A-maize-ing Dynamic Cornfield Toolkit

Ian J. Coony & Dale Mayeda
Walt Disney Feature Animation
Ian.Coony@disney.com   Dale.Mayeda@disney.com

Introduction

For Walt Disney Feature Animation's, CHICKEN LITTLE, the EFX team was faced with a large number of vast, dynamic cornfield shots. These shots provided for some interesting challenges as they needed to be able to react to characters and dynamic forces such as wind, turbulence, and gravity. To further complicate matters, another requirement was to allow for the corn to be chopped by blade wielding aliens into predetermined crop circles. We realized early on that the system had to be easy enough for various departments to work with, yet flexible enough to get under the hood, as needed. Finally, any work created had to travel downstream in a compact form, without overwhelming other departments or bringing the pipeline to a grinding halt.

Multiple Systems "Glued" Together

To solve these challenges we took a novel approach by combining several separate existing systems, both off the shelf and proprietary, into one coherent whole. We utilized our in house user interface creation tool as a front end to this system to create and control **Maya Paint Effects** (PFX) corn brushes. This system also integrated our custom **particle instance replacement** software to populate the bulk of the cornfield. **Softbodies**, in conjunction with **dynamics,** were used as control guides for any PFX cornstalks that needed to interact with characters. By leveraging the power of **Maya Embedded Language** (MEL) scripting, it was possible to "glue" all the various systems in place and to automate time consuming tasks as much as possible.

Figure 1 – An "On Demand" Dynamic Cornfield © Disney

A Walk Through the Cornfield

The setup process began in the Layout Department (LO) where **corn placement grids** could be adjusted according to the needs of the shot. Each vertex in a grid would act as a proxy for a cornstalk. Using MEL tagging tools, the EFX department could define which stalks needed to interact with characters on a grid by grid basis – taking care to use dynamic corn (PFX) sparingly and only where necessary, for efficiency sake. All other grids could then be used as a template for placing particles into rows. The particles were then fed into a particle cache and instance replacement program. Any grids outside of the camera view were tagged as having "no corn" and ignored to save on render times. Once grid tagging was complete, the cornfield was then **procedurally generated** based on the user input. The entire building process was completely scripted in MEL and only took a few minutes to complete – a huge time saver.

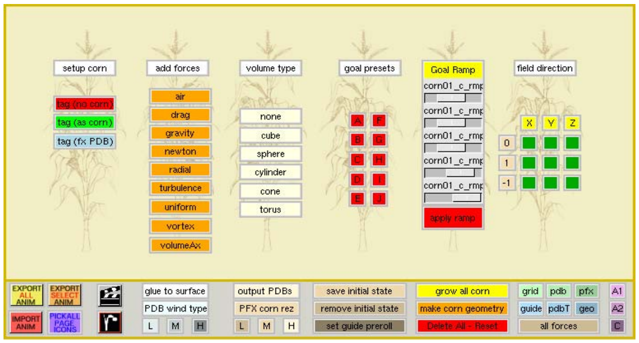

Figure 2 – The Cornfield User Interface (UI) © Disney

Dynamic, Rigged and "Canned" Corn

This system could deal with three types of corn stalks: Instances, PFX and "Hero" rigged corn for specific character interactions which were deemed difficult to achieve with dynamics.

Instanced corn consisted of canned cycles of animation which were stored in **RIB archives** at various user selectable wind speeds. At render time, the particles were replaced with dynamically spawned geometry using a custom **Renderman DSO**. This DSO would replace each particle with a RIB archive cornstalk. The cornstalks were also rotate jittered and the animated frames offset for a more organic feel which helped to prevent twinning. Feedback was available inside Maya by converting the PFX cornstalk brushes to polygons. This was done to help visualize dynamic stalks which would interact with characters. Stalks which were to be chopped were designated by setting individual per-particle states using a special "**proxypaint**" brush designed specifically for chopping and crop circle creation. The system also allowed for grayscale texture maps as an input which could then act as a placement tool for the cornstalks.

The PFX cornstalks were created using a custom designed brush. As each stalk was "grown" via MEL using this brush, it was possible to also generate and map the PFX to a softbody **guide curve**, then hook up all dynamic relationships "on the fly". Dynamic forces could be constrained to characters to get motion for free. As a character moved through the PFX stalks, the softbody guides were affected by the forces which would then perturb any surrounding PFX cornstalks. The softbody guides had **goal weights** assigned to each vertex in the guide curve. These goal weights could be edited through the cornfield UI to vary the stiffness of the cornstalks as needed.

Once animation was approved, the PFX and instanced geometry were then exported as RIB archives to pass downstream. The scene would be rendered as a whole with the instanced corn blending seamlessly into the PFX dynamically animated corn.

Conclusion

Creatively connecting existing toolsets and scripting time consuming setups made it possible to develop a sophisticated, efficient, yet flexible UI driven system - allowing us to finish a complicated sequence of shots under budget and in record time.