What’s Left Todo:

1. The Mesh Systems in SigFx are currently affected by any attractor placed down. Users will want to select what attractors do/don’t affect it. You can look at how ParticleSystems do this already.
2. The bounding box used for selection of MeshSystems is currently fixed at a small value. Using the combined boxes of all emitted MeshParticles will result in a better fitting bounding box.
3. There is a default shader written that doesn’t require a texture. However during the build process the builder will warn that the build wasn’t complete because the particle system didn’t have a ‘material’ applied to it. Look at the constructor of tParticleSysem( const tParticleSystemDef\* def ), and also currently I am auto-applying a standard shader that does require a texture, and that auto-applying is done in: tNewParticleSystemObjectCursor.
4. Debug geometry for the “Travel-over-Time” gizmo is not showing up. This geometry is very useful for visualization of what’s going on. Either re-enable debug geometry, or look in tEffectPositionOffsetGizmo::fOnTick and use something besides debug geometry for visualization.
5. The way I keep Tool vs. Binary data separate for ParticleSystems, Attractors and MeshSystems is done with a class that contains pointers to Tool or Binary data, and then every time a graph is sampled from one of those systems, we check if either Tool or Binary data is available; then use the graph from that to sample from. Would be much cleaner/faster if we didn’t need to do those checks. The three holding classes to look at and change are: tParticleSystemState, tAttractorData, and tMeshSystemData.
6. New/More shader variety for particles.