Math

- **SFEulerAngles3f**, **SFQuaternion**, **SFUnitVectore3f** are new math modules which help to work with transforms. **SFTransform3f** is a new **3D** trasform module.

Pipeline

- **SFPipeline** no more generate 2 shaders if the shaders use the same programs.
- **SFPipelineGrid** now can keep some **SFFunctions** which can be exploited in order to automate geoemtries term corrections.
- **SFPipelineMemory** has been simplified.
- **SFPipelineTrasform3f** and **SFPipelineTransforms** are new pipeline module used to manage modelviews.
- **SFRigidTransform3fArray** is a new abstraction for arrays of Transforms.
- **SFPrimitive** has been strongly rearranged. Now it introces the new enum PrimitiveBlock which is used to describe values which are managed from the Primitive itself. Primitive keeps a list of all the Grid Instances which are used by the Primitive itself and offer some utilities in order to help SFPrimitiveIndices in its work.
- **SFPrimitiveIndices** has been reworked to operate with a simple array of ints. This highly simplify the workaround of primitives management.
- **SFPrimitiveGrid** is a new module used by Primitive.

Builder

- **SFPipelineBuilder** is a new module which has been extracted from previous **SFPipelineLoader**. **SFPipelineBuilder** allows pipeline building even in situations in which the pipeline description is not in a file. **SFPipelineBuilder** comes with several modules for its own working.

Renderers and Scenegraphs

- The old **SFNode** has become an interface. **SFObjectModel** and **SFReferenceNode** are two alternative implementations. They are both subclasses of **SFTransformNode**. **SFBone** is another node, but its a work-in-progress.
- **SFTextureReference** and **SFTexture** are two new modules required to deal with any **SFRenderedTexturesSet.**
- **SFGraphicsAsset** is no more.

Images and Textures (High Level)

- **SFBitmapTexture**, **SFDrawnRenderedTexture** and **SFFilterdRenderedTexture** are 3 new high level types describing textures. They all implements the new interface **SFRenderedTexturesSet**.

Images and Textures (Low Level)

- **SFFormat** has been renamed to **SFImageFormat**.
- **SFTextureData** has been renamed to **SFPipelineTexture**.

Materials (High Level)

- **SFDataLightStep**, **SFNoDataLightStep**, **SFPassAllLodFilter** are some usefull modules which can be used for a fast construction of Renderers.

Geometries (High Level)

Geometry Classes

- **SFMeshGeometry** is no more abstract.
- **SFQuadsSurfaceGeometry** has been completely reworked in order to better use some of the new SF functionalities.

Curves

- **SFCurve** is Back from SF1.0. The new interface works with any SFValuenf, so that it can represent either 2D, 3D or 4D Curves.
- **SFValuesIterator** is a new abstraction used to iterate other a set of Values.
- **SFValueList** is a new abstraction for modules keeping a set of Values.
- **SFBasisSpline2**, **SFBezie2**, **SFBezier3**, **SFLine, SFPlacedCurve**, **SFStandardAbstractCurve**, **SFSpline**, **SFRationalCurve3f** are new valid instances of **SFCurve**.

Functions

- **SFBicurvedLoftedSurface**, **SFGuidedSurface**, **SFRadialSurfaceFunction**, **SFRectangle2DFunction** and **SFSplineCurvedTubeFunction** are new **SFSurfaceFunction**, which can be used in combination with **SFQuadsSurfaceGeometry**.
- **SFCurvedTubeFunction** has been reworked, in order to take advantage of the last changes.
- **SFArcLengthuv, SFCompositeGeometryuv, SFSimpleObjPlaneTexCoordGeometry, SFSimpleTexCoordGeometryuv** are new TextureFunctions (**SFSurfaceGeometryTexCoordFunctionuv**).

Geometry Generation

- **shadow.geometry.editing** has been removed because extractors are no more required since extraction mechanism has been automated at a lower level.

Geometries (Low Level)

- **SFExtruder** is a new module which is in charge of extrusion computation.
- **SFGridMap**, **SFStandardQuadExtractor**, **SFStandardQuadToQuadExtractor**, **SFStandardQuadToTriangleExtractor, SFGridOperations, SFTriangularGrid** are new modules used on geometries elaboration.

World

- Old SF1.0 world modules have been planted into SF2.0. A complete SF2.0 version of world libraries is a work-in-progress.

Data

Datasets

- **SFAbstractDatasetFactories** are now responsible for the way in which dataset are stored into SFStreams.
- SFLibraryReference is no more; it has been removed in order to validate the saparation introduced by **SFDataAssets (see Assets)**.

- **SFShort** and **SFInt** has new methods which allow to write each byte of both.
- **SFBinaryObject** is a new Object used to store single BinaryValues.
- **SFGenericInfoObjectBuilder** is a common utility used to generate CompositeDataObject for the purposes of **DataAssets**.

Assets

- Assets are a new important instrument of the entire framework. **SFDataAssets** are Datasets, so they can be stored into files and put into ObjectsLibraries, but they can generate rendering resources (which all are **SFInitiables**). This mechanism brings to a complete separation of rendering responsabilites from data responsabilities.

Renderers and Scenegraphs

- **SFBoneData**, **SFObjectModelData**, **SFReferenceNodeData** are **DataAsset** used in scenegraph description.
- **SFCameraData** and **SF2DCameraData** allow to generate and store cameras.
- **SFOneSteAlgorithmData**, **SFRenderedData**, are used to generate and store **Renderers** with their rendering algorithm.
- **SFStructureReferenceData** is used to generate and stores data for materials and lights.
- **SFClonedArrayReference, SFIndexedProxyDataCenter** and **SFIndexedProxyDataObject** are new data modules used to optimize the generation of array of models with similar properties.

Pipeline Data

- **Pipelines** may now be stored as well. In this way you can pre-compile your pipeline and use the compiled version. All classes in **shadow.pipeline.data** are in charge of this functionalities.

Texture Data

- **SFSimplePerlinNoiseData** is a new DataAsset which can be used to generate and store **SimplePerlinNoise** Bitmaps.
- **SFBitmapTextureData**, **SFDrawnRenderedTextureData**, **SFFilteredRenderedTexturedData** and **SFTextureDataObject** are new data modules to generate and stores **SFRenderedTexturesSet**.

Geometry Data

- **SFBasisSplineData, SFBinarySpline2D, SFCurveData,SFCurveData2D, SFCurveData3D, SFCurvesVerticesData, SFLineData, SFSplineData, SFUniformeBezier33fData** and **SFWeightData** are new **SFDataAsset** classes which can be used to generate and store different types of **SFCurves**.
- **SFFixedFloat**, **SFPoint2DData**, **SFPoint3DData** and **SFVertexFixedListData** are new classes which can be used to generate and store **SFValuenf,** most of all used in curves descriptions.
- **SFSimpleObjPlaneTexCoordGeometryData, SFSimpleTexCoordGeometryuvData** are new classes which can be used to generate and store **SFSurfaceGeometryTexCoordFunctionuv**.
- **SFBicurvedLoftedSurfaceData, SFCurvedTubeFunctionData, SFRadialSurfaceFunctionData, SFRectangle2D FunctionData, SFSplineCurvedTubeFunctionData** and **SFTwoCurvesFunctionData** are new classes which can be used to generate and store **SFSurfaceFunction**.
- **SFQuadsSurfaceGeometryData** is a new class which can be used to generate and store generates valid **SFGeometry**.