Course # 13, SIGGRAPH 2007
San Diego, CA USA


WRITTEN & PRESENTED BY:

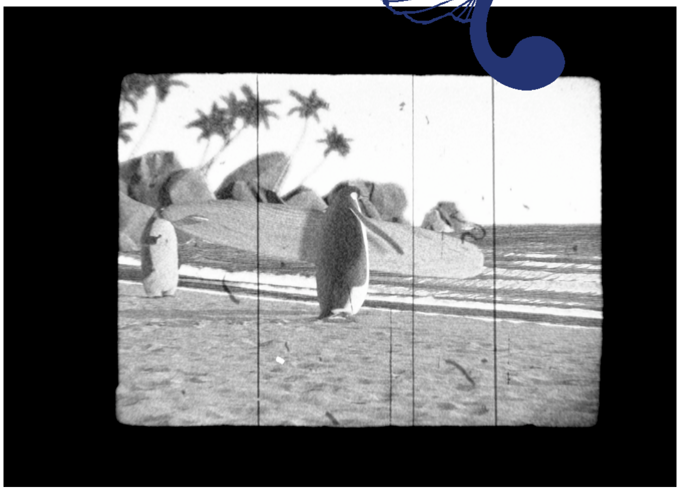
Rob Bredow
David Schaub
Daniel Kramer
Matthew Hausman
Danny Dimian
R. Stirling Duguid

Sony Pictures Imageworks, Culver City, CA, USA

# About these notes

These course notes cover new tools, processes and pipelines that were developed to make *Surf's Up* a reality. These course notes are intended to serve as a reference for some of the more unique aspects of the *Surf's Up* production pipeline. Many aspects of the production of the film share a lot in common with other CG feature films and have been intentionally omitted.

This document is best viewed in "Facing Pages" mode if viewing digitally.




- Zeke Topanga

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

# INTRODUCTION

# 1.1 Abstract


The CG animated documentary *Surf's Up* called for unique production techniques to be leveraged throughout the creation of the film. This half day course presents an in-depth look at several of the key aspects of the production:


- Integration of a live action hand-held camera into the virtual world to allow the movie to be "shot" in a documentary style.
- Development of the wave system and its integration and use by all departments on the film during the production of the movie.
- Animation techniques and choices made to support the documentary nature of the film.
- Novel effects animation techniques created to support the large number of breaking waves featured in the film.
- Look development and lighting techniques leveraged to render the realistic breaking waves as seen in the film.


# 1.2 About the Film

*Surf's Up* is an animated documentary that delves behind the scenes of the high-octane world of competitive surfing. The film profiles teenage Rockhopper penguin Cody Maverick (Shia LaBeouf), an up-and-coming surfer, as he enters his first pro competition. Followed by a camera crew to document his experiences, Cody leaves his family and home in Shiverpool, Antarctica to travel to Pen Gu Island for the Big Z Memorial Surf Off. Along the way, Cody meets surf nut Chicken Joe (Jon Heder), famous surf promoter Reggie Belafonte (James Woods), surf talent scout Mikey Abromowitz (Mario Cantone), and spirited lifeguard Lani Aliikai (Zooey Deschanel), all of whom recognize Cody's passion for surfing. Cody believes that winning will bring him the admiration and respect he desires, but when he unexpectedly comes face-to-face with a washed-up old surfer named Geek (Jeff Bridges), Cody begins to find his own way, and discovers that a true winner isn't always the one who comes in first.

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

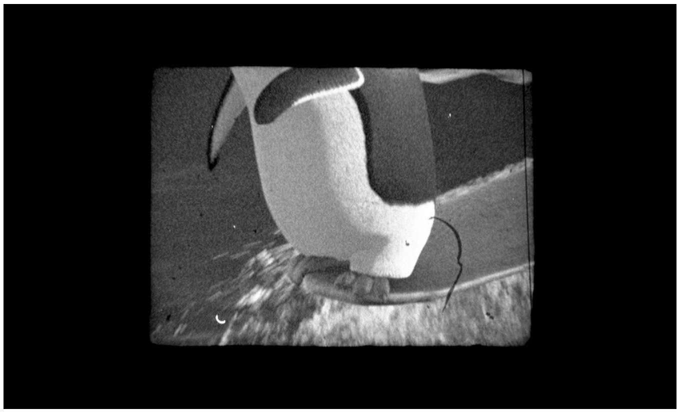
# MAKING AN ANIMATED DOCUMENTARY

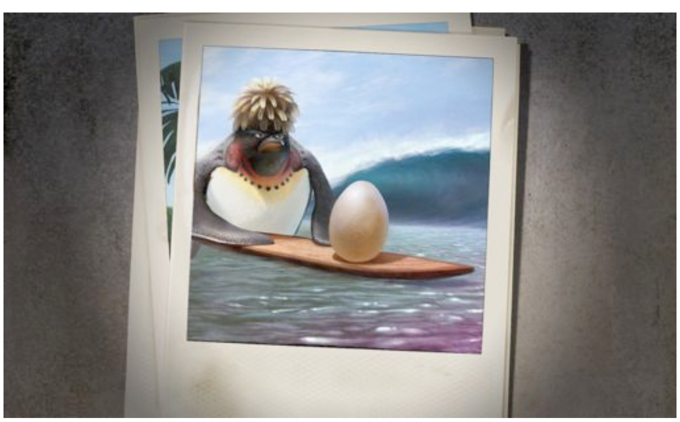
# 2.1 The Premise

The premise of shooting *Surf's Up* as an animated documentary provided a welcome opportunity to use new techniques to bring a fresh look to an animated feature. The idea that the film was shot by a medium-budget documentary crew informed every area of the production; from the choices made with the camera moves to the lighting setups used throughout the film.

Some of the most important items that lead to the "documentary" feel of the movie were identified after studying films like *Step Into Liquid* (2003), *The Endless Summer* (1966) and *Second Thoughts* (2004). They all contained "found footage," lots of hand-held cameras and extensively used slow motion and other time manipulation techniques.

# 2.2 FOUND FOOTAGE





Like any good documentary, *Surf's Up* contains footage from various sources that is being used throughout the film. This footage required an authentic "archive" feel to help inform the audience that the film has a real history (albeit fabricated). The archive footage was used heavily in the first act of the film to help establish the world and the genre of the movie but was also referenced in the second and third acts to refer back to historical events that played an important role in the story.

The filmmakers also "found" footage from modern day sources and leveraged shots from SPEN (the "Sports Penguin Entertainment Network"). This footage was played as if it was shot on high definition video so it had a distinctive sharp look, color artifacts, and over-the-top motion graphics left over from being shot by a sports network always looking for that punchy feel for their viewers.

The primary types of footage that Imageworks conveyed during the film included:

- 1920s film
- 1950s film
- 1970s film
- 1970s still photographs (Polaroid, instamatic)
- Circa 2000 1st unit photography
- Circa 2000 Video

# 2.2 Found Footage cont'd

A large assortment of elements went into creating the various looks including:

- Grain
- Film discoloration
- Scratches
- Negative dirt (white)
- Print dirt (black)
- Small hairs
- Fuzzy projection mask
- Light leaks (yellow/orange glow on right edge of screen)
- Roll outs
- Tape splices
- Lens vignetting
- Projection vignetting
- Projection strobing
- Lens color aberrations
- Negative/Clear leader marker lines
- China marker lines (for process)
- Video scanlines/interlace/de-interlaced method
- Motion graphics

The various film and video looks used throughout the film were implemented as compositing macros that were applied to degrade the full color, high resolution images generated by the render and created the look appropriate for the shot or sequence. The macros were developed by the Compositing Supervisor and other lead lighting/compositing artists and each of the individual lighting artists adjusted the effects as needed for their particular shot and sequence.


KELLY SLATER
Pro Surfer
SPEN


SPEN
TANK "THE SHREDDER" EVANS
Nine Time Champion

## 2.3 The Live-Action Camera

Any low or medium budget documentary takes advantage of the speed and convenience of a handheld camera throughout production. The filmmakers and Visual Effects Supervisor desired to convey this same feeling throughout *Surf's Up*.

# 2.3.1 The HandyCam System

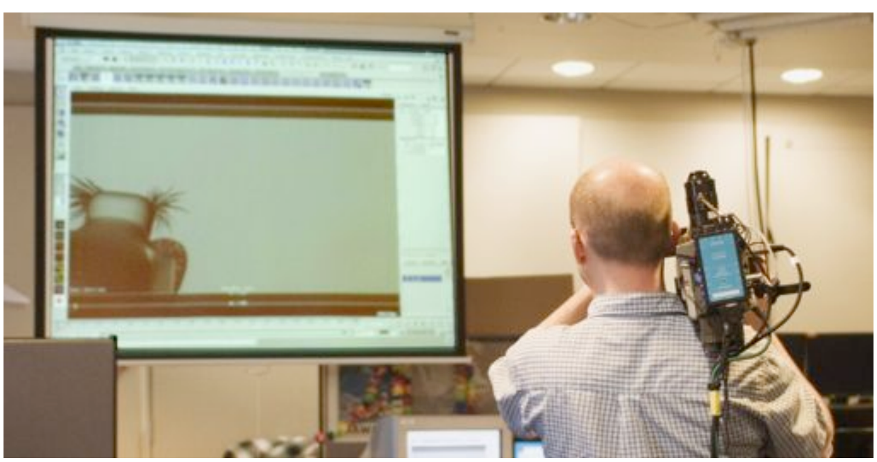
Sony Pictures Imageworks built a new live-action camera system for the film, dubbed the "HandyCam." The HandyCam system allowed a live action camera to be used to "shoot" an animated scene. The camera operator would operate the physical camera and the capture system recorded those movements and used them to drive the virtual camera for the shot. Since the hardware and software used in this system was real-time, the operator would get instant feedback of the performance in the form of a 3D smooth shaded version of the virtual world fed back into the eyepiece of the camera.

The very first shot that used the HandyCam system proved to be very educational. It seemed like a simple shot which had already been blocked out by key-framing a camera in Maya. It consisted of a camera watching some pieces of a surfboard wash up on the beach and then tilt up to find a surfer on a wave near the horizon. Interestingly, after nearly 20 takes and several different operators, the camera move felt completely awkward and un-natural. After studying the original key-framed camera move, it became obvious to the Head of Layout that the angles called for in the move were almost impossible for a real operator to achieve. A simple adjustment was made where the camera operator stood in a slightly different location on the beach to get the shot from a more natural angle and a couple of takes later, the camera was finished and ready to go into the film. The final camera work looked as though it could have been caught "in the moment" by a cameraman on the scene and felt correct for the movie—a significant improvement over the original key-framed version of the shoot.

From that point on, the general rule for the HandyCam system was that if you couldn't get the shot in a few takes, something was probably wrong with the concept or the basic idea of the shot and it should be simplified. Since the movie was a documentary, the filmmakers wanted it to feel as though it had been captured in real-time as the events took place and there was no chance to go back and get a second take.

Since the system operated in real-time, it allowed the production the flexibility to use the HandyCam system on as many shots in the film as was appropriate. In all, the crew used this tool on over half the shots in the film including helicopter shots and shots "filmed" underwater.

In addition to providing a natural hand-held feel to the camera work in the movie, the HandyCam helped to give an authenticity to the camera work throughout the film by keeping the camera moves simple. It lent more credibility to the documentary feel, even if the audience only perceives it subliminally.



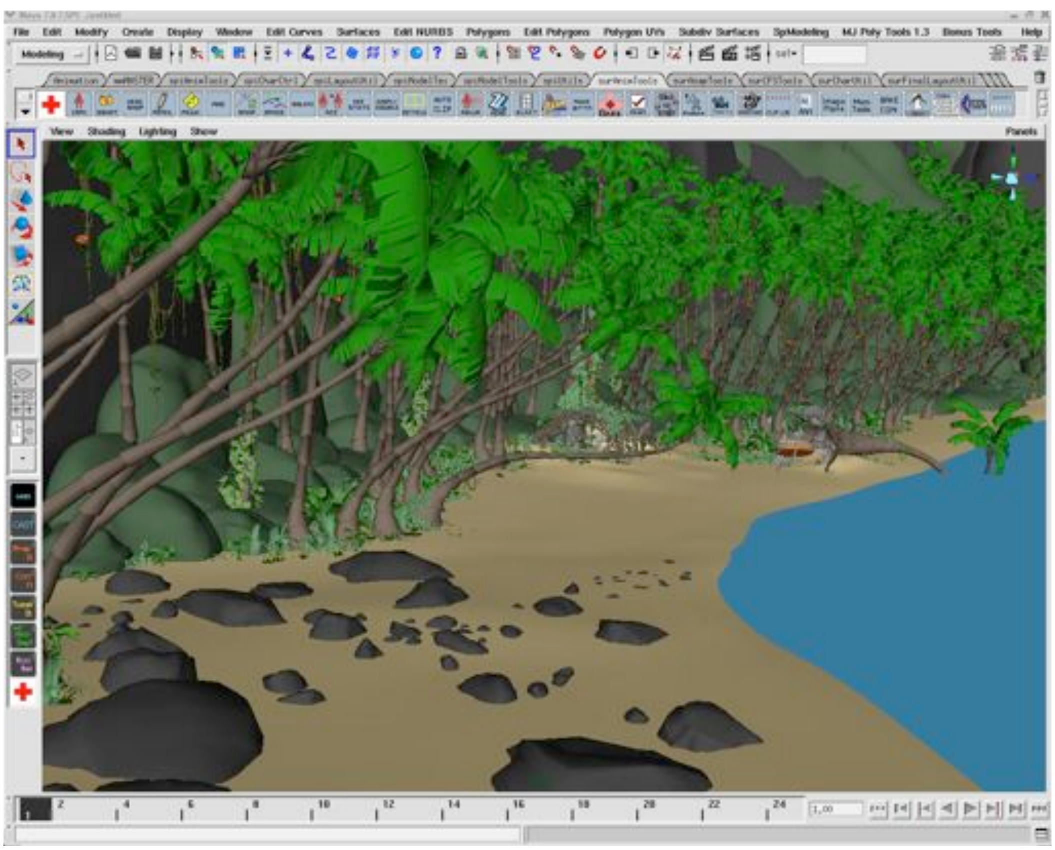
*HandyCam in action*

# 2.3.2 The Hardware and Software

Imageworks is highly experienced with motion capture and integrating live-action cameras into a CG world as evidenced by the work on *The Polar Express* (2004), *Monster House* (2006), and others currently in production. On *Surf's Up*, the Visual Effects Supervisor wanted to build on those existing camera capture systems (previously referred to as "Wheels"; see *From Mocap to Movie: The Making of The Polar Express* course notes (2005)) and improve them in two specific areas.

