**lod的核心脚本**

Posted on 2013年04月19日 by U3d / [Unity3D脚本/插件](http://www.unitymanual.com/category/script)/被围观 130 次

lod的核心脚本，主要是功能就是通过距离来判断哪些东西隐藏，哪些东西显示。

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 001 | *// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |
| 002 | *// Speed LOD manager, set for 5 LOD* |
| 003 | *// (c) Nikko /Unity3dx.com* |
| 004 | *// All Rights reserved* |
| 005 | *// http://unity3dx.com* |
| 006 | *//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |
| 007 | *// How to use: this script needs to be attached to the LOD prefab.* |
| 008 |  |
| 009 | *//@script ExecuteInEditMode* |
| 010 |  |
| 011 | **private** **enum** \_Nikko\_LOD\_LEVEL { LOD0, LOD1, LOD2, LOD3, LOD4,LOD5} |
| 012 | **public** **var** nlod:**int**; |
| 013 | **public** **var** lodMesh0 : GameObject; |
| 014 | **public** **var** lodMesh1 : GameObject; |
| 015 | **public** **var** lodMesh2 : GameObject; |
| 016 | **public** **var** lodMesh3 : GameObject; |
| 017 | **public** **var** lodMesh4 : GameObject; |
| 018 | **public** **var** mycam : Camera; |
| 019 | **public** **var** delaycounter:**int**; |
| 020 | **public** **var** distancehide:**float**; |
| 021 |  |
| 022 | **public** **var** distanceLOD1 : **float**; |
| 023 | **public** **var** distanceLOD2 : **float**; |
| 024 | **public** **var** distanceLOD3 : **float**; |
| 025 | **public** **var** distanceLOD4 : **float**; |
| 026 |  |
| 027 | **private** **var** currentLOD : \_Nikko\_LOD\_LEVEL= \_Nikko\_LOD\_LEVEL.LOD4; |
| 028 | **private** **var** counter:**int**; |
| 029 |  |
| 030 | function getLODNbr() |
| 031 | { |
| 032 | **return** nlod; |
| 033 | } |
| 034 | function setLODNbr(v:**int**) |
| 035 | { |
| 036 | nlod=v; |
| 037 | } |
| 038 | function getComponentName() |
| 039 | { |
| 040 | **return** "Nikko\_LODManager5"; |
| 041 | } |
| 042 |  |
| 043 |  |
| 044 | *// Library to access internal values of the LOD manager.* |
| 045 | function setLOD(l:**int**) |
| 046 | { |
| 047 | **switch** (l) |
| 048 | { |
| 049 | **case** \_Nikko\_LOD\_LEVEL.LOD0: |
| 050 | lodMesh0.SetActiveRecursively(**true**); |
| 051 | lodMesh1.SetActiveRecursively(**false**); |
| 052 | lodMesh2.SetActiveRecursively(**false**); |
| 053 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 054 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 055 | **break**; |
| 056 | **case** \_Nikko\_LOD\_LEVEL.LOD1: |
| 057 | lodMesh0.SetActiveRecursively(**false**); |
| 058 | lodMesh1.SetActiveRecursively(**true**); |
| 059 | lodMesh2.SetActiveRecursively(**false**); |
| 060 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 061 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 062 | **break**; |
| 063 | **case** \_Nikko\_LOD\_LEVEL.LOD2: |
| 064 | lodMesh0.SetActiveRecursively(**false**); |
| 065 | lodMesh1.SetActiveRecursively(**false**); |
| 066 | lodMesh2.SetActiveRecursively(**true**); |
| 067 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 068 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 069 | **break**; |
| 070 | **case** \_Nikko\_LOD\_LEVEL.LOD3: |
| 071 | lodMesh0.SetActiveRecursively(**false**); |
| 072 | lodMesh1.SetActiveRecursively(**false**); |
| 073 | lodMesh2.SetActiveRecursively(**false**); |
| 074 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**true**); |
| 075 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 076 | **break**; |
| 077 | **case** \_Nikko\_LOD\_LEVEL.LOD4: |
| 078 | lodMesh0.SetActiveRecursively(**false**); |
| 079 | lodMesh1.SetActiveRecursively(**false**); |
| 080 | lodMesh2.SetActiveRecursively(**false**); |
| 081 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 082 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**true**); |
| 083 | **break**; |
| 084 | **case** \_Nikko\_LOD\_LEVEL.LOD5: |
| 085 | lodMesh0.SetActiveRecursively(**false**); |
| 086 | lodMesh1.SetActiveRecursively(**false**); |
| 087 | lodMesh2.SetActiveRecursively(**false**); |
