Improving Crowd Quality Through Interdepartmental Collaboration on *Madagascar 3: Europe's Most Wanted*

Nathaniel Dirksen Justin Fischer Jung-Hyun Kim Kevin Vassey Rob Vogt
DreamWorks Animation

1 Introduction

For *Madagascar 3: Europe's Most Wanted*, DreamWorks Animation set itself the goal of creating crowds which rivaled our main characters in terms of quality of appearance and performance. A key component to our success was the tight collaboration between the Crowd and Character Technical Director departments. Working together, we were able to ensure our crowd characters looked as good as possible, and when problems arose, could solve them either in the Crowd or Character TD domains, whichever turned out to be more appropriate and most efficient.

Our primary advancement to achieve top quality deformations was driving the full animation rigs created by Character TD with the results of Crowds' Massive sims. In previous films, crowds were made up of simplified assets, which would not hold up to the visual standards of our hero characters. However, even once techniques were developed to drive the full character rig with Massive's output, disk space and network bandwidth constraints made it infeasible to bake out large crowds. Instead, Crowds worked with Pipeline Engineering to develop an architecture for Render-Time Deformations (RTD), where character geometry is generated on the fly. This technique also required significant changes to our animation rigs. Character TDs created Multidimensional Rigs, where a character can run in a service mode, interfacing with the RTD server to deliver any model for any pose as requested by the renderer.

High quality deformations allow crowd characters to be pushed closer to camera, where their acting performance is highlighted. Right from the beginning of production, all of our generic characters were designed so that their facial expression would read the same way when the same animation was applied, allowing the Crowds department to cast generic characters with any head variation with no additional overhead. With crowd characters much closer to camera, they also receive more Director feedback. To allow direction of theses performances, we developed a system called Hero Promotion, which makes hero assets from crowd characters. Unlike previous systems that worked with minimal knowledge of the character rig, by integrating required data into each other's workflows, Crowds and Character TD could ensure that the hero promoted animation keys matched both Massive's agent positioning for shot composition, and the original hand-animated cycle for easy animation augmentation.

2 Render-Time Deformations

Similar to many other studios [Ryu and Kanyuk 2007], within the DreamWorks pipeline, all crowd cycles are key-framed by animators. Massive is used to blend these cycles, outputting joint locations that drive our animation rigs, from which models are baked and ultimately rendered. For even mid-sized crowds, the disk-space and network bandwidth requirements make this approach infeasible if full resolution models are used. Baked out geometry consumes about 2 megabytes per-frame per-character. There are several shots which have thousands of characters in the circus scenes. Each shot would use upwards of 500 gigabytes if all character geometry was baked to disk, with several terabytes required to finish certain sequences – an unfeasibly large amount for our production.

Instead, we developed an approach for Render-Time Deformations (RTD), where whenever the renderer determines it needs a piece of a character's geometry, a request is sent to the RTD server. The request specifies the type of rig that it needs to be used, and the animation file that the rig will use to produce geometry. The model is made on the server, and the results are passed back to the renderer.

Figure 1: *A panning crowd shot rendered with RTD, which features thousands of characters, many of them close to camera.*

This technique bypasses the need to store these models on disk; the RTD server requires only the joint positions, animation curves and character variation, decreasing disk usage to one percent of the conventional method. Each animation file is about 20 kilobytes but models produced from it are about 2 megabytes. Running the RTD server on the same box as the renderer provides similar improvements to network bandwidth consumption. Surprisingly, we often found network bandwidth to be the limiting factor rather than disk space. DreamWorks has offsite render-farms, and transferring huge volumes of data to them proved difficult. Network bandwidth between each site is 20 gigabits per second. When there are 1000 characters on screen, the geometry could require 20 gigabytes of space and it would take 8 seconds to transfer them when full network is used for this one frame. If the shot has 100 frames, just transferring data would take about 13 minutes. With motion blur, the amount of data will be doubled, and it would take 26 minutes just for crowd model data to transfer. Even worse, the network pipe for intercontinental transfers is more limited: transferring models for a render in India could take nearly 2 hours. Render-Time Deformations decrease the network load to less than one percent of the time for a full model transfer, which is as important as the improvement to disk usage.