First, the 3D playback portion of the system needed to work in real-time in Maya. Since the front-end of the Imageworks pipeline is built around Maya, the overhead involved in moving complicated scenes into another 3D package, which is better optimized for 3D viewing, can be taxing to the efficiency of a production. In particular, since the *Surf's Up* scenes required complex interaction between the wave and the characters, an automated translation step would have been very complicated or impossible.

As a result, the CG Supervisors on *Surf's Up* worked closely with the Imageworks Software department to develop an optimized playback system inside Maya. This custom plug-in enabled real-time caching and playback of hundreds of thousands of polygons per second while keeping all the data in the format native to the facility pipeline. The tool was so convenient and provided such quick feedback for complicated scenes that it was quickly adopted throughout the Imageworks pipeline as the primary display engine for complicated 3D geometry.



*Example of complex scene data from Big Z's beach*

The second major improvement was the hardware and software used to capture the position and location of the camera. In previous movies that Imageworks had completed, the hand-held camera moves were designed to simulate a high-end steady-cam as used in most big budget features. This had the technical advantage that the capture system could have small errors and noise in the data which could later be filtered out by smoothing and would result in a camera move which felt correct for a high end film.

For *Surf's Up*, the team desired the little bumps and imperfections that are characteristic, and part of the charm and authenticity of a low-budget documentary. These are a result of not having access to a steady-cam and other high-end equipment when shooting in remote locations. Technically, this required a highly accurate capture device that would need very little post-filtering to achieve the final look.

For these requirements, the Visual Effects Supervisor turned to 3rdTech to leverage their HiBall-3100 Wide-Area tracker. This hardware was ideally suited to the requirements of the show as it easily covered a medium size capture area of 16'x16' and works indoors with a low ceiling. The system also yields highly accurate data since the beacon arrays are statically mounted to the ceiling and the high resolution sensor is attached to the camera. Using that type of design, a small change in the angle of the camera provides a large change to the position of the markers across the sensor; so the accuracy of the rotational data is extremely robust. The HiBall-3100 tracker has a very low latency as well.

The Imageworks team also installed encoders onto the zoom controls on the camera so the operator could manipulate the zoom as well as the position and orientation in a completely intuitive way. All of this data was fed into a cohesive library which managed and recorded the data and sent the appropriate streams to Maya for real-time display.

# 2.4 Slow Motion and Time Manipulation

Documentary filming relies extensively on manipulation of time for purposes of storytelling, setting the mood and generally making something out of a limited amount of footage. For *Surf's Up*, several different types of time manipulations were implemented to allow the filmmakers the same amount of flexibility that you would expect if you were shooting a live-action surfing documentary.

## 2.4.1 High Speed Photography

The first type of time manipulation was to allow any shot to be photographed using **High Speed Photography** (aka: slow motion, or shooting over-cranked). This is to mimic the live action camera that is configured to shoot more than 24 frames in one second. The slow motion effect is smooth and graceful. **High Speed Photography** gives a high-end and almost hyper-realistic look when shooting water. Very high speed cameras shoot 200 or 300 frames per second and are great for shooting effects work or close up crashing waves. *Step Into Liquid* provided great reference of breaking waves shot at 200 fps.

For **High Speed Photography**, the Layout department determined a *camera speed*. This speed was recorded in a database and displayed on the slate for everyone to reference. The *camera speed* indicated the frame rate of the camera if it were shot on a live action set. A value of 48 fps slowed everything in the shot down by a factor of 2. The Layout, Animation and Effects departments all used this frame rate and adjusted their workflow using various tools so that the final output of the department looked as though it was shot at the desired frame rate.

For example, Layout had a shot that they wanted to shoot at 240 fps (extreme slow motion). Layout recorded a real-time handheld camera move and then slowed it down by a factor of 10 to get the appropriate speed for the shot. Animation would take that shot when it arrived in their department and squeeze it down by a factor of 10 to first animate the character in real-time. Once the real-time blocking is approved, the animation file would then be stretched back out to 10 times the length so that it appears to be moving in slow-motion and any last touch-ups would be completed. The Effects department also built all of their tools to respect the *camera speed* variable so that particle effects and other simulations would slow down correctly for the shots.

The procedural implementation of the high speed camera allowed the filmmakers to use slow motion at will during the production of the film without significant budgetary impact.

## 2.4.2 Step Printing

Another tool often used in documentaries is to **Step Print** a shot (aka: double-printing or triple-printing). Often, footage is acquired at a given frame rate during the shoot but later it is decided in the editorial post-production that the shot needs to be slowed down. At that point, the only option is to print each frame in the shot twice, three or four times to make the action fill more time. This results in a stepped slow motion effect that looks more low-end and gritty. It's ideal for cutting action into faster paced sequences or anywhere that a smooth slow-motion effect wouldn't feel right.

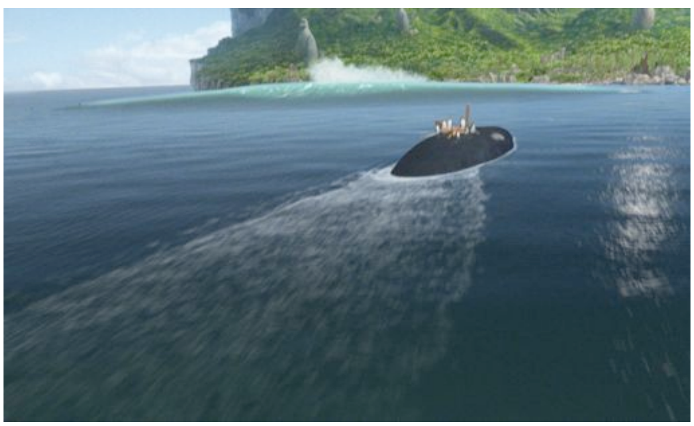
For **Step Printing** on *Surf's Up*, the production again mimicked the live-action paradigm and the editorial department could step print and adjust the timing of shots in "post" at anytime up until the edit was complete. This allowed the filmmakers the flexibility to make timing changes to shots after they were animated and lit and certain frames were double or triple-printed to achieve the necessary slow down. The result was that the final shot appeared to have been shot in real time (or at the *camera speed*) and then the *playback speed* was adjusted in "post".

The difference between the two types of slow motion is somewhat subtle but the audience picks up on the variety of different techniques used in the film. The conventions adopted by *Surf's Up* mimic the conventions an audience is used to seeing in any surfing documentary and helps to aid the authenticity of the movie.

## 2.4.3 RAMPED CAMERA SPEEDS

The final type of time manipulation used during the production of *Surf's Up* was the **Ramped Speed** shot. For these shots, the *camera speed* or the *playback speed* was adjusted during the shot for maximum impact. On a live action set, a ramped camera actually allows the camera to change frame rates during a shot. It's a complicated technique that requires the coordination of several things to actually adjust a "frame rate" dial on the camera while simultaneously adjusting the aperture of the lens to maintain proper exposure. Ramping the camera speed allows for a shot to go from real-time to slow motion during the course of a shot and has been used recently in high-end feature films.

1



2

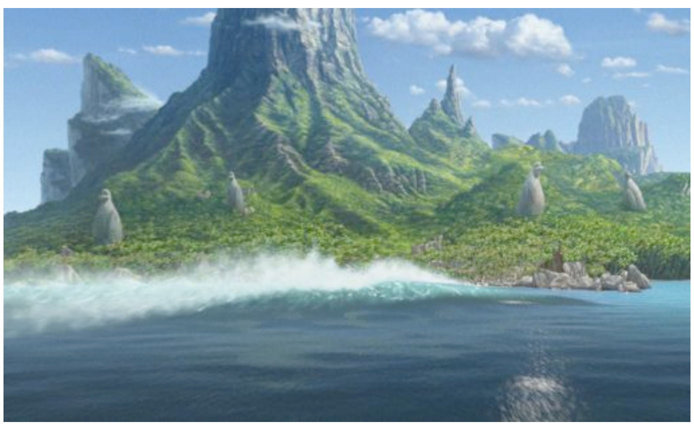
5

6

The **Ramped Speed** shots in *Surf's Up* were managed as "one-offs" since the automated tools did not have the ability to handle the various speeds on behalf of the user. In each of these shots, the animation, FX and camera were massaged to have a ramped speed feel to exaggerate the dramatic timing of the shot and draw the audience's attention to the part of the shot that the filmmakers were interested in focusing on. The **Ramped Speed** shots always required some combination of the above techniques to create the illusion of smooth changes between camera speeds.

The combination of all three time manipulation techniques was designed to continually remind the audience that the movie was shot with a camera and edited like a traditional surfing documentary—the surfers just happen to be penguins.

**3**



**4**

**7**

**8**

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

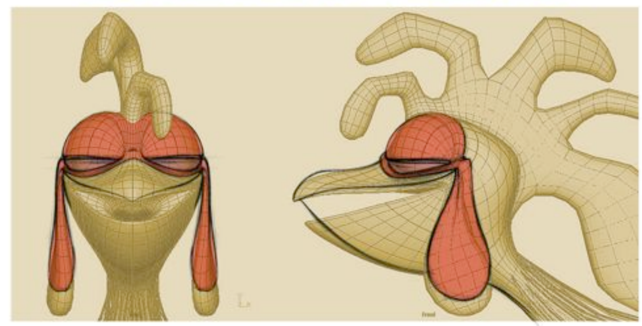
# 3.1 Animation Team

The *Surf's Up* animation crew consisted of the Animation Director, four Lead Animators and approximately 60 Animators at full capacity. The Lead Animators were sequence-based (rather than character-based), and would be cast sequences based on their strengths and specialties. For example, the surfing sequences were managed almost exclusively by the surfing team because of the level of specialty that surfing required. The same applied to sequences that predominantly featured a certain character that a given crew may have shown a knack for.

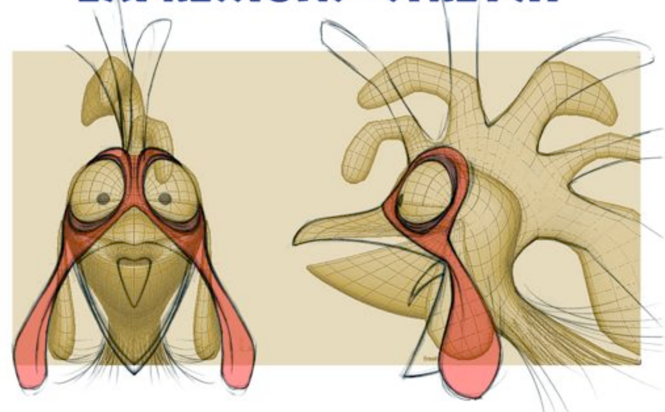
It was the goal of the Directors to create a film (and environment) where the animators could truly have fun with the animation. Many shots were staged as an open palette where animators could cut loose and allow the animated performances to drive the sequence. The crew took to the idea that this was an "animator's movie" with incredible performance opportunities that don't come along very often.

The Layout, Animation and Effects crews were joined at the hip, and got even tighter as time went on. It was extremely important to maintain close communication between departments for every shot that traveled through the pipe. At no time did one department "throw-it-over-the-fence" to the next department and move on. The smallest change (or detail overlooked) would always have a way of affecting other departments downstream. The goal was to always keep shots moving forward through the pipe; never backward. Ongoing communication between departments was the key to making that happen.

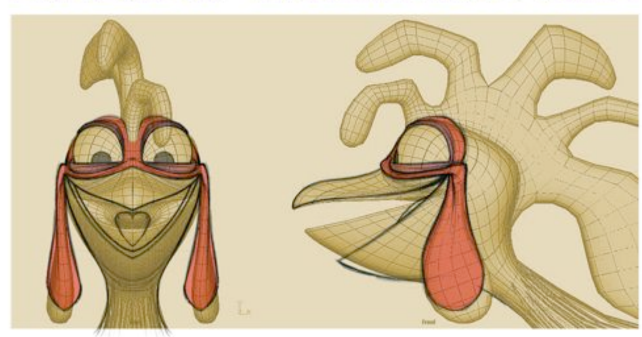
EXPRESSIONS - SQUASH



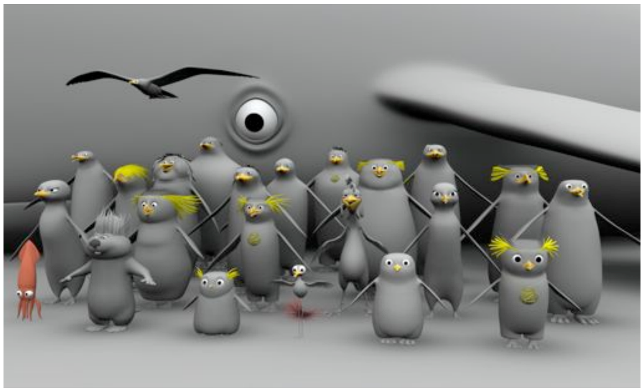
EXPRESSIONS - STRETCH



EXPRESSIONS - EXTREME HAPPY SMILE

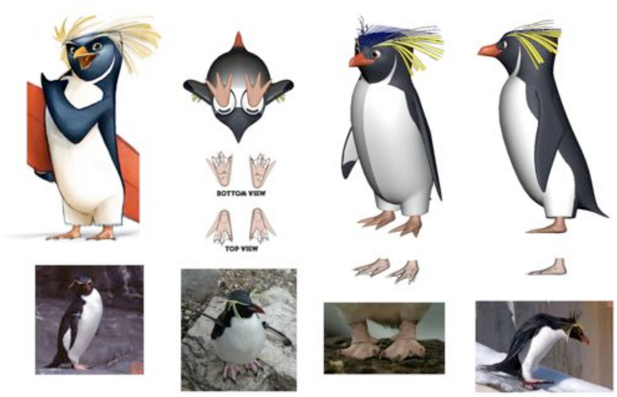


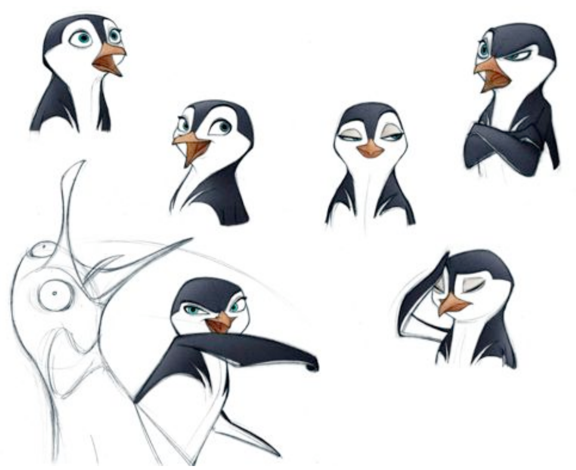
# 3.2 CHARACTER DESIGNS



Character designs for *Surf's Up* began in January 2005. While the characters are loosely based on reference from specific penguin types (rockhoppers, emperor penguins, etc.), the adaptation of these references are steeped in a rich heritage of classic 2D design. A very distinctive feature of the penguin designs is the painted patterns on their feathers. The Geek has the subtle hibiscus pattern, whereas Tank Evans has prominent tattoos. These distinct lines and patterns are unique for each of the lead characters. The directors knew that they wanted to cast Jeff Bridges for the role of Geek, so it was important to move quickly with that design first so that Jeff Bridges could be sold on the idea as well. The concept and design won over the actor, and the Geek design ultimately drove the look of the other characters, all of which underwent much iteration over the year leading up to production.

