demo
8. Restructured InteractionObject to add events that are easier to understand.

## Common

1. Added IKExecutionOrder for easy editing of the order in which the IK components update their solvers.
2. All components of Final IK now have “User Manual” and “Script Reference” buttons in their context menu.
3. All component custom inspectors have a warning box now to inform you of invalid/incomplete setups without spamming the console.
4. Many bugfixes for custom inspectors
5. Revised component menu structure
6. Improved scene view handle and button scaling
7. Added range sliders to inspectors
8. Fixed the bug with IK components that found the wrong Animator/Animation component from the character hierarchy to get the Animate Physics value from
9. Improved the MechSpider demo, the spider is now capable of climbing vertical surfaces

## LimbIK

1. Changed IKRotation to match the orientation of the last bone, like FBBIK effectorRotation
2. SetBendGoalPosition now takes a weight parameter.
3. Added the “Goal” bend modifier that allows you to assign a bend goal Transform.

## Grounder

1. Small fixes, Grounder components can be added in runtime without errors
2. Added lowerPelvisWeight and liftPelvisWeight to the Grounding solver
3. Added horizontal wall running to the demo scene
4. Enabled strafing for the biped character controller in the demo scene (switch Move Mode to Strafe)
5. Added OnPreGrounder and OnPostGrounder delegates to the Grounder components

## BipedIK

1. Simplified Pelvis constraints. Instead of bipedIK.solvers.pelvis.positionConstraint.position you can now use bipedIK.solvers.pelvis.position. Same with positionWeight, positionOffset, rotation, rotationWeight and rotationOffset.

## FABRIK

1. 2-3 times faster constrained FABRIK chains.
2. Removed `IKSolverFABRIK.updateBoneLengths`. It will always update bone lengths and axes now, making it possible to skip animated bones in the hierarchy.

## Third Party Support

1. Playmaker actions for all IK components and the Interaction System

## Upgrade Guide

1. **Backup your project before upgrading!**
2. LimbIK `IKRotation` has been changed to match the orientation of the last bone like `FBBIK effectorRotation`. If you are using LimbIK or BipedIK somewhere, you will have to rotate the targets to match the exact desired rotations of the hand bones. This change will simplify setting up IK targets in the future (just copy the hand bone, pose it and use it as the target).
3. If you have used BipedIK pelvis constraints, you need to change `bipedIK.solvers.pelvis.positionConstraint.position` to `bipedIK.solvers.pelvis.position`. Same with `positionWeight`, `positionOffset`, `rotation`, `rotationWeight` and `rotationOffset`.
4. Integrated `InteractionLookAt` to `IntegrationSystem` to reduce the number of components. All used `InteractionLookAt` components need to be removed, Unity will give a warning if it finds any.
5. All `InteractionObjects` will need to specify `PoserWeight` curve or multiplier if you wish to use `HandPosers`.
6. Restructured `InteractionObject` to an event based system. Some properties like `triggerTime`, `releaseTime`, the animations and message recipients will have to be reassigned for the events.

## Version 0.3 - 07.04.2014

### FullBodyBipedIK

1. Added OffsetEffector.cs and demo scene
2. Added Interaction Walls demo scene and script
3. Fixed FixTransforms bug that was causing some twitching on some unanimated rigs
4. Improved spine mapping performance and accuracy. Not all spine bones need to be included in the spine references. It works the fastest if spine length is 2, first bone in the spine is the root node and the other is the last spine bone.
5. Improved solver weight blending. You can now weigh out the solver without dislocating the limbs even when the effectors are pinned
6. Improved the custom inspector and the validation of the biped references.

### FABRIK

1. Improved solver stability under constraints.

### LimblIK

1. Animated bones can be skipped in the hierarchy when assigning bones for LimblIK and TrigonometricIK. The last bone will still be solved to the correct position.

### Grounder

1. Added the GrounderFBBIK, GrounderBipedIK, GrounderIK and GrounderQuadruped components and the Grounder demo scene.

### Common

1. Added OnPreInitiate, OnPostInitiate, OnPreUpdate and OnPostUpdate delegates to all IK solvers.
2. IK Component inspectors draw the scene view handles for disabled IK components
3. All IK components not will look up the hierarchy to find the first Animation/Animator component to know if animatePhysics is on or off for the character.
4. Improved CameraController demo script

### Upgrade Guide

1. **Backup your project before upgrading!**

## Version 0.22 - 13.03.2014

### FullBodyBipedIK

1. ShoulderRotator now works for characters that have animatePhysics enabled.
2. Added BipedLimbOrientations. It is now very easy to fix limb bending directions for UMA, 3ds Max and other skeleton types if necessary  
(ik.solver.SetLimbOrientations(BipedLimbOrientations.UMA);). Removed IKConstraintBend.SetBendDirection() and IKMappingLimb.SetBendDirection().
3. Switched FBBIK limbs from 1DOF joints to 3DOF joints. This does not enforce the limbs to behave like hinge joints anymore and will allow for lossless solving and mapping of the limbs, meaning that if you have FBBIK on with 0 effector weights, the animation will remain the same.
4. Removed IKConstraintBend.BendBone because it is not necessary anymore after switching to 3DOF joints.
5. Restructured FBBIK chain structure to remove object composition cycle. This change is required for upgrading to Unity 4.6 (Beta).