Finally RTD also provides Lighting flexibility in setting level of detail, since models at any requested resolution can be generated on the fly. When crowds are delivered to Lighting, the LOD is determined by a character's screen space. However, Lighting has the ability to change a character's resolution as needed. Since geometry is not baked out for RTD characters, Lighting is able to control geometry resolution themselves, instead of requesting Crowds deliver a new version of the models. Because the RTD server needs to produce models at any resolution, the server may require moderately more memory to hold geometry at different levels of detail for deformation. However, the time cost of rendering a high-polycount character far exceeds the cost of generating low-resolution geom-

https://doi.org/10.1145/2370919.2370922.

etry, so giving this flexibility to Lighting allows them to pick the representation of crowd assets that best suits their workflow and the shot composition.

Figure 2: *ManA's Multidimensional Rig outputs any of 3 bodies, 6 heads, and various wardrobe items for a total of 119,750,400 unique combinations.*

3 Multidimensional Rigs

The development of Render-Time Deformations creates a new set of requirements on the character rig; any geometric variation must be able to execute when models are requested without pulling any data from disk, which would slow down the evaluation. A conventional character must be told before a shot is launched what combination of body, face, hair, and clothing to produce. Loading the character in this way requires about 1 gigabyte of memory and 30 seconds. This method clearly does not scale for a shot with even 1000 characters; that would require 1 terabyte of memory and about 8.5 hours of loading time. There are methods to get around this problem; splitting the shot up, baking models from the farm, or reducing the number of characters. However, none of these address the fundamental problem that one generic character should be able to create any of its millions of model combinations from a single instance.

On previous shows, the Crowds department created a simplified asset setup based on an automated skinning of the character and clothing geometry to the joints output from Massive. This simple setup could load all of its geometric variation simultaneously. However, in practice, the deformation quality was insufficient for characters located close to camera. Though finding a solution purely inside the Crowds department was difficult, working closely with the Character TDs, we were able to develop techniques to allow the full production rig to be used.

Our solution is Multidimensional Rigs, which can be loaded in a stateless context, outputting models for any geometric variation or animation as needed. Now character instantiation occurs only once, and models for the entire crowd can be cheaply generated afterward. Essentially, a single generic character can be reused over and over within a shot; for each member of the crowd, animation is applied to a character, models are requested and returned from the server, and then we repeat the process for each character that will be rendered. To allow this functionality, all potential character body, face, hair and wardrobe variations must be loaded simultaneously.

This character multidimensionality was accomplished via improvements to our rig architecture, allowing changes to the body and face to flow downstream to secondary components, like hair and clothing, that must adjust themselves in response. For example, if we switch to a new head, the hair changes how it warps around the face. To make this happen for one component there are many steps involved.

To set the body into RTD mode, first the correct component level objects must be selected. In a normal animated shot we would select a single component like the 'body-0' object. For RTD we would instead select all of the body objects. These objects contain lists of file paths and in RTD mode the object knows to select an alternate list. This gives us the flexibility of using a slightly different loading structure on the RTD side. We added a special include that allows us to string replace object names as they are instantiated. This solves two issues in a component. The first is to avoid name clashing we can uniquely name objects that are specific to a particular model variation. The second is we can load data that exists on disk once as many times as needed, commonly once for each unique model. The remaining issue is that we now have many versions of each component loaded and the downstream components need a unified input. We create a set of outputs for each component, one for each object that is needed downstream. This output object takes in many models and passes out the proper model based on switches that are set by Crowds.

This process works for the primary components like the body and face but components that are downstream from these need additional information. A component like the hair will have generated data relative to a specific head. For example, binding information would be unique for each head/hair combination. This information needs to be recalled depending on which head is requested. This component needs to load all of the hair variations plus all the data that needs to vary based on an upstream component. These varying inputs are selected based on values set by Crowds, then the proper information is evaluated.

