14 Annexes

14.1 Shaders list

non fx shader					
Shader name (non fx)	**Short name**	**Description**	**Texture 1**	**Texture 2**	**Texture 3**
AddAlphaDiff	AddAlphaDiff	No texture, additive vertex alpha with diffuse colour.	none	none	none
AddATex	AddATex	Texture mapped, no lighting applied, using additive alpha from texture's alpha channel	RGB: Colour A: Transp.	none	none
AddATexAlphaDiff		Texture mapped, with diffuse colour, using additive alpha from texture's alpha channel combined with vertex alpha			
BlendATexDiff	BlendATexDiff	Texture mapped, with diffuse colour, using additive alpha from texture's alpha channel	RGB: Colour A: Transp.	none	none
AddDiffuse		No texture, with diffuse colour, using additive alpha			
AddTex	AddTex	Texture mapped, no lighting applied, using additive alpha Useful for headlight or rearlight glass.	RGB: Colour	none	none
AddTexAlphaDiff		Texture mapped, with diffuse colour, with additive vertex alpha			
AddTexDiff		Texture mapped, with diffuse colour, using additive alpha			
BlendAlphaDiff		No texture, vertex alpha blending with diffuse colour			
BlendATex		Texture mapped, no lighting applied, using alpha blending from texture's alpha channel			
BlendATexAlphaDiff		Texture mapped, with diffuse colour, using alpha blending from texture's alpha channel combined with vertex alpha			
BlendATexDiffTrans		Texture mapped, diffuse colour, alpha blending from texture's alpha channel, pixels with alpha=0 are transparent (e.g. alphaed fences).			
BlendTexAlphaDiff´		Texture mapped, with diffuse colour, with vertex alpha blending			
BridgeSplit		Not drawn. Use to define areas where track crosses over itself.			
Diffuse		No texture, just diffuse colour			
DualAddATexDiffDestBlend		Dual textured, diffuse colour, first pass additive, and second pass blended alpha with the alpha of the first texture (e.g. puddles).			
DualBlendATexDiffAdd		Dual textured, with diffuse colour, using alpha blending for first pass and additive alpha for second pass			
DualTexDiffAdd		Dual textured, with diffuse colour, using additive alpha for second texture			
DualTexDiffAddWithLightIntens		Add second pass to first pass, brightness of second pass affected by lightmaps if used			
DualTexDiffAddWithoutLightIntens		Add second pass to first pass, brightness of second pass not affected by lightmaps if used			
DualTexDiffInvisibleStencilBlend		Dual textured, with diffuse colour, first pass invisible, second pass alphaed using alpha of first pass texture			
DualTexDiffStencilAdd		Dual textured, with diffuse colour, using additive alpha for second texture only where first texture has solid alpha			
DualTexDiffStencilBlend		Dual textured, with diffuse colour, using blended alpha for second texture only where first texture has solid alpha			

non fx shader					
Shader name (non fx)	**Short name**	**Description**	**Texture 1**	**Texture 2**	**Texture 3**
DualTexDiffTAlpha		Dual textured, with diffuse colour, using second texture's alpha channel to blend between textures			
DualTexDiffTrans		Dual textured, with diffuse colour, using second texture's transparency			
DualTexDiffVAlpha		Dual textured, with diffuse colour, using vertex alpha to blend between textures			
EmbossBumpmap		Bumpmap for Train 2 prototype or something like that			
Invisible		Nothing is drawn - use for invisible collision barriers			
Tex	Tex	Texture mapped, no lighting applied	RGB: Colour	none	none
TexDiff	TexDiff	Texture mapped with single texture, diffuse colour applied	RGB: Colour	none	none
TripleGlossMap		Triple texture, 2nd pass contains gloss map in alpha channel, 3rd pass (reflection) texture drawn additively			
TripleGlossMapWithLightIntens		Triple texture, 2nd pass alpha channel gloss map, 3rd pass drawn additively affected by lightmaps if used			
TripleGlossMapWithoutLightIntens		Triple texture, 2nd pass alpha channel gloss map, 3rd pass drawn additively not affected by lightmaps if used			
TripleTexDiffAddAdd		Triple textured, 2nd and 3rd passes are drawn additively			
TripleTexDiffTAlpha		Triple textured, with diffuse colour, using each texture's alpha channels to blend between each pair of passes			
TripleTexDiffTAlphaVAlpha		Triple textured, with diffuse colour, pass 2 uses texture alpha for blending, pass 3 uses vertex alpha for blending			
TripleTexDiffVAlpha		Triple textured, with diffuse colour, using same vertex alpha to blend between each pair of passes			
TripleTexDiffVAlphaTAlpha		Triple textured, with diffuse colour, pass 2 uses vertex alpha for blending, pass 3 uses texture alpha for blending			