### AimIK

1. Added the Aim Swing demo scene.

### FABRIKRoot

1. Restructured to remove object composition cycle. This change is required for upgrading to Unity 4.6 (Beta).

### Upgrade Guide

#### **1. Backup your project before upgrading!**

1. IKConstraintBend.BendBone was removed, if you have any code using it, just delete it, will not be necessary anymore.
2. FullBodyBipedIK chain structure was restructured, so all used FBBIK components have to be reinitialized. Just right-click on the FBBIK component and select Reinitiate from the context menu. Pull and Reach values of the chains will reset to defaults.
3. FABRIKRoot was restructured and the chains have to be rebuilt in the inspector.

## Version 0.21 - 20.02.2014

1. Removed Button.cs, a relic testing script that was unused and not namespaced.

## Version 0.2 - 19.02.2014

### FullBodyBipedIK

1. Better scaling of the effector handles (for extra large/small characters)
2. Fixed a bend constraint bug that occurred with very tiny characters
3. Added shortcuts to limb IK mappings (IKSolverFullBodyBiped.leftArmMapping, IKSolverFullBodyBiped.rightArmMapping, ...)
4. Added IKMappingLimb.weight for spherical interpolation of the limbs and for the possibility of disabling the effect of IK for a limb.
5. Added reach smoothing modes (FBIKChain.reachSmoothing).
6. Added IKSolverFullBodyBiped.GetLimbMapping(FullBodyBipedEffector).
7. Added IKConstraintBend.SetBendDirection() and IKMappingLimb.SetBendDirection() to enable you to change the bending direction of the limb.
8. Added Amplifier and a demo scene for it.
9. Added OffsetPose.Apply(IKSolverFullBodyBiped solver, float weight, Quaternion rotation)
10. Fixes to Inertia deltaTime issues.
11. Removed IKEffector.Mode, you can use IKEffector.maintainRelativePositionWeight now for smooth blending between the former MaintainAnimatedPosition and MaintainRelativePosition
12. Added OffsetModifier that will be the base abstract class for Inertia, BodyTilt, Amplifier, EffectorOffset and all other FBBIK effector positionOffset modifiers in the future. OffsetModifier works with animatePhysics, uses delegates safely and makes it easy to apply limits to the offset. It will also make it easier for you to create your custom offset modifiers, check out EffectorOffset.cs.
13. Added the TerrainOffset demo that was used to make the AimIK - Redirecting Animation tutorial.
14. Fixed IKSolverLookAt.SetChain. The LookAt solver now works with no head and nulls can be passed to SetChain.
15. Added GenericPoser, which is similar to HandPoser, but enables you to pose hierarchies that have a different number of bones.
16. Added the Interaction System and with it 3 demo scenes: Interaction, Interaction Character2Character and Interaction Pickup2Handed.

### LookAtIK

1. Improved IKSolverLookAt. It now looks better with animations that have strong amplitude on the spine such as running and sprinting.

### AimIK



1. RotationLimits can be used on the Aim Transform of AimIK now.

## Rotation Limits

1. Fixed RotationLimitAngle twist limit when swing limit is 0.

## Common

1. Added V3Tools to help dealing with vector algebra.
2. Added Fix Transforms option to all the IK components. Its now possible to use FBBIK and BipedIK with no Animation/Animator component. With Fix Transforms set to true, there will be more issues with unanimated bones.
3. Clamped all solver weights to 0-1.

## Documentation

1. Updated User Manual and Script Reference to 0.2
2. Added diagrams to the Script Reference

## Upgrade Guide

1. **Backup your project before upgrading!**
2. The new Fix Transforms option for IK components will be defaulted to true. You can turn it off for a small performance gain on solvers you don't need it for. It will also make any unanimated IK chain reset to it's initial pose in each Update before solving, so if you need additive solving of your CCD/FABRIK/FABRIKRoot chains, turn it off.
3. IKEffector.Mode was changed to IKEffector.maintainRelativePositionWeight, so if you used MaintainRelativePosition anywhere, you will have to change it to effector.maintainRelativePositionWeight = 1.
4. Changes to BodyTilt behaviour, you may need to adjust the OffsetPoses for tilting.

**BETA (0.1) - 15.01.2014**

Initial Release