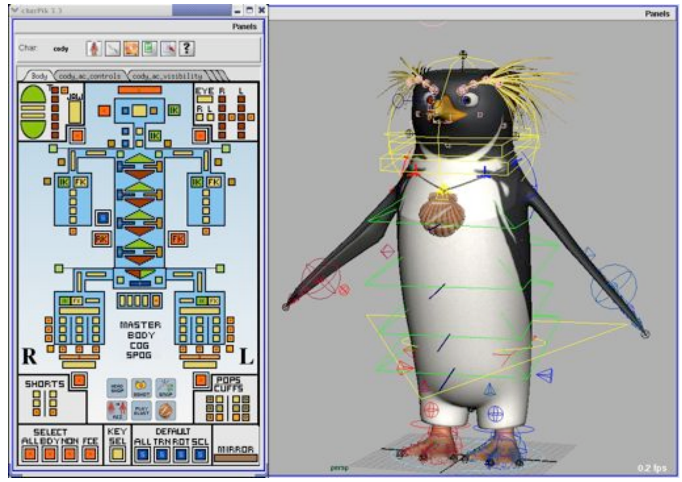
By January 2006, the designs for all of the characters were approved by the Directors and were able to move forward into modeling. Each design was clearly defined from all angles with the traditional array of model and pose sheets, as well as artwork for the complete set of phonemes and facial expressions. The goal was to replicate these designs in 3D with a high degree of accuracy. Even after modeling and rigging began, the Character Designer remained on hand to consult with the Animation Director and Modeling Supervisor so that the intended designs translated seamlessly to a 3D world.



# 3.3 Cody's Animation Rig and Controls

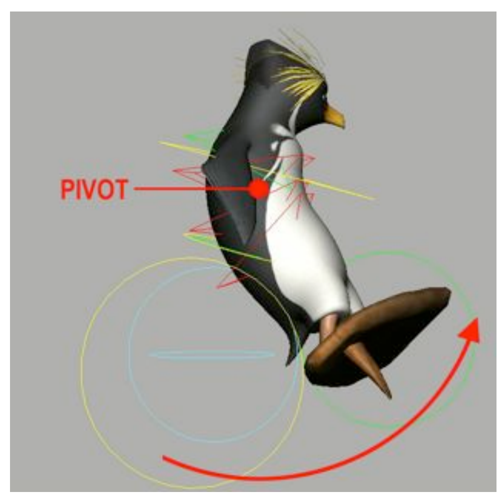
The most important goal for the animation rigs on *Surf's Up* was that they be fast and responsive. From the perspective of the Animation Director it was better to animate with a basic, stripped-down puppet that was *fast*, rather than use a slower rig with a lot of cool features. This way the animators could do many more iterations of a shot, and experiment with new acting ideas that may never have been attempted with a slower rig. When it came to the characters in this film, it was less about the technology and more about the heart behind the performance. The technology could not stand in the way of that creative process. There were a lot of "cool" features built into the rigs up front that were ultimately stripped out because of the performance hit. What was left in the end was a very efficient rig with only a few unique features that are summarized in this Section.




## 3.3.1 Body Rig

### 3.3.1.1 Reverse Spine

*Surf's Up* was not an extreme squash-and-stretch style of film but it did have a fair amount of poses that couldn't be achieved with a standard FK spine. With that in mind the FK spine was equipped with a special feature that allowed the animator to pick a point along the spine at which to "reverse" it, essentially folding the spine in the opposite direction of the normal spine. The reverse spine was particularly useful when the animators were posing a surfing character. Surfers are often getting their feet above their heads and using the reverse spine allowed animators to keep the top part of the body in a fixed pose while rotating the lower half of the body from a pivot point in the chest area.

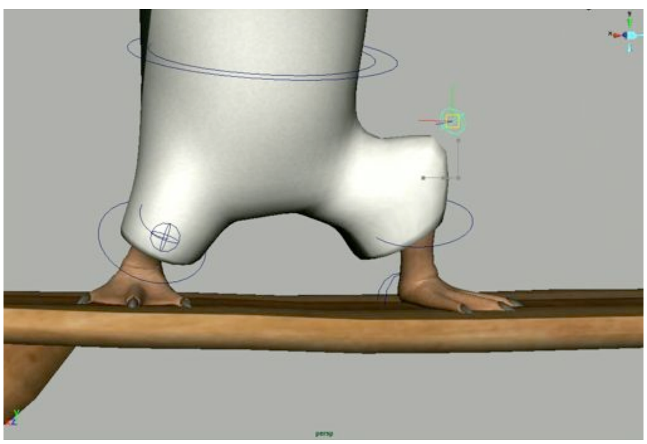



*Reverse spine pivot*

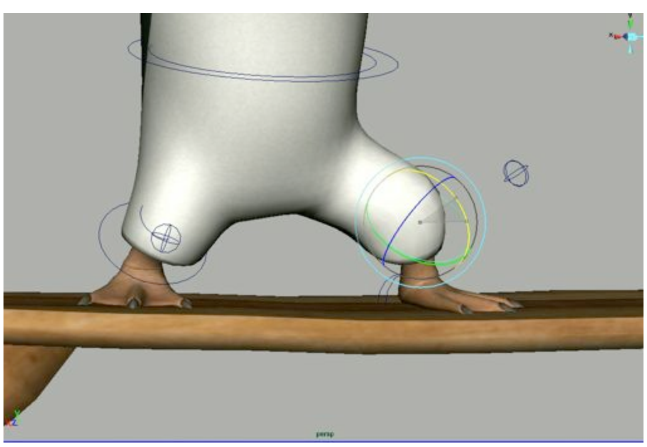
# 3.3.1.2 KNEE POPPERS AND CUFF CONTROLS

One of the most interesting aspects of the characters was the setup and animation of penguin's legs. Early in the design phase of the characters, a "board short" look was created for the legs of the penguins. This was not an anatomically correct penguin leg, but for this movie the penguins needed to surf. The knees of an anatomically correct penguin are actually enclosed inside their bodies (as is most of the leg) making them look as if they were walking inside a sack of potatoes. In contrast, Cody and most of the other lead penguins made a complete departure from reality and had short visible legs that needed controls to achieve a wide variety of poses. The setup team created controls that allowed animators to move the shorts independently of the inner leg. The FK short joints coupled with the cuff controls allowed for the animators to pose the shorts into the exact poses they needed. The inner legs also had controls that could "pop" the knees in the situations where the animators needed a more visible and defined knee shape. This was especially helpful when the characters were surfing as the legs were often scaled beyond their default length to achieve the desired poses.

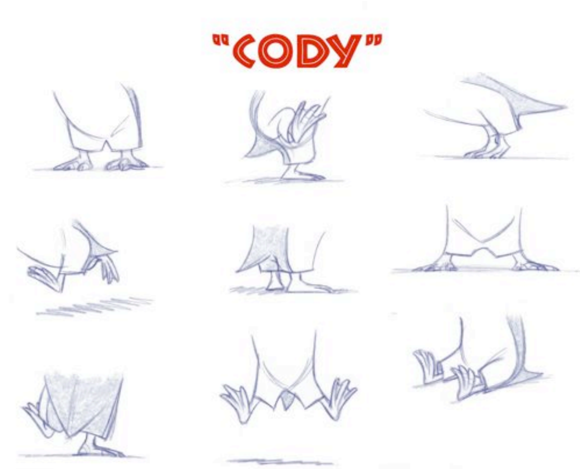
*Anatomically correct skeleton vs. Cody's modified skeleton*



*Knee poppers*



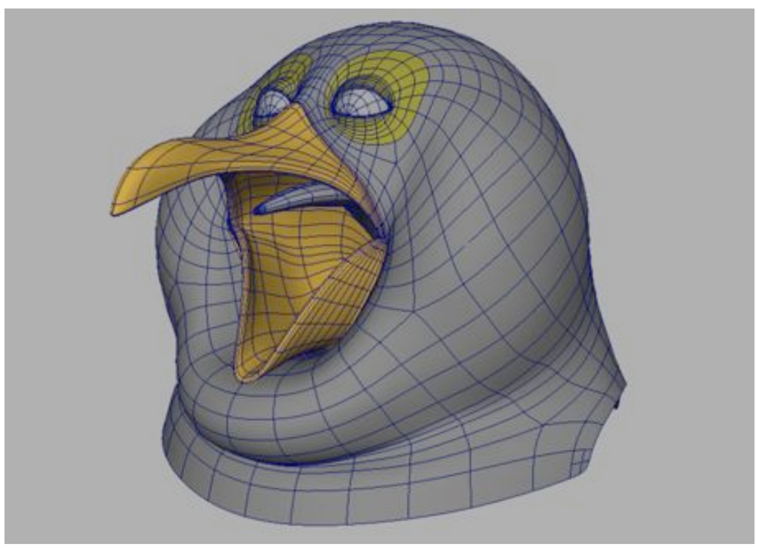
*Cuff controls*

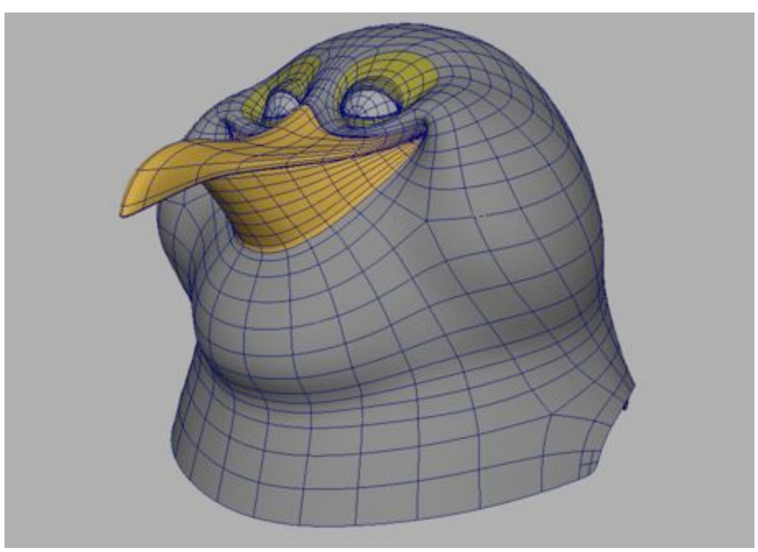


# 3.3.2 Face Rig

## 3.3.2.1 Muscles vs. Blend Shapes

*Surf's Up* face rigs are a hybrid between a blend-shape system and a muscle-based system. The muscle system was used solely for the brows and eyes of the characters. The blend-shape system was used for the beaks and cheek area. Due to the "relatively" rigid beaks of these characters, the non-linear-in-between nature of a simple blend-shape system sufficed. The muscle system used on the brows provided a fleshy feel to the character's skin that came across nicely in the final result. By splitting the face into two system regions, the workload was more efficiently distributed between the Rigging and Modeling departments.





# 3.3.2.2 Beak Rig

The initial face rig was based on penguin anatomical studies and reference. The beaks of real penguins are rigid, but the emulation of a real penguin's beak was quickly deemed the "Muppet Mouth" due to its puppet-like performance limitations. Coupled with the natural, off-beat voice work by the actors, the rigid beaks could not achieve a believable level of expression. Even so, the anatomical approach to the rig was the basis for these characters, and the first step toward converting the essence of a real penguin into a more artistic/stylized interpretation. The goal was to infuse more deformation controls into the beak rig that would allow for more "humanistic" behaviors.
This implementation introduced such features as:

- Controls along the length of the beak to sculpt the curvature of the lip-line.
- Controls to position the mouth corners in multiple directions.
- Controls to deform and roll the end of the beak.
- Controls to deform and shape the length and curvature of the mandible.
- Controls to deform the entire beak volume up, down and side to side.
- Controls to puff up the beak to simulate air pressure inside (to accent hard consonants in dialog).

Although the beaks didn't deform in a truly anatomical nature, the acting performance was much more successful with the added level of articulation.