| 088 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 089 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 090 | **break**; |
| 091 | **default**: |
| 092 | lodMesh0.SetActiveRecursively(**false**); |
| 093 | lodMesh1.SetActiveRecursively(**false**); |
| 094 | lodMesh2.SetActiveRecursively(**false**); |
| 095 | **if** (nlod>3) lodMesh3.SetActiveRecursively(**false**); |
| 096 | **if** (nlod>4) lodMesh4.SetActiveRecursively(**false**); |
| 097 | } |
| 098 | } |
| 099 |  |
| 100 | *// Library to access internal values of the LOD manager.* |
| 101 | function setDistanceLOD(lod:**int**,dist:**float**) |
| 102 | { |
| 103 | **switch** (lod) |
| 104 | { |
| 105 | **case** \_Nikko\_LOD\_LEVEL.LOD0: |
| 106 | distanceLOD0=dist; |
| 107 | **break**; |
| 108 | **case** \_Nikko\_LOD\_LEVEL.LOD1: |
| 109 | distanceLOD1=dist; |
| 110 | **break**; |
| 111 | **case** \_Nikko\_LOD\_LEVEL.LOD2: |
| 112 | distanceLOD2=dist; |
| 113 | **break**; |
| 114 | **case** \_Nikko\_LOD\_LEVEL.LOD3: |
| 115 | distanceLOD3=dist; |
| 116 | **break**; |
| 117 | **case** \_Nikko\_LOD\_LEVEL.LOD4: |
| 118 | distanceLOD4=dist; |
| 119 | **break**; |
| 120 | **default**: |
| 121 | Debug.LogError("SetDistanceLOD: Wrong LOD value");return; |
| 122 |  |
| 123 | } |
| 124 | } |
| 125 |  |
| 126 | *// Change the time when this LOD script will execute.* |
| 127 | function SetCounter(c:**int**) |
| 128 | { |
| 129 | counter=c; |
| 130 | } |
| 131 |  |
| 132 |  |
| 133 | function Start() |
| 134 | { |
| 135 | *// spread the autoupdate within the range of the delay counter variable so that each LOD check itself every x number of frames.* |
| 136 | counter=Random.Range(1, delaycounter); |
| 137 | *// check if some object are too far, if yes, disable them* |
| 138 | **var** distanceFromObject : **float** = Vector3.Distance(transform.position, mycam.transform.position); |
| 139 | **if** (distanceFromObject > distancehide) |
| 140 | { |
| 141 | **switch** (nlod) *// each LOD (3,4,5) has to be done separately* |
| 142 | { |
| 143 | **case** 3: |
| 144 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD3; |
| 145 | lodMesh0.SetActiveRecursively(**false**); |
| 146 | lodMesh1.SetActiveRecursively(**false**); |
| 147 | lodMesh2.SetActiveRecursively(**false**); |
| 148 | **break**; |
| 149 | **case** 4: |
| 150 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD4; |
| 151 | lodMesh0.SetActiveRecursively(**false**); |
| 152 | lodMesh1.SetActiveRecursively(**false**); |
| 153 | lodMesh2.SetActiveRecursively(**false**); |
| 154 | lodMesh3.SetActiveRecursively(**false**); |
| 155 | **break**; |
| 156 | **case** 5: |
| 157 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD5; |
| 158 | lodMesh0.SetActiveRecursively(**false**); |
| 159 | lodMesh1.SetActiveRecursively(**false**); |
| 160 | lodMesh2.SetActiveRecursively(**false**); |
| 161 | lodMesh3.SetActiveRecursively(**false**); |
| 162 | lodMesh4.SetActiveRecursively(**false**); |
| 163 | **break**; |
| 164 | } |
| 165 | } |
| 166 | } |
| 167 |  |
| 168 | function LateUpdate() |
| 169 | { |
| 170 |  |
| 171 | */\*if (currentLOD == \_Nikko\_LOD\_LEVEL.LOD0)* |
| 172 | *{* |
| 173 | *// we are on the closest LOD* |
| 174 | *// place here your real time effect, for your closest objects, like wind, movements, animations.* |
| 175 | *}* |
| 176 | *\*/* |
| 177 | counter=counter-1; |
| 178 | *// check that we are at the time we need to check ourself and maybe change our LOD* |
| 179 | **if** (counter<0) |
| 180 | { |
| 181 | counter=delaycounter; |
| 182 | *// quick method to calculate the distance with the camera.* |
| 183 | **var** distanceFromObject : **float** = Vector3.Distance(transform.position, mycam.transform.position); |
| 184 |  |
| 185 | **switch** (nlod) *// each LOD (3,4,5) has to be done separately* |
| 186 | { |
| 187 |  |
| 188 | *//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LOD 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |
| 189 | **case** 3: |
| 190 | *// LOD0 is set here* |
