

Dalam mewujudkan fitur gerakan *cycle* dalam ekor Dhemyth agar terus bergerak, digunakan *expression editor* dalam perancangan *rigging* pada karakter hantu terutama Dhemyth. Penggunaan *expression* ini akan berguna dalam menciptakan gerakan berulang tersebut tanpa harus menambahkan *keyframe* pada aplikasi Maya untuk menciptakan animasi *cycle*.

2. STUDI LITERATUR

Ada beberapa landasan teori yang akan digunakan pada skripsi ini, diantaranya landasan teori mengenai *rigging* dengan bantuan *expression editor*, kemudian sebagai tambahan akan ada teori-teori mengenai *body rigging* secara keseluruhan terutama pada karakter hantu.

2.1. LANDASAN TEORI PENCIPTAAN

1. Teori utama yang digunakan adalah mengenai penggunaan *expression editor* dalam proses pembentukan *rigging* serta bagaimana pemograman *expression* dapat membantu animator dalam menyesuaikan bentuk animasi hingga bisa menghasilkan *rig* karakter yang efisien dengan bantuan *expression editor*.
2. Teori pendukung dalam tulisan ini adalah bagaimana sistem gerak pada 3D atau *rigging* di implementasikan dalam animasi 3D. Teori pendukung yang digunakan adalah mengenai penggunaan sistem *rigging* itu sendiri serta bagaimana peran seorang *rigging artist* atau *rigger* dalam keseluruhan proses produksi animasi.

2.2. *Expression editor*

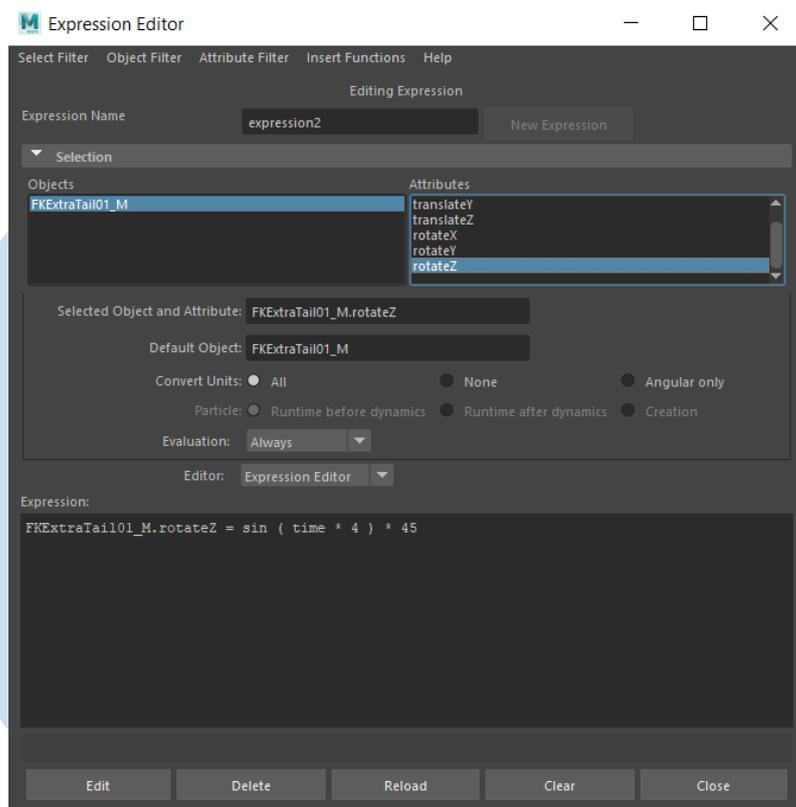
Dalam *website* panduan manual aplikasi, Maya memiliki bahasa-bahasa pemograman yang bisa digunakan untuk keperluan pembuatan produksi, bahasa pemograman ini lebih sering digunakan oleh *technical artist* dan *rigging artist* karena mereka yang lebih dekat dalam urusan teknis. Bahasa pemograman yang biasa digunakan adalah Python, MEL, dan *expression*. MEL merupakan bahasa pemograman dasar yang ada pada Maya, kebanyakan *tools*

ataupun alat yang sudah ada di Maya menggunakan bahasa MEL, seperti memunculkan bentuk mesh dasar, *cut*, *bevel*, dan *tools* sejenisnya.