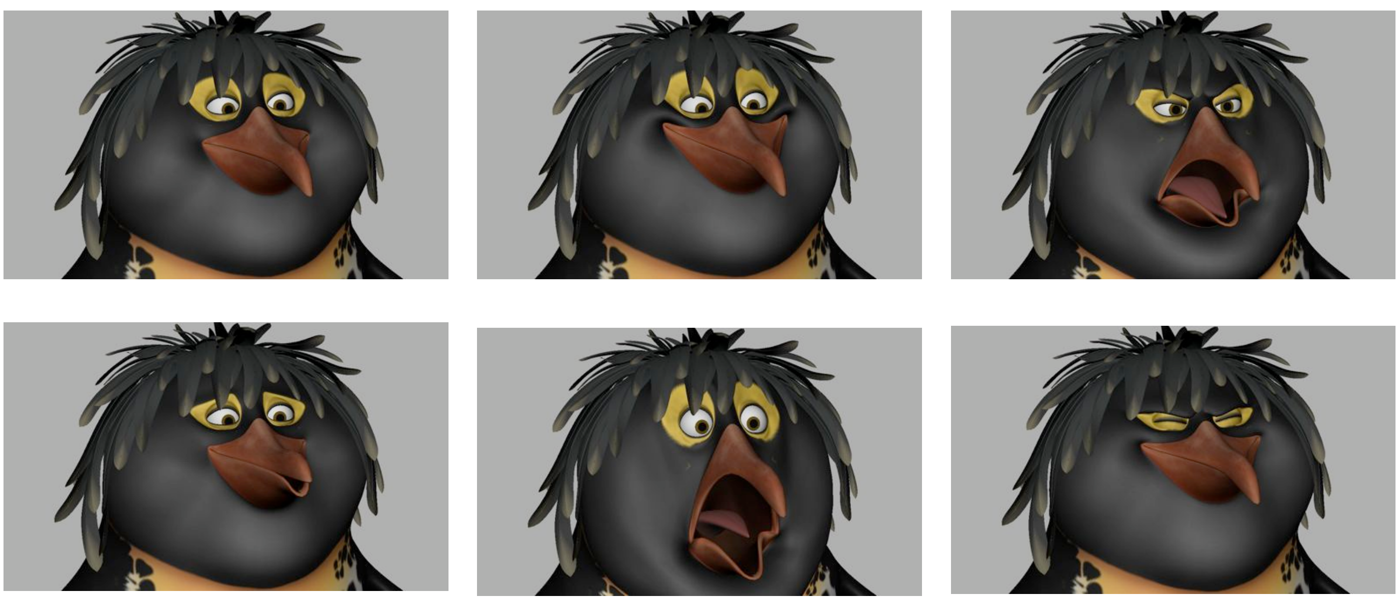

# 3.3.2.3 Eyelid Rig

The foundation of the eyelid rig is a series of scaling joints located on the edge of the eyelid. There is one joint for every poly edge that flows into the eye. This seemingly overabundant level of control provided animators the ability to adjust every span of the eyelid to achieve any desired eye pose.

The eyelid rigs also had two distinct features that were designed to both enhance and simplify the process of animation. They are the "Sum-Of-Ten Blink System" and the "Eyeball Follow Feature."

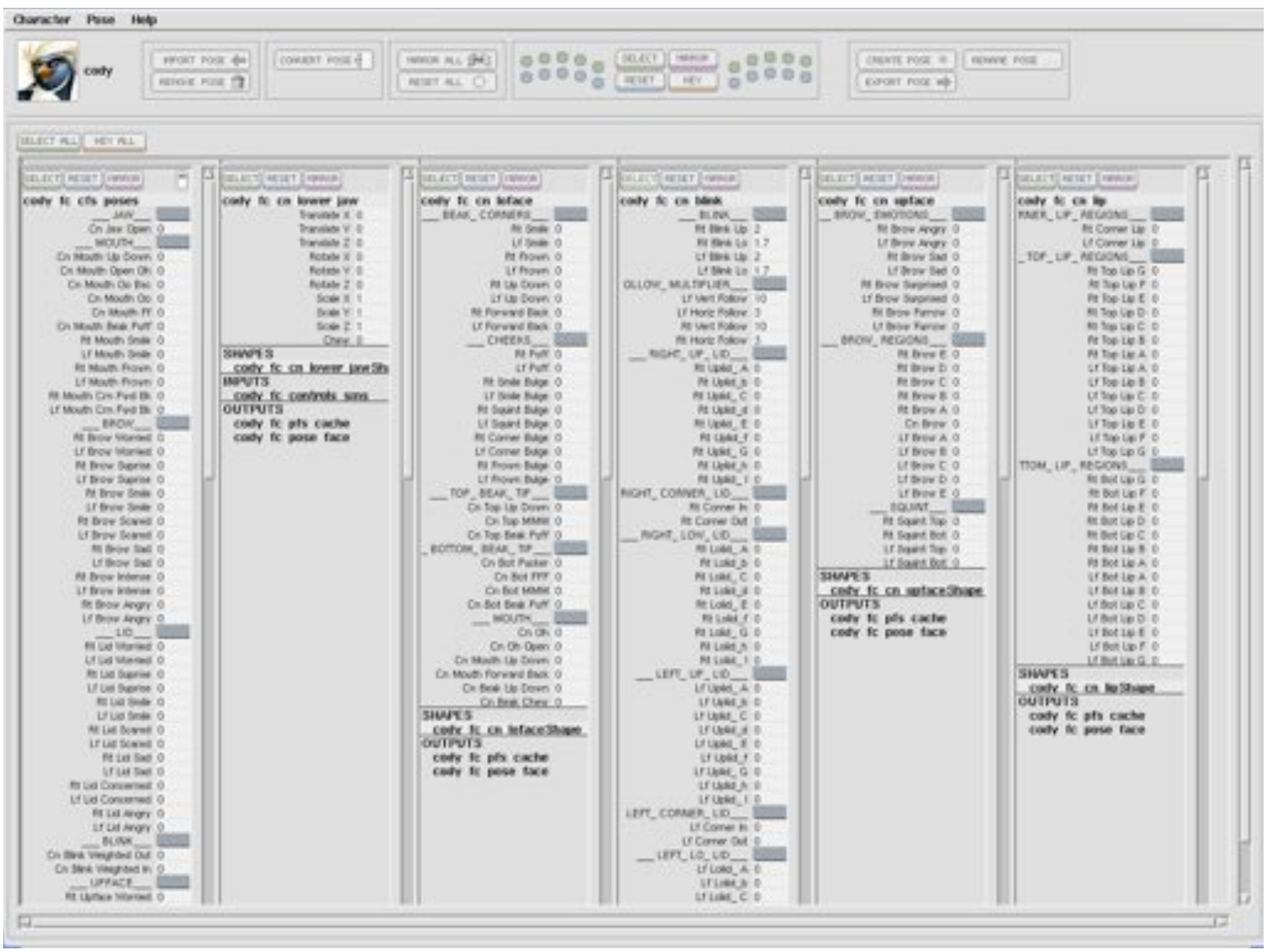
The "Sum-Of-Ten Blink System" helped animators ensure that the character's eyelids are completely blinked regardless of the existing eyelid pose. Each eyelid control contains an upper eyelid and lower eyelid blink attribute. The logic of this system is as follows: As long as the sum of the upper and lower eyelid attribute value is 10, then the mating edge of the lids will meet precisely. Animators can control the location of the lid convergence by altering the bias of the two numbers. The default value of each attribute is 0. For example, a value of 9 on the upper eyelid attribute and a value of 1 on the lower eyelid attribute would result in a low blink line. The value of 9 means that the upper eyelid travels 90% of the distance and the value of 1 drives the lower lid 10% of the distance.

The "Eyeball Follow Feature" was created to infuse more human-like qualities into the penguin characters. When a human eye looks up, down, left or right, the eyelid stretches and contracts, folds and unfolds to track the motion of the eyeball. Without this deformation feature, the eyeball looks like a marble rotating inside the head of a rigid mannequin. This feature has become a standard feature in the Imageworks character rigs.



# 3.3.2.4 Facial Animation UI (Surfshop)

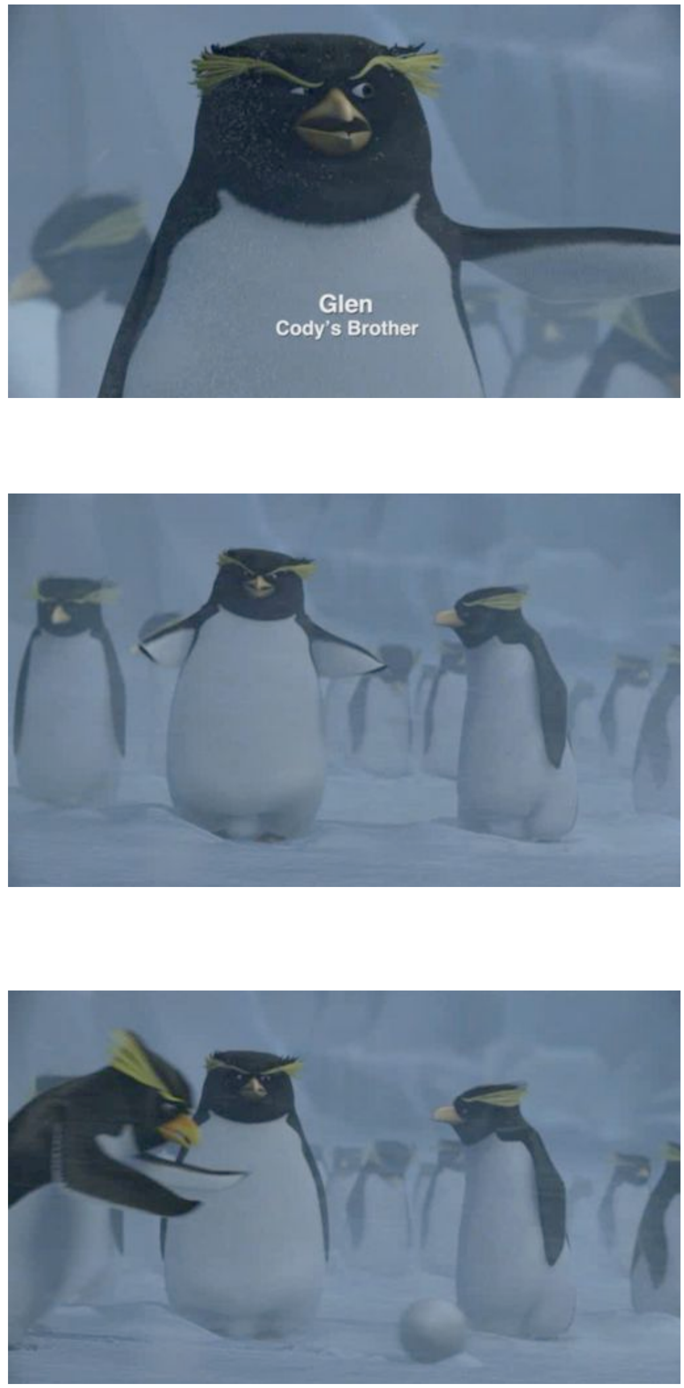
"Surfshop" is a User Interface for the facial animation system. It organizes all of the facial controls in a logical, accessible interface for the animators. Leveraging off of Maya's channel box widget, Surfshop displays multiple categorized columns of attributes. Only the chosen facial control nodes are displayed in the UI. With this interface, animators can easily view every keyable attribute in the facial rig at the same time. This interface eliminates the need to traverse through different facial controls and node hierarchies in search of desired rig functions. In Surfshop's first channel box column is the facial rig's pose node. The pose node houses a dynamic list of animator-created-and-named pose attributes. The facial poses driven by these pose attributes differ from the rig's other attributes in that they are constructed solely from the rig's other facial controls represented in the remaining columns.



# 3.4 Character Development

## 3.4.1 Development Tests

The first animation tests got underway in January 2006. Tests were done with scratch tracks from various recording sessions by Jeff Bridges so that the animation style for the film could be explored. A marketing shot was also produced for ComicCon in March 2006 that featured the Glen character, voiced by Brian Posehn. Much was learned about what did and did not work through these tests, and the style of the animation began to take shape. The ComicCon shot was ultimately used in the film, although it had to be reworked to suit the animation style that had evolved (see Section 3.2). While the performance tests defined the acting style, broad movement of the characters had not yet been fully explored. It was then time to see how the characters walked, moved and got around.




*The first character test shot*

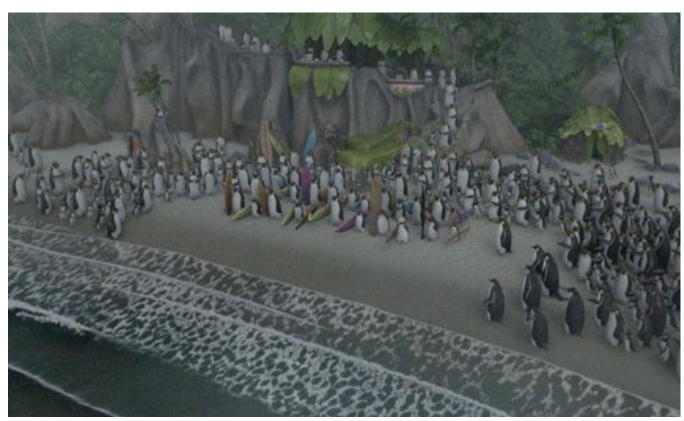
# 3.4.2 Cycles

As animators began rolling on to the show, each one was assigned the task of animating cycles of various types. In addition to the usual walks and runs (and attitude variations for each), paddle cycles on surfboards were done as well.

Much can be learned about a character through the way it moves, and walk cycles are notorious for revealing who a character is, and how it sees itself in its animated world. Surprisingly, the way that a character paddled through the water (and popped up onto his board) revealed just as much about that character as any other performance test that was done. The animators brought great life and ideas to the table that ultimately defined who these characters were.

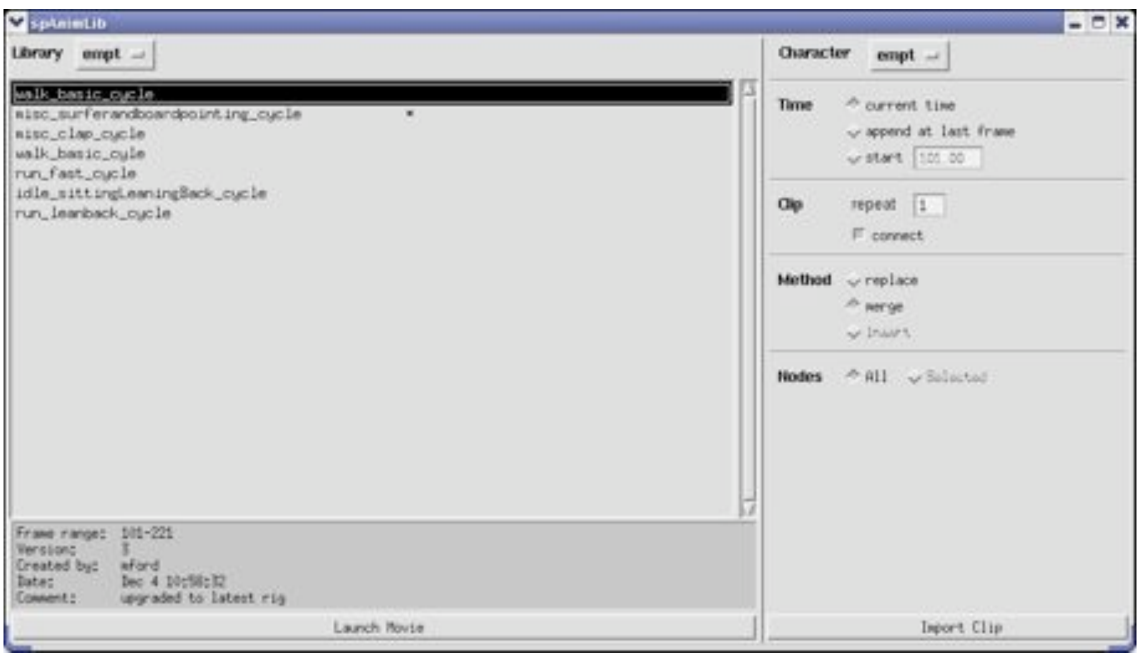
In addition, a whole series of “sitting, talking, cheering, clapping, pointing” cycles were animated as well. There was even a collection of “stand-around-and-do-nothing” cycles which consisted of little more than breathing and looking around. Each of these cycles was between 300 and 1000 frames, and could be used anywhere within the context of a crowd. A single cycle would look great in a crowd if all you did was offset the timing for each character. The result was a very natural looking mix of behaviors.