| 191 | **if** (distanceFromObject < distanceLOD1 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD0) |
| 192 | { |
| 193 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD0; |
| 194 | lodMesh0.SetActiveRecursively(**true**); |
| 195 | lodMesh1.SetActiveRecursively(**false**); |
| 196 | lodMesh2.SetActiveRecursively(**false**); |
| 197 | **return**; |
| 198 |  |
| 199 |  |
| 200 | } |
| 201 | *// LOD 1 is set here* |
| 202 | **else** **if** (distanceFromObject >= distanceLOD1 && distanceFromObject < distanceLOD2 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD1) |
| 203 | { |
| 204 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD1; |
| 205 | lodMesh0.SetActiveRecursively(**false**); |
| 206 | lodMesh1.SetActiveRecursively(**true**); |
| 207 | lodMesh2.SetActiveRecursively(**false**); |
| 208 | **return**; |
| 209 |  |
| 210 | } |
| 211 | *// LOD 2 is set here.* |
| 212 | **else** **if** (distanceFromObject >= distanceLOD2 && distanceFromObject < distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD2) |
| 213 | { |
| 214 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD2; |
| 215 | lodMesh0.SetActiveRecursively(**false**); |
| 216 | lodMesh1.SetActiveRecursively(**false**); |
| 217 | lodMesh2.SetActiveRecursively(**true**); |
| 218 | **return**; |
| 219 | } |
| 220 | **else** **if** (distanceFromObject > distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD5) |
| 221 | { |
| 222 | *// again if we are here, this object has gone too far: disable it!* |
| 223 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD3; |
| 224 | lodMesh0.SetActiveRecursively(**false**); |
| 225 | lodMesh1. (**false**); |
| 226 | lodMesh2.SetActiveRecursively(**false**); |
| 227 | **return**; |
| 228 | } |
| 229 | **break**; |
| 230 | *//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LOD 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |
| 231 | **case** 4: |
| 232 | *// LOD0 is set here* |
| 233 | **if** (distanceFromObject < distanceLOD1 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD0) |
| 234 | { |
| 235 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD0; |
| 236 | lodMesh0.SetActiveRecursively(**true**); |
| 237 | lodMesh1.SetActiveRecursively(**false**); |
| 238 | lodMesh2.SetActiveRecursively(**false**); |
| 239 | lodMesh3.SetActiveRecursively(**false**); |
| 240 | **return**; |
| 241 |  |
| 242 |  |
| 243 | } |
| 244 | *// LOD 1 is set here* |
| 245 | **else** **if** (distanceFromObject >= distanceLOD1 && distanceFromObject < distanceLOD2 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD1) |
| 246 | { |
| 247 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD1; |
| 248 | lodMesh0.SetActiveRecursively(**false**); |
| 249 | lodMesh1.SetActiveRecursively(**true**); |
| 250 | lodMesh2.SetActiveRecursively(**false**); |
| 251 | lodMesh3.SetActiveRecursively(**false**); |
| 252 | **return**; |
| 253 |  |
| 254 | } |
| 255 | *// LOD 2 is set here.* |
| 256 | **else** **if** (distanceFromObject >= distanceLOD2 && distanceFromObject < distanceLOD3 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD2) |
| 257 | { |
| 258 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD2; |
| 259 | lodMesh0.SetActiveRecursively(**false**); |
| 260 | lodMesh1.SetActiveRecursively(**false**); |
| 261 | lodMesh2.SetActiveRecursively(**true**); |
| 262 | lodMesh3.SetActiveRecursively(**false**); |
| 263 | **return**; |
| 264 | } |
| 265 | *// LOD 3 is set here.* |
| 266 | **else** **if** (distanceFromObject >= distanceLOD3 && distanceFromObject < distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD3) |
| 267 | { |
| 268 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD3; |
| 269 | lodMesh0.SetActiveRecursively(**false**); |
| 270 | lodMesh1.SetActiveRecursively(**false**); |
| 271 | lodMesh2.SetActiveRecursively(**false**); |
| 272 | lodMesh3.SetActiveRecursively(**true**); |
| 273 |  |
| 274 | **return**; |
| 275 | } |
| 276 | **else** **if** (distanceFromObject > distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD5) |