Kemudian *expression* berperan dalam mengatur *attribute* yang ada pada tiap objek 3D, *expression* dapat membuat sebuah objek memiliki gerakan tanpa harus menambahkan *keyframe* pada Maya, karena itulah dengan *expression* seorang *rigger* dapat menciptakan sistem otomatisasi gerakan tertentu untuk sebuah objek 3D. proses dengan menggunakan bahasa pemrograman ini disebut sebagai *scripting*. (Alias, 2003)

Menurut Khieu Van Bang (2003) *expression* berfungsi dalam memberikan pergerakan pada objek berdasarkan angka-angka yang ditentukan, *expression* juga dapat menggunakan beberapa *command* yang ada pada MEL. Bagaimanapun kekurangan dari *expression* adalah *expression* tidak akan bisa berfungsi apabila pada objek yang ditentukan sudah memiliki *keyframe* sebelumnya, hal ini akan membuat error pada sistem Maya. Beberapa komando yang tidak bisa dilakukan *expression* diantaranya seperti menambah atau menghapus objek dan menyambungkan atau memutuskan hubungan antara *attribute*. (Bang, 2003)

Dalam Autodesk Support (2018) *expression* dijelaskan sebagai instruksi yang diberikan dari pengguna Maya ke aplikasi untuk mengendalikan *attribute* dari satu waktu ke waktu lain atau bisa disebut juga *frame*. Control dari *attribute* yang dimaksud seperti *translateX*, *rotateY*, *visibility*, atau bahkan *attribute* tertentu bisa ditambahkan sendiri oleh pengguna.



Gambar 2.1 Tampilan *expression editor* pada Maya
(sumber: dokumentasi pribadi)

Beberapa fungsi dari tampilan *expression editor* tersebut diantaranya pada bagian kiri atas merupakan objek yang dipilih untuk diberikan *expression*, kemudian pada bagian kanan atas merupakan *attributes* yang dipilih untuk menentukan sumbu mana yang akan digerakan atau diputar, lalu pada bagian tengah merupakan indikasi bagian objek serta *attribute* yang sedang dipilih sehingga *rigger* dapat mengetahui bagain mana yang sedang di *select* saat akan melakukan *scripting*. Terakhir pada bagian bawah merupakan tempat menuliskan *script expression*. Pada gambar 2.1 *script* yang digunakan merupakan *script* untuk melakukan *loop animation*.

2.3. Rigging

Arshad dan Manaf (2019), dalam *International Journal of Recent Technology and Engineering (IJRTE)* ISSN: 2277-3878, Volume-8 Issue-

3, menjelaskan bahwa *rigging* atau bisa disebut sebagai *bone setup* merupakan definisi dari pembuatan sistem kerangka yang dimasukkan kedalam karakter 3D, kerangka ini bisa disebut juga sebagai tulang atau *joint* dalam aplikasi 3D. Saat *modelling artist* telah selesai membuat *model* 3D maka untuk dapat membuat *model* 3D tersebut dapat bergerak secara fleksibel membutuhkan kerangka didalamnya, layaknya manusia yang membutuhkan kerangka tulang untuk dapat bergerak sesuai sendir-sendinya. Tugas utama dari seorang *rigging artist* adalah melakukan penciptaan, pengembangan, sampai modifikasi pada sistem kerangka untuk karakter, kemudian setelah selesai dengan penempatan kerangka berikutnya adalah penyatuan antara *mesh 3D* dengan kerangka-kerangka atau *joint* yang disebut *skinning*. *Skinning* adalah proses penyatuan antara *joint* dengan karakter atau *mesh 3D*, sehingga kerangka dan *controller* yang telah dibuat dapat berfungsi dan dapat digerakkan.

Arshad dan Manaf (2019) menuliskan ada beberapa tipe dari sistem *rigging* tergantung dari bagaimana yang dibutuhkan oleh produksi. Salah satu tipenya adalah *traditional bone object*, biped, dan *Character Animation Toolkit (CAT)*. *Traditional bone object* merupakan metode *rigging* dimana seorang *rigger* memasang satu-satu setiap *joint* dalam 3D *mesh*. Kemudian untuk biped dan CAT ini biasanya sudah berbentuk *template* atau bentuk kerangka secara keseluruhan sudah disediakan aplikasi 3D, sehingga seorang *rigger* bisa fokus pada *skinning* dan memberi beberapa modifikasi apabila memang dibutuhkan (hlm. 4138 - 4139).

Dalam produksi animasi “*Scary?*” sistem *rigging* menggunakan biped, namun di aplikasi Maya biped tidak secara langsung disediakan sehingga harus menggunakan *plug-in* tambahan yaitu “*Advanced Skeleton (AS)*”. Dengan menggunakan AS proses *rigging* akan sangat menghemat waktu. Tidak hanya itu *rigger* juga nantinya akan memiliki lebih banyak waktu untuk memberikan fitur-fitur tambahan pada *rig* sampai memodifikasinya sehingga dapat *rig* dapat berfungsi dengan lebih praktis dan efisien ketika digunakan.