The character can now run on the RTD server. Running as a service, the multidimensional rig can dynamically change both its geometric variation and animated pose as requested by the RTD server, and export any combination of models as requested. Though the time and memory overhead to load up a multidimensional instance of a character is higher than a normal version, you'd find in a shot, this cost must only be incurred once, rather than once per character in the scene. In practice, it required 1.8 gigabytes of memory and 2 minutes of load time, and could therefore easily be run on the same machine as the renderer.

4 Facial Animation Workflow

The RTD system allowed higher quality models to be generated by the Crowds department, but it also exposed some weaknesses in our workflows that were not apparent earlier. One difficult area was facial animation. Previous RTD solutions didn't support high-quality faces, but this was of little concern, since the deformation quality wouldn't allow the characters to be placed prominently on camera anyway. However, with the techniques described above, suddenly our crowd characters could be featured front and center. A solution had to be developed.

For non-RTD techniques which bake models to disk, we already had methods for sending our facial animation through Massive. Each animation control is represented as a blend shape curve, allowing blended animation values to be read from Massive's output, converted into our own animation format, and fed into the character. With some small modifications to the character rig, Crowds could also pipe this animation on the fly to the RTD server.

However, even with this technical solution in place, the animation workflow was still too difficult. Each of our five generic humans has six different heads, all of which are quite different in shape. Animation created for one head would not look the same on another. This isn't a problem in a normal shot workflow, since each character is carefully casted and individually animated. However, for crowds, cycle animators would have to create animation for each head variation, for each cycle – an amount of work infeasible to add to their busy schedules.

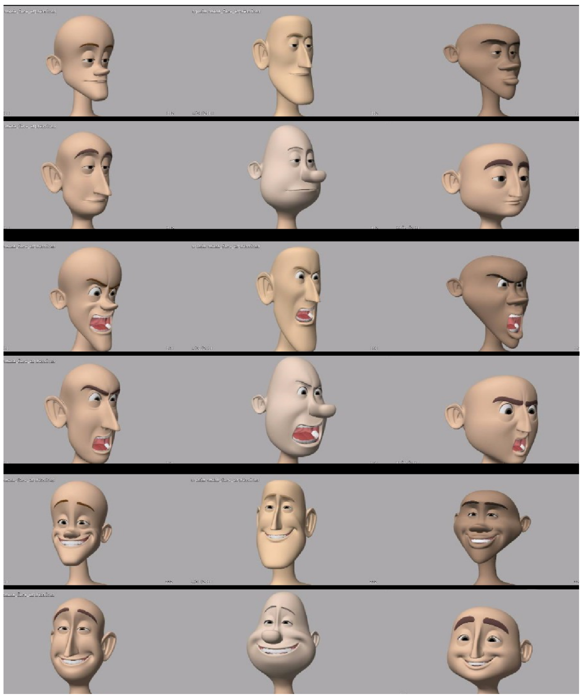

Figure 3: *Identical animation values applied to various heads.*

Fortunately, Crowds recognized early on that this would be a problem, and we were able to adjust the Character TD and Cycle Animation workflows to avoid this extra effort. Even though our faces have 150 animation controls, working with the Animation department, we established 30 controls that could achieve most required facial motion. Character TD's made sure that these 30 controls read the same way for every generic character, so a smile with value of fifty percent would appear to be a half smile on any man or woman's head. Often, the amount of motion to get the same read would be quite different, or controls would mix together in surprising ways, so we built in side-by-side comparisons of generic heads into our facial expression approvals process to ensure that everyone was satisfied with the results. With these 30 controls carefully vetted, Cycle Animators would restrict themselves to this limited set, ensuring that the animation would read the same way, regardless of what generic was cast in the crowd shot.

5 Hero Promotion

With the high-quality deformations allowed by using the full character rig, crowds could be pushed very close to camera. However, this exposed them to a degree of performance direction not typically applied to crowd characters. Using Massive to respond to detailed Director feedback at this level is needlessly difficult – instead through a process call Hero Promotion, we convert the character into a hero asset that the Animation department can update.