| 277 | { |
| 278 | *// again if we are here, this object has gone too far: disable it!* |
| 279 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD4; |
| 280 | lodMesh0.SetActiveRecursively(**false**); |
| 281 | lodMesh1.SetActiveRecursively(**false**); |
| 282 | lodMesh2.SetActiveRecursively(**false**); |
| 283 | lodMesh3.SetActiveRecursively(**false**); |
| 284 | **return**; |
| 285 | } |
| 286 | **break**; |
| 287 |  |
| 288 | *//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LOD 5 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |
| 289 | **case** 5: |
| 290 | *// LOD0 is set here* |
| 291 | **if** (distanceFromObject < distanceLOD1 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD0) |
| 292 | { |
| 293 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD0; |
| 294 | lodMesh0.SetActiveRecursively(**true**); |
| 295 | lodMesh1.SetActiveRecursively(**false**); |
| 296 | lodMesh2.SetActiveRecursively(**false**); |
| 297 | lodMesh3.SetActiveRecursively(**false**); |
| 298 | lodMesh4.SetActiveRecursively(**false**); |
| 299 | **return**; |
| 300 |  |
| 301 |  |
| 302 | } |
| 303 | *// LOD 1 is set here* |
| 304 | **else** **if** (distanceFromObject >= distanceLOD1 && distanceFromObject < distanceLOD2 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD1) |
| 305 | { |
| 306 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD1; |
| 307 | lodMesh0.SetActiveRecursively(**false**); |
| 308 | lodMesh1.SetActiveRecursively(**true**); |
| 309 | lodMesh2.SetActiveRecursively(**false**); |
| 310 | lodMesh3.SetActiveRecursively(**false**); |
| 311 | lodMesh4.SetActiveRecursively(**false**); |
| 312 | **return**; |
| 313 |  |
| 314 | } |
| 315 | *// LOD 2 is set here.* |
| 316 | **else** **if** (distanceFromObject >= distanceLOD2 && distanceFromObject < distanceLOD3 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD2) |
| 317 | { |
| 318 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD2; |
| 319 | lodMesh0.SetActiveRecursively(**false**); |
| 320 | lodMesh1.SetActiveRecursively(**false**); |
| 321 | lodMesh2.SetActiveRecursively(**true**); |
| 322 | lodMesh3.SetActiveRecursively(**false**); |
| 323 | lodMesh4.SetActiveRecursively(**false**); |
| 324 | **return**; |
| 325 | } |
| 326 | *// LOD 3 is set here.* |
| 327 | **else** **if** (distanceFromObject >= distanceLOD3 && distanceFromObject < distanceLOD4 && currentLOD != \_Nikko\_LOD\_LEVEL.LOD3) |
| 328 | { |
| 329 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD3; |
| 330 | lodMesh0.SetActiveRecursively(**false**); |
| 331 | lodMesh1.SetActiveRecursively(**false**); |
| 332 | lodMesh2.SetActiveRecursively(**false**); |
| 333 | lodMesh3.SetActiveRecursively(**true**); |
| 334 | lodMesh4.SetActiveRecursively(**false**); |
| 335 | **return**; |
| 336 | } |
| 337 | *// LOD 4 is set here.* |
| 338 | **else** **if** (distanceFromObject >= distanceLOD4 && distanceFromObject < distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD4) |
| 339 | { |
| 340 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD4; |
| 341 | lodMesh0.SetActiveRecursively(**false**); |
| 342 | lodMesh1.SetActiveRecursively(**false**); |
| 343 | lodMesh2.SetActiveRecursively(**false**); |
| 344 | lodMesh3.SetActiveRecursively(**true**); |
| 345 | lodMesh4.SetActiveRecursively(**false**); |
| 346 | **return**; |
| 347 | } |
| 348 | **else** **if** (distanceFromObject > distancehide && currentLOD != \_Nikko\_LOD\_LEVEL.LOD5) |
| 349 | { |
| 350 | *// again if we are here, this object has gone too far: disable it!* |
| 351 | currentLOD = \_Nikko\_LOD\_LEVEL.LOD5; |
| 352 | lodMesh0.SetActiveRecursively(**false**); |
| 353 | lodMesh1.SetActiveRecursively(**false**); |
| 354 | lodMesh2.SetActiveRecursively(**false**); |
| 355 | lodMesh3.SetActiveRecursively(**false**); |
| 356 | lodMesh4.SetActiveRecursively(**false**); |
| 357 | **return**; |
| 358 | } |
| 359 | **break**; |
| 360 |  |
| 361 | } |
| 362 |  |
| 363 | } |
| 364 |  |
| 365 |  |
| 366 | } |