

# M3X

## Technical Design Document

Jacques Gasselin de Richebourg  
`jacques.gasselin@ironmonkeystudios.com`

June 19, 2008

## **Abstract**

The M3X project aims to provide tools for asset manipulation for the Mobile 3D Graphics API (M3G). M3X is being developed as a complement to Java Specification Request (JSR) #184 and #297.

JSR184 (M3G 1.0) and JSR297 (M3G 2.0) provide a solid interface for presenting 3D assets on Java capable devices supporting Connected Limited Device Configuration (CLDC) 1.1 .

A binary format for importing assets is clearly defined in the specification of both APIs. The proper ordering of references is defined as are the required binary data structures. A well defined approach to creating the binary asset is intentionally left out of the specification.

The M3X project supplies tools and a specification for creating binary assets that comply with M3G 1.0 and M3G 2.0. The intention is to remove a mechanical step from the asset pipeline; using an XML schema that presents a human readable interface to the M3G binary format.

XML files are easily edited by hand or using transformation scripts. Text format files also behave better than binary format files in most Version Control Systems. This is an important aspect for software development.

Version based change logs can be tracked. XML files can be easily validated to conform to a schema. Using the M3X tools; XML to binary conversion can be integrated into build pipelines.

## Chapter 1

# Application Programming Interface

### 1.1 M3X XML Data

An XML data file conforming to the m3x XML Schema can be manipulated using the `m3x.jaxb` package.

### 1.2 M3G Binary Data

A binary data file conforming either M3G 1.0 or M3G 2.0 can be manipulated using the `m3x.m3g` package.

### 1.3 External Data

External data can be converted to M3G or XML data using the translation layer in the `m3x.translation` package.