fx shader					
shader name (fx)	**Short name**	**Description**	**Texture 1**	**Texture 2**	**Texture 3**
TrainEnv.fx	TrEnv		RGB: Colour	RGB: Dummy	none
LoftTexDiff.fx	LoftTexDiff		RGB: Colour	none	none
LoftTexDiffTrans.fx	LoftTexDiffTr		RGB: Colour A: Transp.	none	none
LoftBump.fx		Diffuse texture and normal map			
LoftBumpAlpha.fx		Diffuse texure with alpha and normal map			
LoftBumpTrans.fx		Diffuse texture with 1-bit alpha and normal map			
SkinAmbient.fx		Single colour skinned			
SkinDiffuse.fx	Skin	Textured skinned.	RGB: Colour A: Transp.	none	none
SkinGloss.fx		Textured, normal mapped, specular with gloss map, and skinned.	RGB: Colour	RGB: Normal Map	RGB: Gloss Map
SkinNormal.fx		Textured, normal mapped, specular and skinned.	RGB: Colour	RGB: Normal Map	none
SkinSpecular.fx		Textured, specular and skinned.	RGB: Colour	none	none
StencilShadow.fx	Shadow	Stencil shadow objects, material must begin with shadow_ to be detected. Not used anymore in TS2013 with TSX mode.	RGB: Colour	none	none
TrainBasicObjectDiffuse.fx	TrDiff	Single texture, dynamic lighting.	RGB: Colour	none	none
		Diffuse texture + black and white (no grey) alpha channel. Set keyword AlphaTestMode to 1 for the alpha channel to be used. Can be used for inscriptions.	RGB: Colour A: Transp.	none	none
TrainBasicObjectSpecular.fx	TrSpec	Texture, colour modulated specular.	RGB: Colour A: Transp.	RGB: Spec color map	none
TrainBumpEnv.fx		Textured, normal mapped, environment mapped.	RGB: Colour	RGB Normal Map	RGB: Dummy (Cubic Env)
TrainBumpEnvMask.fx		Textured, normal mapped, masked environment map.	RGB: Colour A: Env Mask	RGB: Normal map	RGB: Dummy (Cubic Env)
TrainBumpSpec.fx	TrBumpSpec	Textured, normal mapped, specular.	RGB: Colour A: Transp.	none	none
TrainBumpSpecEnv.fx	TrBumpSpecEM	Textured, normal mapped, environment map and specular.	RGB: Colour	RGB Normal Map	RGB: Dummy (Cubic Env)
TrainBumpSpecEnvMask.fx		Textured, normal mapped, masked environment map and specular.	RGB: Colour A: Env & Spec Mask	RGB: Normal map	RGB: Dummy (Cubic Env)