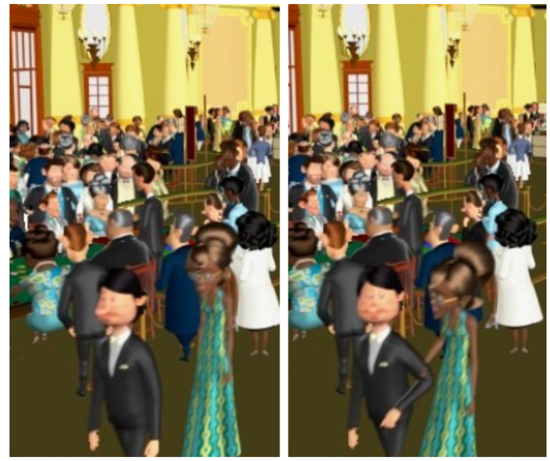

Figure 4: *Example hero promoted shot. Hand animation significantly improves the performance of characters close to camera.*

To allow Animation to work on the hero rig, the output from Massive must first be converted into animation curves which represent the same poses. Previously, with only the joints output from Massive to determine the pose, the Character TDs developed a tool which plotted each joint's position, calculated the rotation and translation values necessary to move the motion system into position using a subset of the controls, and set the appropriate keys in each frame of the animation file. However, even if a curve simplification algorithm is run to simplify the resulting curves, they can still be difficult to work with. The controls used will differ (IK vs. FK, twist vs. orbit, foot rocker vs. foot rotation, etc.), and animation keys associated with a particular key pose or breakdown pose will no longer line up on the same frame.

The Hero Promotion system significantly improves upon plotted animation by re-timing the keys from the original hand-animated cycles to match the sim. This ensures the controls and animation methodology are maintained. However, there are surprising challenges involved with taking Massive's output and creating usable animation curves. Even though we have the animation from the original cycles, it is not trivial to determine which cycles Massive is blending together at any given time. We are able to extract the original cycle IDs and frame ranges to re-time from Massive's output by specially annotating each input cycle with animation curves representing the cycle identification number and a value indicating how far through the action we are. All of the cycle animation curves (including our special annotations) are converted into blend shape curves, so that they are recognized and blended by the Massive crowd sim. Examining how Massive has blended these curves in its output, we can determine for any frame which actions are active, and which parts of them Massive is blending together. We then access the original cycle animation curves, and copy the keys corresponding to the blended parts of the input cycle into the output animation curves. This technique provides key-identical output, as long as Massive is only playing back one action. In the short frame ranges where Massive switches between or blends multiple actions, animation cycles cannot simply be re-timed, and we instead blend evaluated curve values at each frame.

Re-timing keys from the original cycle will not place any globally spaced controls (like the body or IK goals) correctly. Positioning these from Massive's output is difficult, since typical mocap rigs do

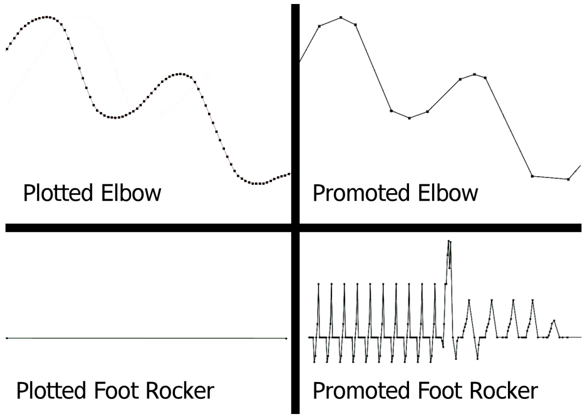

Figure 5: *Plotted elbow bend (upper left) has keys every frame, Hero Promoted version (upper right) has keys only where animators placed them. Foot rocker control (bottom right) is used in Hero Promoted animation, but has motion too difficult to reverse engineer for plotted solutions*

not have a stable reference platform from which to calculate them. We augmented both our Massive and animation rigs with a new joint which tracks the position of the hips, but only has locomotive motion applied to it. This new joint replaces the hips as the root of our hierarchy in Massive, and by examining its output position, as well as the relationship between it, the body, and the body pivot in the corresponding frame of the original cycle, we can drive the body control to place the character correctly. Similar techniques can be used for the IK goals.