On the flip side, much was learned about the animators, and what their strengths and capabilities were. The cycle development phase of the project ultimately became a great casting tool for the shots that followed.



# 3.4.3 Cycle Library

The cycles were initially compiled into a web-based archive with paths to the various Maya file curves for each cycle. By mid-show, a more elegant interface was developed in Maya. The interface allowed users to choose a cycle from a list of choices, and preview the rendered version of the cycle in Imageworks' animation media player. From there the user could import those animation curves onto the selected character (even if the character was not the same as what was used in the original animation), and specify offsets and insertion points for the animation cycle.



# 3.5 Performance Animation

## 3.5.1 Premise

*Surf's Up* was structured like "Survivor-style" reality TV with lots of "unrehearsed" interviews and impromptu performances along the way. The camera was often "hand-held" to produce very organic moves with the natural bumps and bobbles of a live camera operator. There was a magic in this imperfection, and that was what the *Surf's Up* team tried to capitalize on in the animation as well. The performances needed to be portrayed as off-the-cuff, spontaneous and real. The result was an honest, voyeuristic glimpse into the true nature of these characters as they played out their lives before (and off) the camera.

Dialog was recorded with multiple actors improvising and playing off of each other (and talking over the top of each other as well). This is very unusual for an animated film where the recordings are generally more pristine and directed. The voice tracks in this film included microphone bumps, feedback, verbal fumbling and dialog quirks that made the voice performances feel raw and spontaneous. An interesting discovery was that the words people use don't necessarily articulate what they truly mean. If you transcribed the dialog from some of these tracks you would end up with a lot of verbal clutter that really makes no sense. True communication is *not* in the words, but in the phrasing, delivery and most importantly, the *body language* that supports it.

The documentary format allows the characters to drive the scene. The illusion is that the camera just "happens" to be there to capture the moment. Animators rarely get the opportunity to play out such long, extended performances where the characters carry the shot completely. From an acting perspective, it was an animator's dream come true.

# 3.5.2 Body Language

Body language reveals the emotional state of the character behind its words. It is the posture, subtle eye shape changes, the eye darts, the direction of the gaze, a lift of the brow, a tilt of the head, and orchestrating the delicate timing between all of these elements. Animators manipulate the body language to reveal a character's emotional state and the direction of its thoughts. Body language can reveal things about the characters that they may not even be aware of themselves.

In general, the goal of animation is to embellish the dialog with body language that directly supports the delivery of that line. In *Surf's Up*, animators are going for subtext beyond the obvious. What the character is saying and what he is thinking (or what he really means) might be two entirely different things. There is another layer of complexity here because the documentary camera is running, and characters might have different agendas that they reveal (or hide) depending upon whether the camera is rolling. Are they self conscious in front of the camera? Are they using the camera to shamelessly promote themselves? How does that personality change when the cameras are off?

# 3.5.3 ACTING

The acting challenge was to deliver performances that felt spontaneous and unrehearsed. It could be categorized as *"behavior animation."* It is the business that actors might do *between* takes when they stop acting and become themselves. If the camera caught them off-guard, then the team would animate through those awkward moments and try to capture some magic in the process. The audience must believe that the character is telling the truth or at least being true to himself at that moment (even if he is being true to a lie). Again, this is where the subtext comes through.

What appears *"spontaneous"* in animation is a result of crafting the performance down to the last little eye-dart. The subtlety of execution is evident down to such things as the breath that they take (another "body language" fundamental that defines the character's emotional state). The goal is to create something fresh and unique with every shot without relying on standard acting formulas.

The difference between a believable performance and one that is "overplayed" is often incredibly subtle. It can be a bottom eyelid raised just a touch too high. Backing off that lower-lid by 10% can change the emotional state of the character entirely. There are very fine lines in the subtle acting style portrayed in this film. The goal is to make the acting clear, but not hit the audience in the face with overly dramatic character clichés.

# 3.5.4 Style

The animation style of this film could be called "*caricatured reality.*" It is real-world dynamics pushed to caricature without breaking the fundamental rules of physics and gravity. Although the style is generally not pose-to-pose with cartoon physics, that line is sometimes crossed to keep certain performances sharp and punchy for comic value (like the natives, for example). Even so, there is a respect for the physics of the world that the Directors established up front. Most importantly, once the rules were in place they had to remain consistent throughout the entire film. Physics could not change to suit a certain animator's style, regardless of how interesting that performance might be.

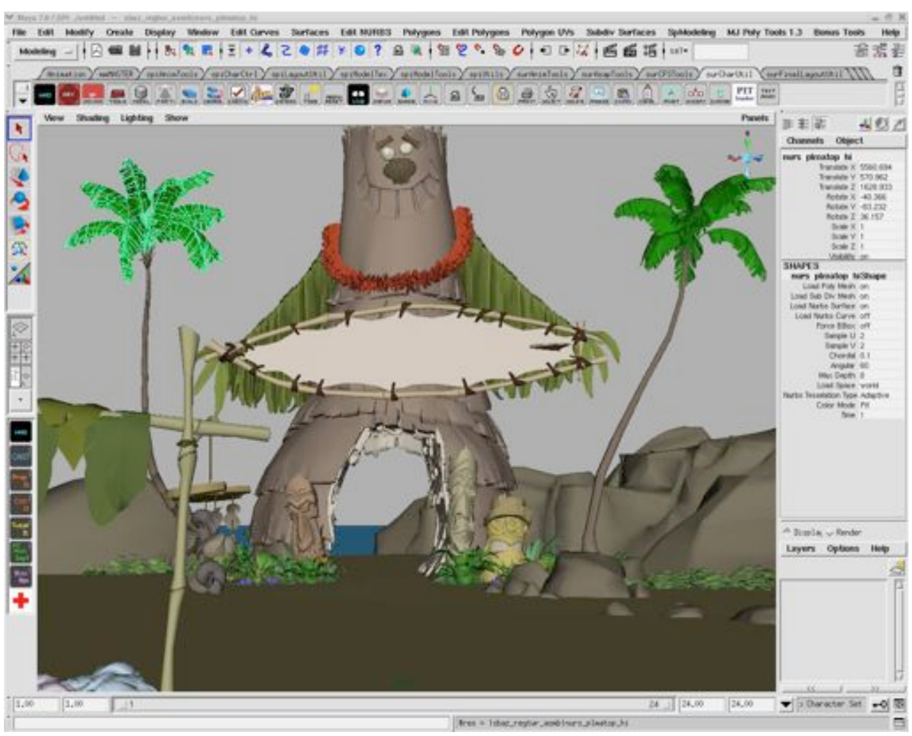
The naturalistic animation style is attributed in large part to the documentary format of this film. The natural feel of the "hand-held" camera was integral to that style. Snappy cartoon physics in the characters would certainly conflict with that reality. The human touch in the camera gives audiences a sense that a crew is present, who are just as much a part of the action as the primary characters (although behind the scenes). The characters often engage with the unseen crew, which reminds us that this film is being shot on location (rather than being created in the dark corridors of an animation facility). It was important to support that naturalistic style in the animation as well, or the two worlds would disconnect.

Cartoon physics, and the broad performances that go with it, tend to be more forgiving on the "believability scale" because it is a world where rules are made to be broken and everyone knows it. As we push closer to reality, the fine lines need to be considered. If a gesture or expression is pushed just a little too far, then the performance may not ring true, and it may feel a touch too dramatic. If you don't go far enough then you miss the mark on the other side. The goal is to hit the sweet spot where everything undeniably gels.

# 3.6 Layout / Animation Pipeline

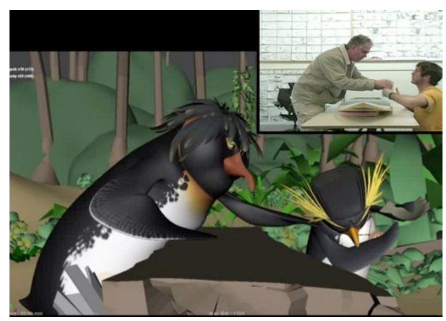
## 3.6.1 Rough Layout

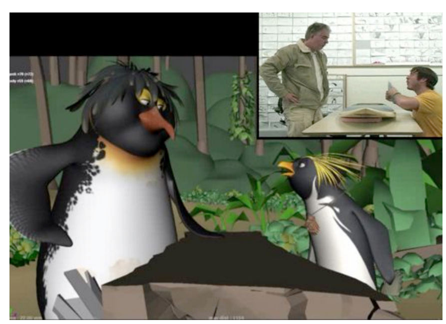
When a sequence was completed in storyboards and timed in Editorial it was launched into Layout. The Layout launch was a meeting led by the Directors and the Layout Supervisor and attended by the Production Designer, the Animation Director and the Visual Effects Supervisor. The object of the launch was to determine the cinematic style of the sequence and for any creative issues and concerns about the sequence to be voiced and addressed before moving the sequence into full production. The digital handoff was represented by an editorial cutlist which included timings for the storyboard panels and an audio track.  It was Layout's responsibility to arrange the panels into shot cuts. A tool was developed that allowed the panels to be grouped into individual shots but also allowed the shots themselves to be grouped into scenes. A scene represents a group of shots that share the same environments or the same characters. This data was exported into Maya which loaded the editorial information on a scene basis.  By grouping multiple shots that share similar elements into one Maya file, Layout could effectively choreograph the shots and make sure that modifications to the camera, character blocking or set dressing were automatically carried through into subsequent shots.  Rendering, camera animation and shot length changes were all taken care of through a single application launched from within the Maya session. When the sequence was approved and ready to move into animation, this same interface allowed Layout to launch a process which automatically broke the scenes into individual shots ready for animation.



# 3.6.2 Layout / Animation Interaction

Several sequences in this film consisted of little more than two or three characters interacting with dialog, and not much else in the way of specifically defined business. One example was "Making the Board," where Cody and Geek are standing in place and interacting without any obvious broad actions that could possibly allow Layout to define the best placement for the cameras up front. What would ultimately drive the camera placement (as well as the camera moves and cuts) would be the subtle looks and exchanges between the characters during their performance (which had not been defined yet). Again, the goal was to make this film look like it was shooting live footage and the camera operators needed to respond to the performance of the characters – not the other way around. To solve this problem, the Animation Director and the assigned animator worked out the acting for the entire sequence and then staged the performance so that Layout could shoot it on video. Four cameras were typically used for these sessions, with the sequence audio providing the audio cues for the acting. The reference video was edited by Layout as if it were a live-action shoot. Once the edited sequence was approved by the Directors, the camera angles were recreated in Maya, and the sequence was broken into shots as described in Section 3.6.1.





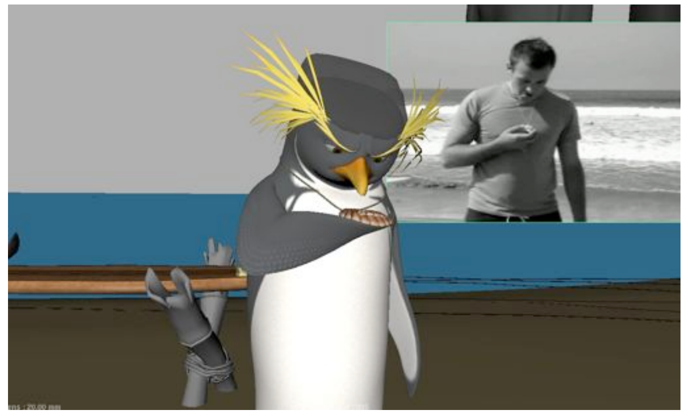
## 3.6.3 Animation Pipeline

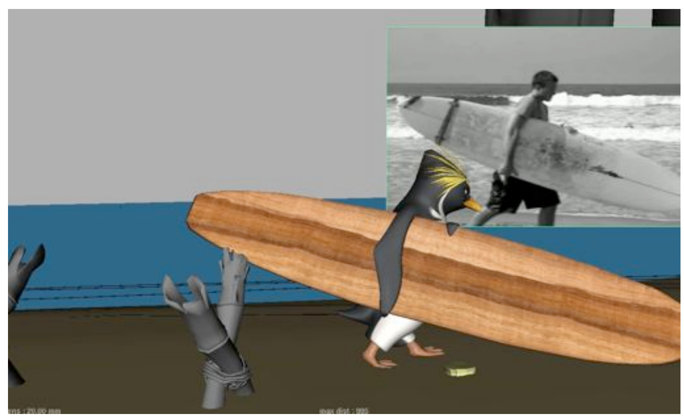
### 3.6.3.1 Shot/Sequence Launch

Once a sequence was completed in Rough Layout, it was launched into Animation. The animation launch meeting, led by the Directors and the Animation Director, was attended by the Lead Animator for the sequence and the animators that were cast to the sequence. The object of the launch was to get a complete download from the Directors about the goal and intent of each shot. It was during these sessions that animators would pitch performance and gag ideas back to the Directors as well. It was a very interactive and creative process, and the Directors were receptive and responsive to all of the creative input they received. What came out of the sequence launch was clear direction on what was important to the Directors in each shot, and in which areas the animators had flexibility to experiment and explore.