fx shader					
shader name (fx)	**Short name**	**Description**	**Texture 1**	**Texture 2**	**Texture 3**
TrainBumpSpecMask.fx		Textured, normal mapped, masked specular.	RGB: Colour A: Env Mask	RGB: Normal map	none
TrainDecal.fx		Diffuse texture + 8 bit alpha channel for transparency. For the alpha channel any level of grey can be used from black to white. Set keyword ZBufferMode to 3 for the alpha channel to be processed properly. The texture file name must start with decal_. Best choice for inscriptions.	RGB: Colour A: Transp.	none	none
TrainFlora.fx	TrFlora	Ambient lighting, single texture.	RGB: Colour	none	none
TrainGlass.fx		Screen space refractive glass with normal map and diffuse.	RGB: Colour	RGB: Normal map	Back buffer copy
TrainGlassWeatherEffects.fx		See specific table below.			
TrainLightMapWithDiffuse.fx	TrLightMap	Diffuse tex, lightmap, dynamic lighting.	RGB: Colour	RGB Lightmap	none
TrainLightBumpSpecMask.fx		Diffuse tex, normal map, Ambient Occlusion map.	RGB: Colour	RGB: Normal map	RGB: Occlusion map
TrainSkyDome.fx	Sky	Skydome	RGB: Colour	RGB: Dummy (Cubic Env)	none
TrainSpecEnv.fx		Textured, vertex environment mapped with specular.	RGB: Colour	RGB: Dummy (Cubic Env)	none
TrainSpecEnvMask.fx	TrSpecEM	Textured, masked vertex environment mapped with specular.	RGB: Colour A: Env & Spec Mask	RGB: Dummy (Cubic Env)	none
TrainUprightViewFacingFlora.fx	TrUpVFaceFlora	Single texture, globally lit, upright view facing	RGB: Colour A: Transp.	none	none
TrainVertexLit.fx		Diffuse tex, vertex lighting only.	RGB: Colour	none	none
TrainVertexLitWithDiffuse.fx		Diffuse tex, vertex lighting, dynamic lighting.	RGB: Colour	none	none
TrainViewFacingFlora.fx	TrVFaceFlora	Single texture, globally lit, view facing	RGB: Colour A: Transp.	none	none
WaterCubeMap.fx	Water	Splish	RGB: Colour A: Transp.	RGB: Normal map	none
TrainBumpEnv.fx		Textured, vertex environment mapped.	RGB: Colour	RGB: Dummy (Cubic Env)	none
TrainBumpEnvMask.fx		Textured, masked vertex environment map.	RGB: Colour A: Env. Mask	RGB: Dummy (Cubic Env)	none

fx shader						
shader name (fx)	**Short name**	**Description**	**Texture 1**	**Texture 2**	**Texture 3**	**Texture 4**
TrainGlassWeatherEffects.fx	TrGlassWeather	Reflective glass with cubic reflection map and diffuse.	RGB Diffuse A Translucency	Cubic environment map	Normal texture placeholder	Backbuffer placeholder

14.2 Shaders usage examples

	Shader name (texture slot 1)	Main texture (= "texture 1". In texture slot 2)	Bump map	Environment map [1]	Additional settings
Solid texture	TrainBasicObjectDiffuse.fx	name.bmp			
Solid texture with holes	TrainBasicObjectDiffuse.fx	name.tga (transparency in alpha channel. Only black or white)			AlphaTestMode=1 (needed for the alpha channel to be processed as a transparent layer)
Solid texture with holes (for inscriptions such as rolling stock numbers)	TrainDecal.fx	decal_name.tga (transparence in alpha chanel: any grey value between black and white)			ZBufferMode=3 (needed for the alpha channel to be properly processed as a transparent layer)
Texture with specular effects [3]	TrainSpecEnvMask.fx	name.tga (specular in alpha channel)		env.bmp (slot 3)	
Texture with specular effects and normal maps [3]	TrainBumpSpecEnvMask.fx	name.tga (specular in alpha channel)	name_nm.bmp (slot 3)	env.bmp (slot 4)	UV arguments suggested values:[2] CUSTOMPARAM0=32.0 (all other values = 0.0)
Windows only	TrainGlass.fx	name.tga (transparency in alpha channel)		env.bmp (slot 3)	UV arguments suggested values:[2] CUSTOMPARAM0=64 CUSTOMPARAM1=0.8 CUSTOMPARAM2=0.4 (all other values = 0.0)
Windows only	BlendATexDiff	name.tga (transparency in alpha channel)			ZBufferMode = 3 (needed for the alpha channel to be processed as a transparent layer)
Headlight or rearlight glass	AddATex	name.tga (transparency in alpha channel)			ZBufferMode = 3 (needed for the alpha channel to be processed as a transparent layer)
2D Vegetation	TrainUprightViewFacingFlora.fx ou TrainViewFacingFlora.fx	name.tga (transparency in alpha channel. Only black or white)			AlphaTestMode=1 (needed for the alpha channel to be processed as a transparent layer) "Optimized IGS" must be checked.
Animated texture http://railsimilarity.blogspot.fr/2009/01/how-toanimate-textures.html	AddATex	name_anim1.tga which is the first file of the animation.			AnimateUVs=1 NumFrames and FPS according to the animation to implement.

Notes:

(1)

Environment map is usually a 64x64 dummy black bmp file. It can be the same as the primary texture. Used for shaders named *EnvMask.fx

(2)
CUSTOMPARAM0 = specular exponent (between 0 and 64)
CUSTOMPARAM1 = reflection intensity (day)
CUSTOMPARAM2 = reflection illumination (night)

(3) Train Simulator "specular map" is an alpha map of the main texture blended with the environment map.