Hero Promotion is only successful because it creates assets that are easy for the entire production pipeline to work with. From Animation's standpoint, they seem like any other production character, using rigs identical to those for normal assets. A promoted character starts with a pose that matches Massive's output, so that no additional work is required to match the existing shot composition, yet the animation curves match the original cycle animation as closely as possible. Unlike simply plotting the joint positions and running a curve simplification algorithm, animation created via hero promotion uses identical controls to the original cycle, and ensures keys associated with a given pose will remain on the same frame. These qualities make it much easier for animators to update and enhance the performances.

Once dynamically simulated crowd characters have been hero promoted and animated, the final challenge is delivering those assets to the lighting department for rendering. Our system provides flexibility to the back-end of the pipeline, so that the promoted characters can either be delivered as part of the crowd asset, or as actual hero characters. This allows for the Directors to make performance decisions much later in the process, and not affect the current Lighting schedule.

6 Making Collaboration Work

Every animated feature is a huge collaborative effort between hundreds of people. However, at large studios, it's all too common for departments to become isolated, focused on internal tools and sharing data through established pipelines, rather than creating novel cross-departmental solutions. Fortunately specific steps were taken during the production of *Madagascar 3: Europe's Most Wanted* to allow the innovations described above to be created and implemented successfully.

The creation of appealing human characters received major emphasis from the start of production. This lead to the creation of the "Human Task Force" comprised of various department heads and leads, and proved invaluable not just for brainstorming, but also for gaining consensus when various changes were implemented. For example, the work to implement Multidimensional Rigs on the RTD server fell mostly to Character TD and Crowds. However, the ultimate consumer of the generated models was the Lighting department, and their cooperation was also essential.

The Crowds department Supervisor and Technical Director where also brought onto the show much earlier than normal. By starting up with most of the front-end departments, Crowds could work more closely with Character TD than normally possible. Neither RTD nor Hero Promotion would have been possible using the existing deliverables provided by each department. On typical shows, where Crowds starts with the back-end departments, it's often too late to develop novel solutions and incorporate them into existing character rigs. With an early Crowds start, deliverables could be added or adjusted (along with any associated features) right from the beginning of actual production, ensuring that both departments could deliver whatever information the other needed to best accomplish the task at hand.

Having Crowds start on the show early with Character TD would have been useless without good communication. Though it seems obvious, getting members together from the different departments for face-to-face updates makes a huge difference. Putting the people actually implementing the solutions in direct contact allows for far more efficient brainstorming of solutions than if all communication goes via the Department Heads or (even worse) Trouble Tickets.

7 Conclusion

Based on our success during *Madagascar 3: Europe's Most Wanted*, the Crowds and Character TD departments are continuing to work closely together on future films, to ensure that we continue to improve our combined workflows.

As future shows attempt to increase the depth and variety of crowds, there will certainly be new problems to solve. Single computers have a limited amount of CPU and memory resources, and will be unable to handle the renderer and RTD server running concurrently on the same machine. We are exploring cloud-based Render Time Deformations as the long term goal for this system, and are continuing to improve it on future shows. Similarly, Hero Promotion is being improved, so that the globally spaced controls in a character can promote themselves, allowing the technique to be applied as easily to crowds of non-bipedal creatures as it is to humans.

We have also started applying many of the techniques outlined here in collaborations with other departments outside of CharacterTD and Crowds. Multidimensional Rigs are useful in many contexts beyond the RTD server, from providing an easy way to test large selections of generic characters, to providing the Layout department tools for casting characters in their shots. Having a set of facial controls guaranteed to map cleanly between characters turns out to also be useful in establishing an animation library. Each department clearly has expertise valuable beyond itself, if production allows the time for the collaboration to discover it.

References

RYU, D., AND KANYUK, P., 2007. Rivers of Rodents: An Animation-Centric Crowds Pipeline for "Ratatouille". ACM SIGGRAPH 2007 sketches, August.