# 3.6.3.2 Blocking Techniques

The animators on *Surf's Up* came from different backgrounds, each with their own personal workflow and blocking methodologies. Many came from a 2D background, and they would pitch their ideas by drawing all the performance beats by hand before committing to them in Maya. Other animators would videotape themselves acting out their shot and use that footage as their performance reference. Blocking styles varied as well. Most typical was stepped curves, especially from those with a traditional background in animation. Other animators were very efficient with a layered-animation approach. There was no mandate on how the blocking was presented, provided that changes could be easily managed and turned around quickly. It was important that the animator did not go too far before showing a first-pass that clearly defined all of the acting beats in their simplest form.





# 3.6.3.3 APPROVAL CYCLE

Morning dailies and afternoon rounds were held with the Animation Director and the crew. Once the blocking for a shot met the performance goals established in the launch, it was put on the list for afternoon review by the Directors. Additional notes would be acquired at that time, and the animator also had the opportunity to pitch additional ideas that may have come to him/her in the process of blocking. Again, these ideas were highly encouraged and generally received with great enthusiasm by the Directors. Once a shot was approved in blocking, it was kicked forward to the Layout department so that a new camera could be set up to support the animation (see Section 3.6.4).

Blocking approval marked the beginning of the "Primary Animation" phase where the performance was fleshed out in greater detail so that the subtle acting nuances could come through. "Works-in-progress" would continue to be presented to the Leads and the Animation Director each day. Once approved, the shots would be presented to the Directors for "final" approval. After the shot was tagged as "final," by the Directors, the animator generally had a day (or two) to fix penetration problems, animate accessories such as Cody's medallion, and general cleanup – like fixing arcs, etc. Once the Animation Director approved the final cleaned-up version, the shot was published and moved forward to Effects and Lighting.

# 3.6.3.4 Annotation Tools

Imageworks' animation media player, itView, includes the ability to draw on top of rendered images (on a layer) with either a mouse or tablet. The tool incorporates all of the versatility that an artist would expect from a tablet-based drawing package including pressure sensitivity and an unlimited choice of colors. This way, the Animation Director and/or Leads could make annotations on top of any frame in the rendered sequence. The annotated layer could be saved as an annotation file and linked to that specific render. The animator could call up the layers for reference at their workstation, and also load the layers on top of new rendered versions to assure that all the notes were being addressed.


*Annotations in itView*

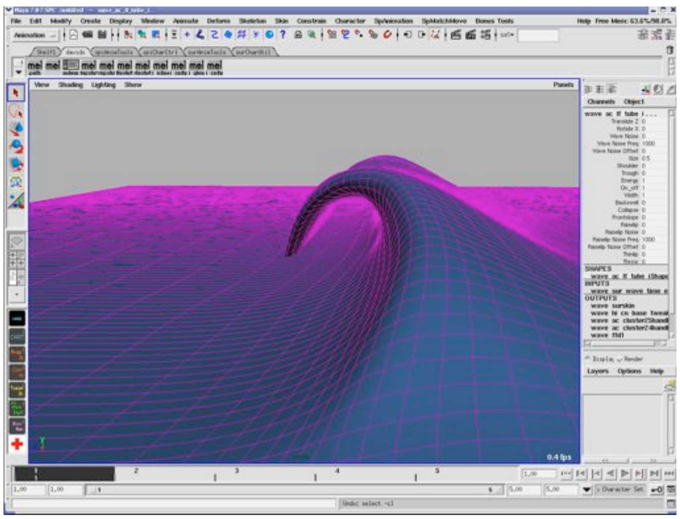
# 3.6.4 Final Layout

The hallmark of most reality film making is the apparent spontaneity of a hand-held camera that responds to the actor's movements. This made the placement and manipulation of the "rough layout" camera impossible before a performance was actually delivered. Until animation blocking was complete, the rough layout camera was regarded as exactly that – "rough." Once the blocking was approved, the camera was kicked forward to Layout so that a final camera could be produced that responded to the performance of the characters in the blocked animation. The goal was to produce a shot that felt as though it was filming (and reacting to) the action of the characters as the performances played out.

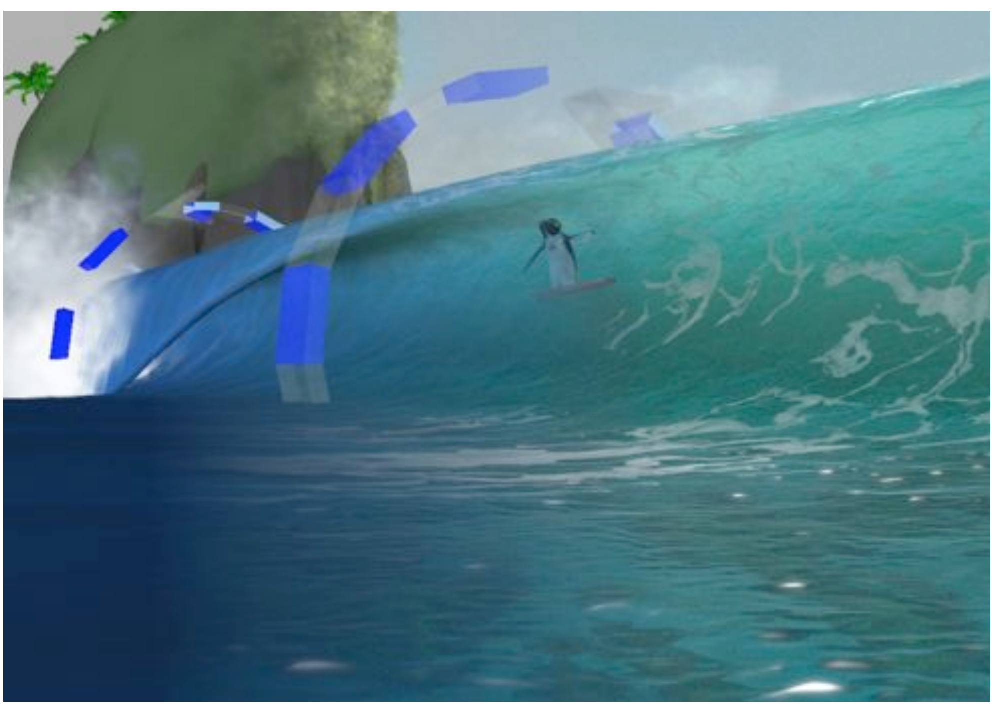
It was vitally important to lock down the camera before proceeding with Primary animation. Any time the camera changed (even in the slightest way), carefully designed silhouettes fell apart, undesirable tangents in the composition were revealed, and any cheats to that particular camera were exposed. By locking down the camera before entering into the Primary Animation phase, the animator was able to spend the rest of their animation time refining subtleties (including the cheats) to a camera that would not change.

# 3.7 Animating the Shape of the Wave

## 3.7.1 Goals



One of the most important elements in *Surf's Up* was the waves that are seen throughout the film. Without waves there wasn't a story to tell. In the early stages of production the team was confronted with the fact that the waves defined the sequences they were in. The waves were a moving environment created and animated in layout, similar to a standard set that would be built and dressed. The waves included hollow tubes for a sequence in which the main characters enjoy a perfect day in the surf, as well as powerful, monstrous 60 foot waves that are showcased in the finale. With that amount of diversity in mind it was important to be able to define every feature of the wave and create several different types and styles of waves. When coupled with the challenging process of animating a wave with the specific interaction of a surfer, it was ultimately decided that it was best to treat the wave as a character instead of a procedural effect.



# 3.7.2 Rig Builds and Wave Types

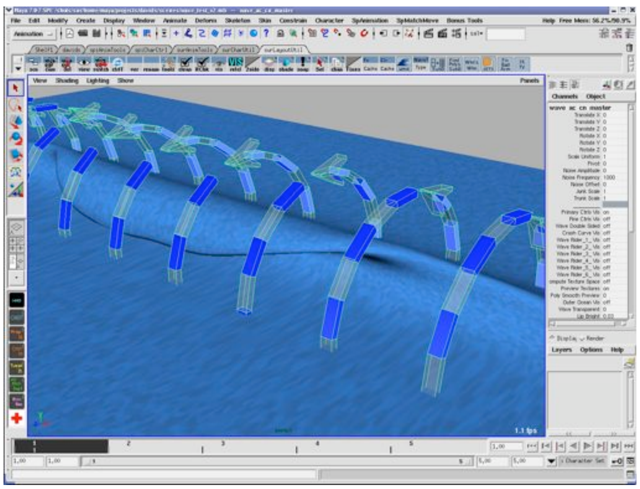
Characters that are built at Imageworks, like Spider-Man and Boog from *Open Season* are built using the same rig-build process. Surprisingly, fitting a wave into this system was actually one the smallest hurdles to get over in building the wave rig. By using the same techniques employed for other more standard characters, the rig would work within Imageworks' layout and animation pipeline and fulfill the creative and technical needs of the production. The basic components of the rigs are animation controls and geometry. The rig that drives the geometry and the wave was no different.

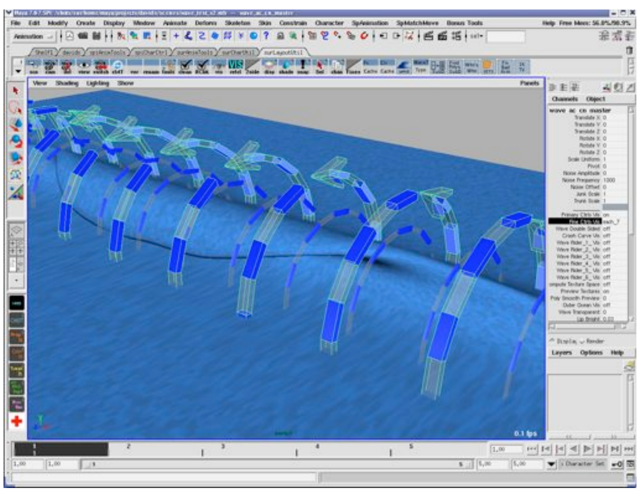
*Pipeline wave*

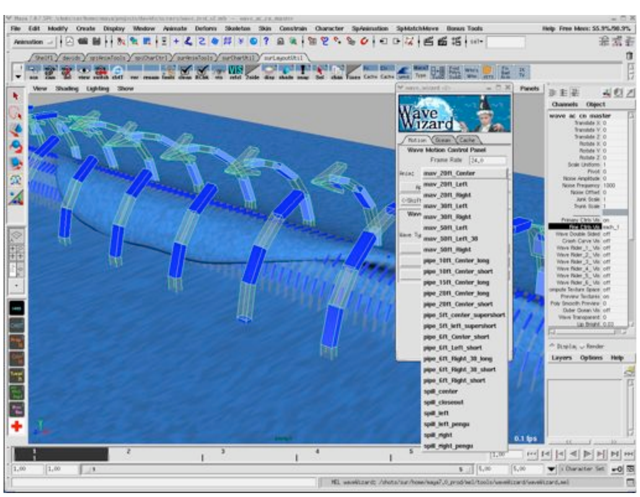
Knowing that an entire ocean surface with an arbitrary number of waves was an impossible workflow to manage, a wave surface created in the rig was built that would eventually be connected to the ocean through a surfacing operation in Houdini. In an effort to simplify the process even further a decision was made to build only two rigs, each very similar but offering different levels of control that would allow the animators to create multiple styles of wave types. These two rigs served as the base that defined a set of three distinct wave types that would populate the film. The "pipeline" wave was modeled after the Banzai Pipeline in Hawaii; a fast, hollow wave that breaks over a shallow reef. The massive waves seen in Big Z's archival footage and the movie's final contest were based on the infamous "Mavericks" in Half Moon Bay, California -- a giant, deep water wave that breaks over a rocky point. The "spilling breaker" was a beach break wave that does not tube.

*Mavericks wave*

## 3.7.3 Animation Rig








The waves breaking/cresting motion was animated using control rings whose rotational data drove a series of predefined target curves, which were blendshaped (morphed) together to "evolve" the breaking wave. Each blendshape target represented a period of time in the breaking motion of the wave. The blendshaped curves were lofted together to form the surface of the wave. Major control rings moved multiple curves or sections of the wave, and minor control rings could control a single curve. This level of flexibility allowed animators to have very fine control of the wave's contours. Other controls allowed for the wave's profiles to be manipulated into the desired type. By changing the size, shape and timing of the wave, the rig was flexible enough to evolve a wave from a 3 foot spilling breaker into a 60 foot behemoth.

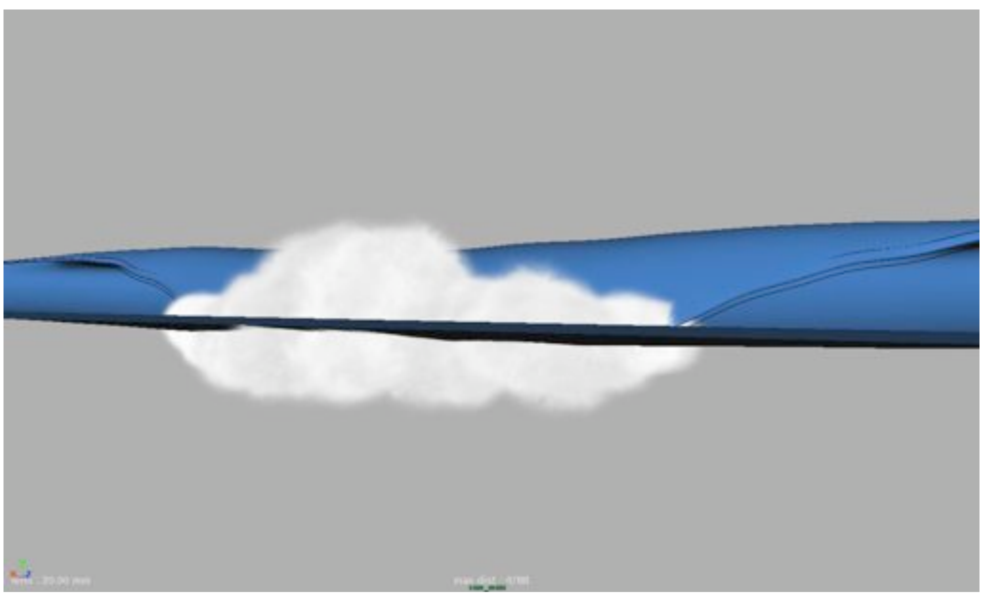
## 3.7.4 Animation Tools

During the layout process, the wave's type settings were defined using a tool dubbed the "Wave Wizard." This tool allowed for the layout artist to set one of the wave types, choose from a stored library of wave animations and turn on and off visual cues that accompanied the wave rig. The rig was updated based on the attributes that were set using this interface.

