Venomous Cattle for *Australia*

Carsten Kolve Dan Bethell Rogier Fransen Johannes Saam Premamurti Paetsch
Rising Sun Pictures

Figure 1: Cattle extension. ©20th Century Fox. All rights reserved.

In Baz Luhrmann's *Australia* the main protagonists drive a massive herd of cattle through the Australian outback. To extend the on-set cattle herd with up to 5000 digital bovines we developed a crowd system based on a *Autodesk Maya* integration of our *Venom* infrastructure. This sketch will give an overview of the crowd workflow and how it utilized Venom features to arrange, simulate, reuse and render animated cattle for numerous shots.

1. Layout

The base data structure for a virtual cattle herd is a *crowd container*. This container holds all general and per agent data necessary to generate a crowd. In each working step more data is added until we can render a fully animated herd.

The most important information added at layout stage is a transformation matrix for each agent, indicating the initial world position and orientation. Transforms can be generated via scripting or, as the system is fully integrated into Maya, by sampling data like particle systems, geometry (used for positioning on a ground plane) and shaders (used for densities). A custom *Venom Maya shape node* allows for overall and per agent transform modification. Like any other shape node, deformers can be applied on the data as well.

Full crowd layouts are used for shot blocking purposes, but typically *not* for the initial positioning of a full crowd simulation. Instead the crowd container layout is divided into smaller sub-layouts based on distance to camera and similar performance.

2. Simulation

A per agent skeleton structure is now added to the crowd container which is animated using the *animation library*. This library is composed of interconnectable loops and transitions. Various techniques are utilizing it to produce animation: Simple hand crafted or randomly generated animation lists drive static crowds without much agent interaction. A function that finds the best fitting next set of animations in the library based on a predicted or predetermined path is used in shots that require more sophisticated navigation. Specific crowd behaviour and offset joint animation can also be scripted by accessing the crowd container directly. Blending is only used to smoothen transitions at the start and end of clips.

Collisions between the often tightly packed cattle are resolved using a relaxation technique that uses clusters of spheres to approximate the collision shapes of agents and obstacles. The resulting offset translation and rotation can be applied gradually to the root, but foot sliding is not automatically prevented. Instead the artist controls the balance between amount of intersections and quality of animation.

For performance reasons agent animation is generated in a two-step process: First, only the animation of the root is calculated as it is often sufficient to judge the quality of a simulation. Only if the root animation is successful the remaining joint animation is computed.

3. Crowd Building Blocks

The result of a simulation is never directly used in shot context. Instead crowd containers are cached over a longer period of time (as compared to an average shot length) and then used as *crowd building blocks*. The Crowd-TD assembles the blocks using a browser like interface in the final scene. Crowd caches and their individual components can be transformed in space and time, all operations described in the *Layout* section are usable here, too. Common editing operations like deletion or duplication are also supported.

This approach has multiple benefits: Producing convincing simulation on a big scale is hard and often requires lengthy iterations, focussing on smaller groups in contrast allows for more iterations and a higher quality result. After a small library of building blocks has been created, the ability to reuse them becomes a major time saver – especially when dealing with multiple shots that require a similar crowd performance. On *Australia* only 6 small caches with an average of 300 agents were used to produce all cattle herding shots, even though the crowd shape changed dramatically. The herd can be assembled for the exact needs of the shot, with the ability to remove performances that are not working from the current camera angle while making successful actions more prominent. Assembly work can also be done by a much less technical artist than simulation work.

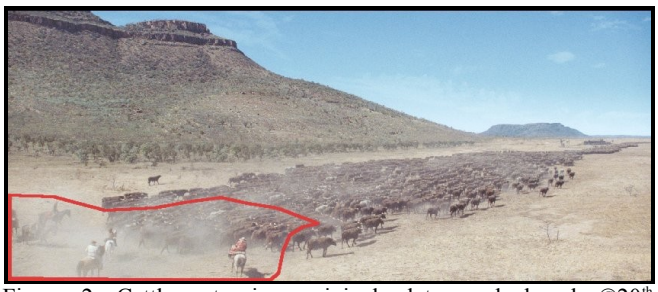

Figure 2: Cattle extension, original plate marked red. ©20th Century Fox. All rights reserved.

4. Rendering

Crowd containers can be rendered in different *contexts* – the most commonly used one being an OpenGL context for preview in Maya and a *3Delight / Renderman* context for high quality rendering.

At the render pre-processing stage for each agent a procedural call is inserted into the rib-stream in conjunction with the agent's bounding box. Then at render-time this procedural uses the skeleton and added model with binding information in the crowd container to skin and render the character on demand.

Crowd containers can be annotated with extra per-agent-data by the Lighting TD and these attributes in turn be utilized by the shader. All variations including colours, textures, displacements, id passes etc. are realized using this technique. Blend shapes controlled in the same way generate different cattle proportions.

Agents have one unique id number used for targeted value assignment independent of production department. For example, the compositor (using the u-id pass) can provide information to a Lighting- or Crowd-TDs on how to change which animal.

5. Outlook

Future work will focus on making the technical, script based workflow accessible to a broader artist base. The current animation system can be enhanced to allow more responsive blending between animation clips (for example using motion-graph techniques) and prevent foot-sliding by incorporating ik solutions.