# 3.7.5 Visualization Aides

## 3.7.5.1 Whitewater

One of the most informative visualization tools created for the wave rig was the whitewater plug-in node. Running in real time, animators could see the amount of whitewater that the wave was generating as it was breaking. The white water node was especially helpful in determining when and how fast the wave was breaking and also aided the animators in knowing whether the character was interacting with the whitewater properly, or whether it was completely obscuring the character. (This element is discussed in further detail in Section 4.4.7.)



## 3.7.5.2 Wake Trail

When a surfer is riding the face of the wave the surfboard is displacing a volume of water as it glides along the surface. To aid the animators and layout artists in determining the surfer's connection to the wave, a wake trail node was created. This node created a lofted plane that was attached to the surface of the wave allowing the artist to visualize whether they were penetrating the surface of the water. The lofted plane looked like a ribbon on the surface of the water that allowed animators to clearly see the path of travel over the surface of the wave and orient the board correctly along the path defined by the wake trail. Before having the ability to visualize this path in animation, effects renders would often reveal that the board was sliding sideways along the surface of the wave (not tracking properly), and would be kicked back to animation to make the fix (a time-consuming and costly step backward). An interesting side effect of the wake trail was that it also allowed the animators to see the speed at which the wave was traveling. If the wake trail moved too quickly up the face of the wave, the speed of the wave would need to be adjusted. (This element is discussed in further detail in Section 4.4.6.)

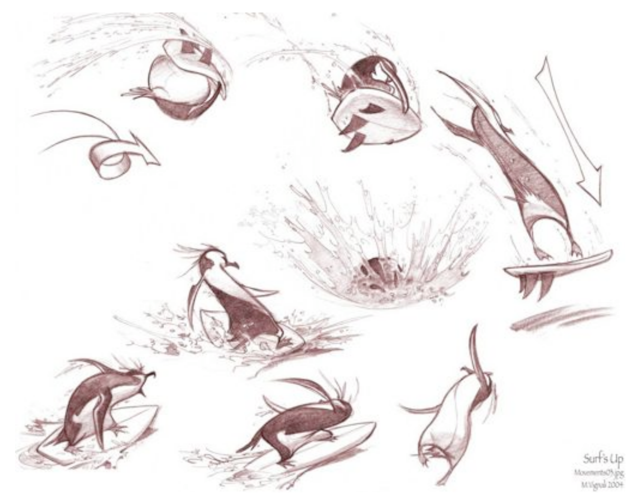
## 3.7.5.3 Z-Depth Water Render

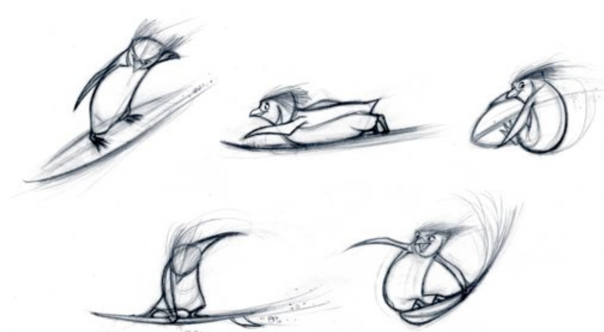
Because of the waves complexity it was virtually impossible to completely visualize the wave's look inside Maya with geometry alone. A z-depth image process was developed to allow the animators to interact with a rendered version of the wave from a specified camera. The render clearly showed the amount of displacement that was being generated from the shaders applied to the surface of the wave. The animators could move the characters in and out of the z-depth image on the camera's imageplane. This proved to be a very valuable and versatile tool that sped up the process of knowing whether the surfer was interacting with the wave properly.

# 3.8 SURFING

The physical dynamics involved in surfing can be very extreme and acrobatic. When surfers move they engage their entire bodies from the tip of their fingers to the tip of their toes in order to throw the board around with great finesse.

An archive of surfing reference was compiled and studied to get the feel for how a surfer manipulates the surfboard, and how the board responds to both the surfer and the water. The live-action reference became the pre-visualization tool. After receiving turnover notes on a surfing shot from the Directors, a collection of surfing clips were assembled that met their criteria. These options were pitched back to the Directors, and a clip would be chosen. Once the surf move was approved on video, the animator could proceed with a layered approach in animation (which seemed to work best for surfing), and the reference was the basis from which the animation was built and then caricatured.

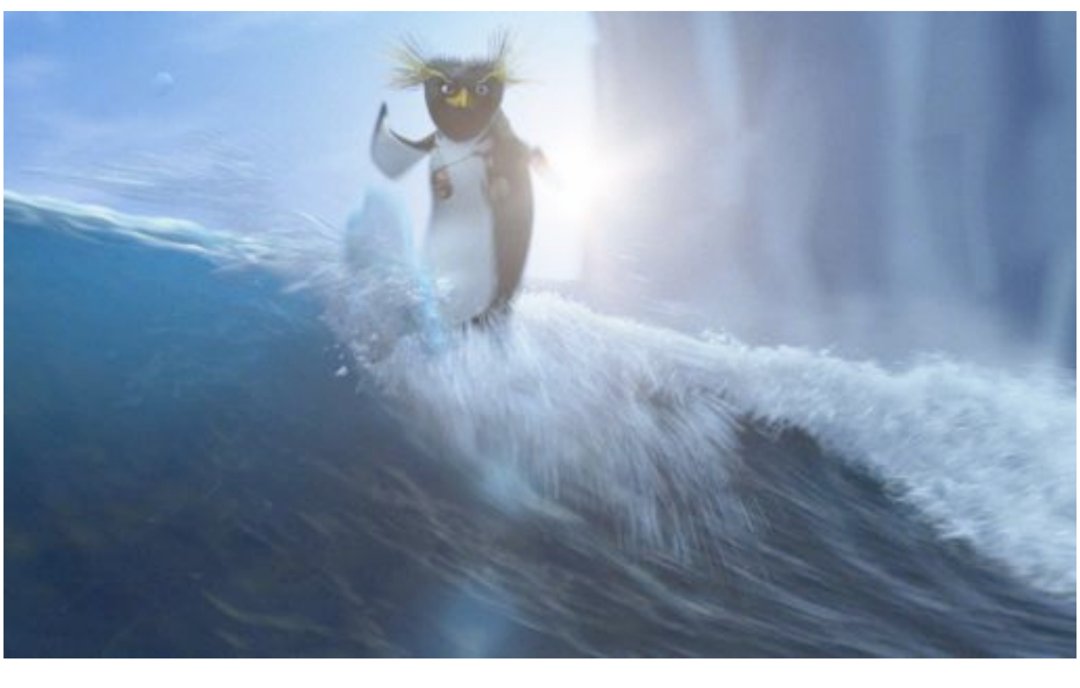
# 3.8.1 Rig Configuration

Surf footage shows that wherever a surfer puts his foot (whether it is front or back); the board will pivot around that point. As a result, the animator needs many different pivot points on the surfboard to work with. With this in mind, the surfboard rig was built with four major pivot points. From the back to the front they are "master, body, cog" and "tip." Each of the four nodes has their own offset nodes (clean channels) with movable pivots on each so that the animator could rotate the board precisely from any location.

The body of the character (penguin) is constrained to the "body" node of the board. The IK-handles for the hands and feet are constrained to the "tip" of the board so that the animator could add turbulence and jitter to the board in the final steps of animation without affecting the overall performance of the character. The hands and head are in a "master rotation" space so that when the board is rotated, the hands and head automatically counter those rotations to remain balanced to the world space around them. Offsets on these nodes gave the animator complete control over the hands and head, but the rotation constraints kept counter-animation to a minimum.

The adjustable pivot on the reverse spine of the characters (described in Section 3.3.1.1) was also used to great effect in the surfing shots. When you watch a surfer you will see that the upper body is relatively stable while the board is being thrown around underneath them. The energy emanates from the chest and drives downward. Imagine if the character were simply attached to the surfboard, and the spine controls rotated upward from the pelvis as they do in a simple forward-kinematic spine rig. Executing these kinds of surf moves would require an enormous amount of counter animation in order to keep the upper body stable while the board is kicked around and twisted from side-to-side.

Because penguin legs are so short, it was difficult to achieve the effect of the legs being the source of energy that drives the surfboard under the penguin. To achieve more dynamic surf poses, the legs were scaled a bit longer than normal, and "knee poppers" (Section 3.3.1.2) were used to sculpt more pleasing knee bends.



# 3.8.2 Constraint System

It is challenging enough to execute convincing performances on solid ground, but on a wave there is a moving "ground plane." The shape of the wave (the "stage") is constantly changing and traveling forward great distances (relative to the surfer) over time. Early surfing tests had animators tracking and chasing the wave through space, while at the same time trying to execute convincing surf poses and dynamics. It worked well enough until the animator was asked to alter the timing in any way. The smallest adjustment to the timing at any point would bury the surfer in the water (or cause him to fly off the surface of the water), and everything from that point forward fell apart. The same problem occurred if any parameter of the wave had to be changed after the surfing animation began. The animator had no choice but to delete all of his animation beyond the area in question and reanimate from that point forward.

Since animation is all about timing (and even more so about *changes*), a better system had to be developed. The solution was a simple constraint system with local spaces that travel forward with the wave and laterally (down the axis of the board) along the face of the wave. In essence, the master node of the surfboard is constrained to the forward translation node of the wave.

# 3.8.3 Surfing Technique

With the constraint system in place, a keyframe was set on the board's "master" node on the first frame and another on the last frame of the shot (but only on the forward translation). With only two keys and a constraint, the surfer would then travel (and track) along the face of the wave for the duration of the shot. At this point it was important to confirm that the speed of the surfer was within certain acceptable limits. The surfboard itself would drive the simulation of the water effects, and any errors or cheats in the speed of the surfer would be revealed later in the water simulations. It was difficult to assess whether the speed looked good in animation with flat-shaded water surfaces, and hence the need for the z-rendered water surface discussed in Section 3.7.5.3. It would be a costly mistake to move forward without confirming that the overall speed was acceptable for downstream departments, otherwise there would be no choice but to reanimate if the effects revealed that the surfer was simply traveling way too fast or too slow.

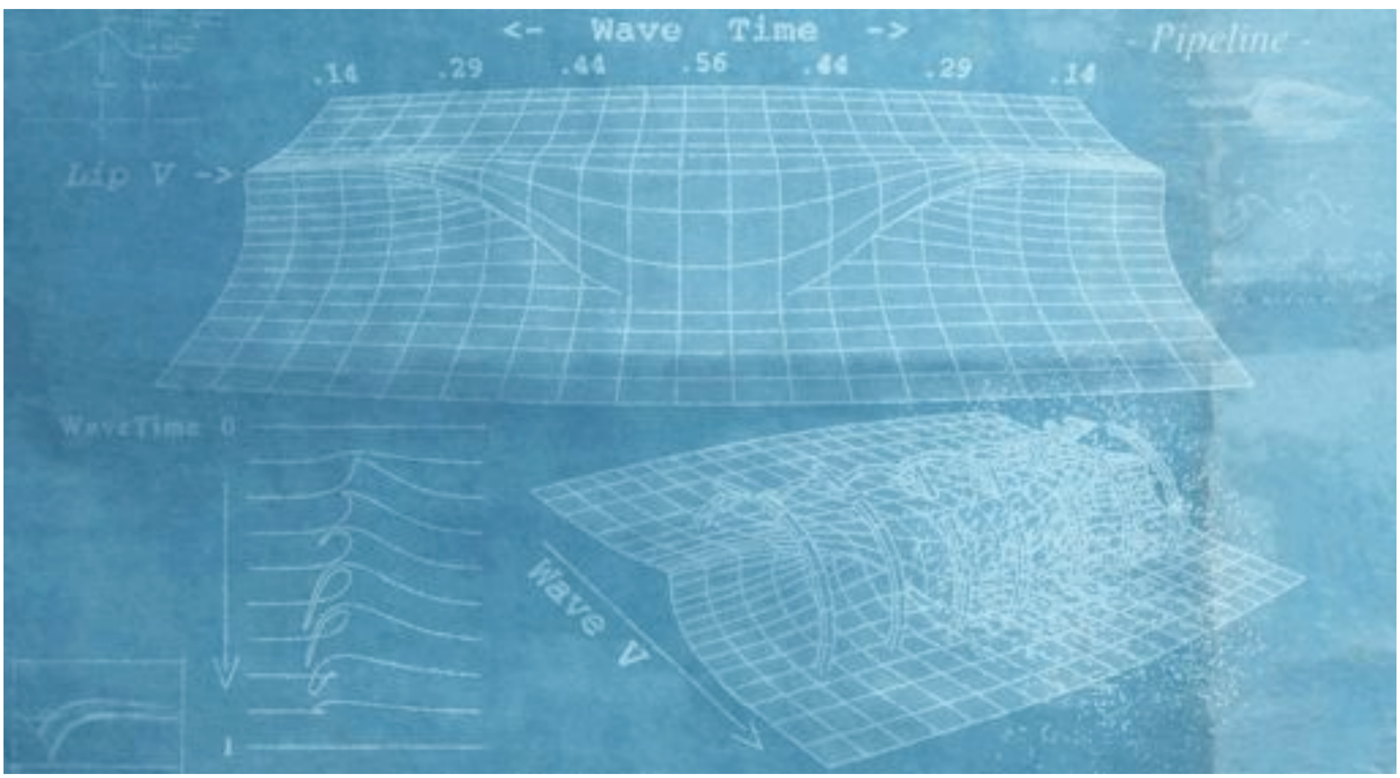
With the "master" node constrained to the wave, all other nodes (and offsets) remained in local space relative to the board which would then travel forward at the correct speed. The remaining nodes were used as offsets to deviate from (and return to) the master path. As an example, a bottom turn would be executed by animating one of the board's offset nodes off of the master trajectory path, down the surface of the wave and then cut back up to where it originally started. The same idea applied to timing changes (accelerations and decelerations) from the master speed that was set up when the two initial keyframes were defined (as described above).

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

# 4.1 INTRODUCTION

*Surf's Up* required a reworking of the production pipeline and a new set of tools to achieve the film's most technically difficult character, the Wave. This non-standard character required close collaboration between the Animation and Effects departments and re-thinking the pipelines normal flow of data between all departments. Imageworks had many challenges: create a visually realistic wave, allow for a high level of direct-ability, handle the unique interdependence and overlap between Animation, Effects and Layout, and design a production pipeline to produce the wave as efficiently as possible with the goal of delivering over twenty minutes of surfing footage.



## 4.2 Goals and Challenges

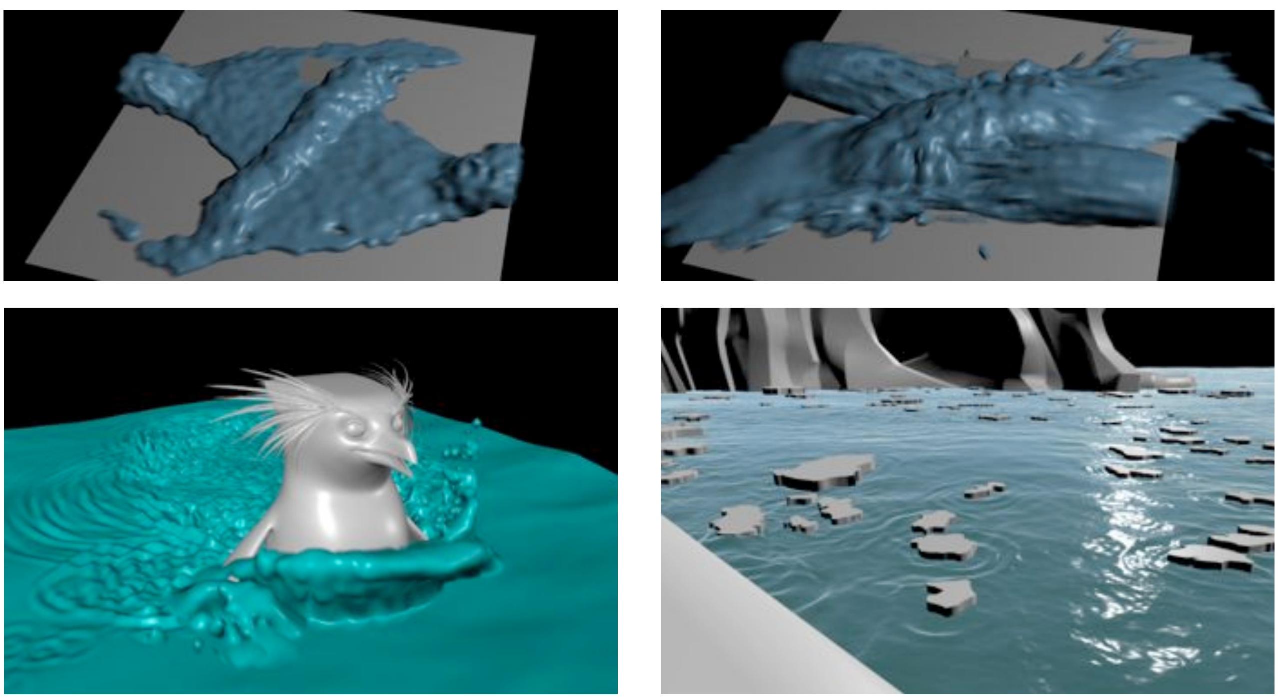
The Directors wanted the waves to be quite realistic and set a target at eighty percent real. Though most of the film's characters are quite stylized, the Directors felt the waves should behave accurately to give the audience the sensation of real surfing while introducing stylization in the color palette and surface patterns. Simplifying the complex surface properties of the wave helped to seat the stylized characters and the waves into the same world visually. In the initial research to establish the look of the waves for *Surf's Up,* it was observed that real waves are quite varied in shape, color and motion; and that it would be necessary to narrow the focus to target just a handful of wave styles and features. Based on Art department concepts and renderings, four wave styles modeled after real waves were settled upon: Pipeline, Mavericks, Spilling Breaker, and the Swell. This gave the *Surf's Up* team a more focused set of goals to meet in order to pare down the overwhelming variety of looks and motion that exist in the real world.

### 4.2.1 Slow Motion

Observing the nature of surfing documentaries, the *Surf's Up* team knew that slow-motion shots were going to be an important element in the film. All wave animation systems and particle effects systems needed to behave properly at extreme slow motion. In order to accomplish this, it was necessary to mind and track the camera speed through the entire pipeline (as mentioned in Section 2.4.1). This information was not typically tracked on previous shows at Imageworks and warranted extra care when designing the effects systems. As a result every effects system needed an extra level of development and complexity to animate at arbitrary camera speeds.

# 4.2.2 Control vs. Simulation

A big goal from the start was to make the wave motion direct-able and predictable. At first blush it appeared that breaking waves would be handled by the Effects department by running physical simulations; but the team knew from experience that simulations, while visually rich, are notoriously time consuming and difficult to control and tweak. Early investigation into existing work on simulating breaking waves was done, however, to make sure every possibility was explored. Although some impressive work was found, none of it appeared mature enough to do on a large scale in a production friendly way. The team needed a simple wave rig that animators could interact with, and adjust if necessary, in real time. As a result, a conscious decision was made early on to simulate as little as possible and to design an animation rig that would satisfy the needs of the Layout, Animation, Effects and Lighting departments.



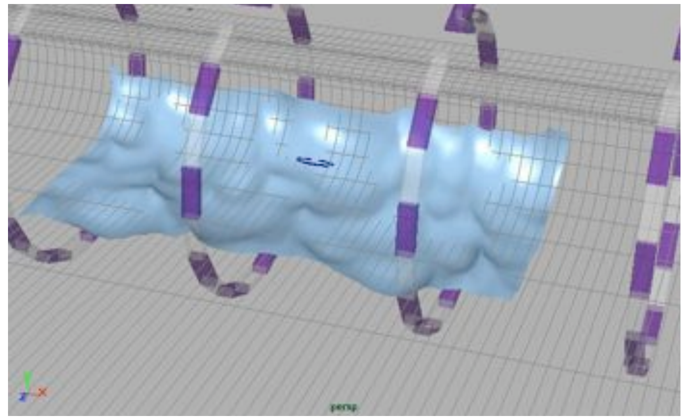
*Early simulation tests*

*First wave test shot*

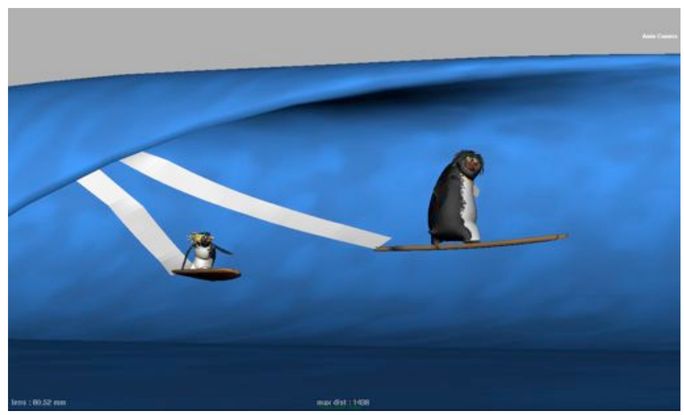
# 4.2.3 Wave Verification

While a non-simulated animation rig can provide a lot of control, it can also have the downside of being too flexible. It would be possible to animate waves in very non-realistic ways and change wave speeds from shot to shot. This could potentially cause problems with shot continuity as well as big problems for the Effects department which used the wave surface and motion as a source for dynamic, simulated secondary effects. Imageworks needed to put several measures in place to police the wave animation and insure consistency across shots. In this way, the team could avoid, as much as possible, passing unusable animation data to the Lighting and Effects departments. The wave animation was verified using pre-visualization tools of various wave features including the following:

- *Water Texture* (also referred to as *Wave Trains*) - to evaluate surface stretching and compression.
- *Wake Trails* - to further evaluate the surface motion as well as the character speed relative to the wave.
- *Whitewater Simulations* - to ensure the wave was crashing properly.



*Water Texture*



*Wake Trails*

All of these visualizations were available in the Layout and Animation stages and helped to give a more complete picture of how well the animation was working. It was essential to the secondary dynamic effects that the waves not deviate too much from plausible behavior or they would not simulate properly. Many times, animation would look good from a particular camera angle but could conceal problems that were not visible until Effects or Lighting attempted to work with the element. Various *key features* that were scrutinized included: forward motion speed, the rate the wave crashed down the line, the falling speed of the wave lip, and surface texture squashing, stretching and motion. Because of the challenge of creating a single wave that worked well for all departments, a library of verified wave animations was set up to be used as a starting place for new shots. A lot of time and care went into designing a proper wave animation.

*Whitewater*

# 4.2.4 ESTABLISHING WAVES EARLY IN PIPELINE

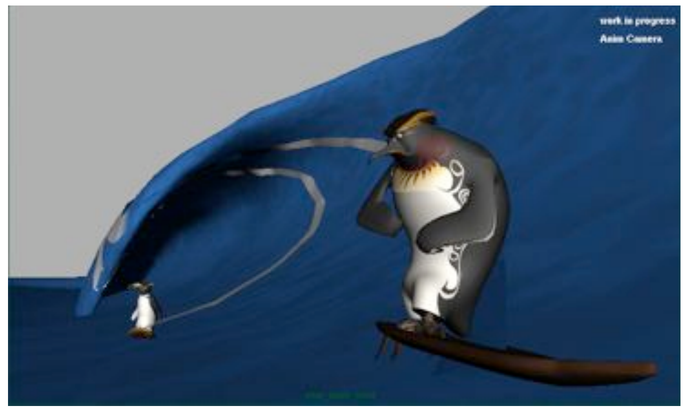
Beyond the obvious visual challenges of animating and rendering a realistic wave, the Imageworks team had to do so in a production friendly manner. Because so many aspects of a shot hinged on the shape and motion of the waves, they needed to be defined very early in the life of that shot. In fact, on *Surf's Up,* final wave animation was approved in Rough Layout, the very first department in the shot specific pipeline. The wave's shape, motion and other prominent features, such as the whitewater explosion, directly affected the actions of the surfing characters as well as camera placement, action and composition. In the end the *Surf's Up* team learned that designing a working wave pipeline was as challenging, if not more so, than achieving the wave visuals themselves.

# 4.3 Wave Shot Pipeline

The life-cycle of a typical surfing shot flows though the following departments:

## 4.3.1 Rough Layout Department

- Articulate/Final wave animation was designed to match storyboards.
- Rough character animation was done to block in surfing characters.
- Whitewater and boardwake pre-visualization tools were turned on and checked.
- A *wave train* style was chosen and published to the shot to define the character of the ocean texture.
- A shot camera was animated to complete rough blocking of the shot.
- A *wave check* render was done to validate the wave's motion, surface stretching, and ocean texture.



*Rough Layout (early stage of shot)*

It was essential that Production sign off on the wave animation and ocean texture at this very early stage. A small change in the wave's animation or limit surface created huge ripple effects downstream for camera placement, surfing animation, and secondary effects animation. This was a big reason why the team built so many pre-visualization tools to help us more completely evaluate the wave prior to moving out of the Rough Layout department. The whitewater and boardwake tools were used by the camera operator, to help compose the shot; character animators, to design the characters motion; and the Effects department to validate that the wave was moving in a plausible way to support secondary dynamic effects. The *wave train* style was published and each *wave train* system (built into Maya, Houdini and RenderMan) used this information to describe the ocean texture, in order to keep everything in sync.

# 4.3.2 Animation Setup Department

- The wave, camera, props, and surfing characters were all prepped for animators.
- A *z-depth* render of the wave was generated to give animators an accurate visualization of the exact ocean surface.
- The wave geometry was translated to disk along with support assets used later by the Lighting and Effect departments:
    - Whitewater particles.
    - The *Crash Curve* which described the lip of the wave (position, energy, velocity, wave time, etc.).
    - Utility images were created to marry the wave patch with the greater ocean surface at render time.

The *z-depth* render was simply a depth-based image of the wave surface rendered from camera through RenderMan with full displacements turned on. This, coupled with the "beauty" *wave check* render, was loaded into an animation scene and "z-composited" with the character rigs. In this way an animator could check the exact intersections of the character against the displaced ocean and wave in near real-time. The whitewater system, described in more detail later in this document (Section 4.4.7), could use the extra data translated at this stage to animate and tweak the whitewater without loading in the wave animation. Everything the whitewater system needed to do its job was baked into the *Crash Curve* and stored on disk.

*Z-depth Wave Check Render*

## 4.3.3 Animation Department

- At this stage, the wave was more or less considered a "static" environment.
- Rough surfing animation was polished to final.
- Characters were baked and translated to disk in Imageworks' proprietary geometry format.

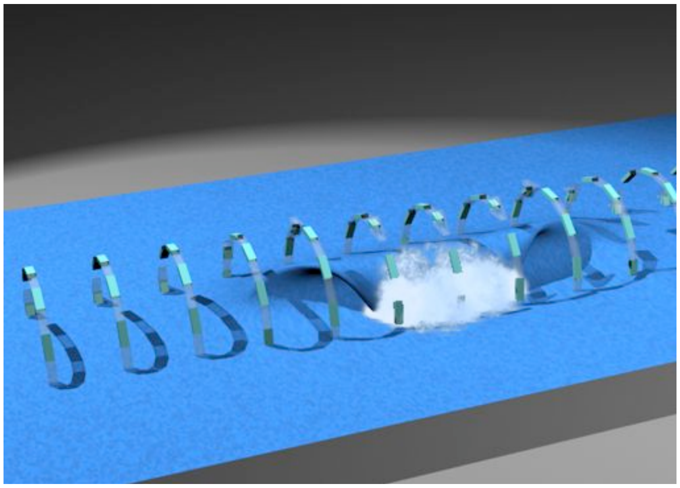
The wave rig was originally designed to be tweaked by animators at this stage; however, after discovering how easy it was to deviate from plausible wave motion, it was decided to lock down the wave prior to character animation. On rare occasions a decision was made to change the wave at this stage and the approach depended on the impact to production. If it was a minor change it was done in place, but if it was larger, the wave character was kicked back to the Rough Layout department to execute.

## 4.3.4 Final Layout Department

- Final camera was touched-up.
- Camera and static environment was translated and published for Lighting and Effects.

The *Surf's Up* pipeline was specifically designed to keep data moving downstream. It is very easy on an effects heavy film to get into a feedback loop where one department is constantly kicking animation back to the previous department to fix problems. The goal was to minimize this as much as possible through visualization tools and production procedures. Front loading and locking down the wave animation in the first department in the pipeline was critical in keeping the shots moving forward.

# 4.4 Wave Rig



*Basic Wave Rig*

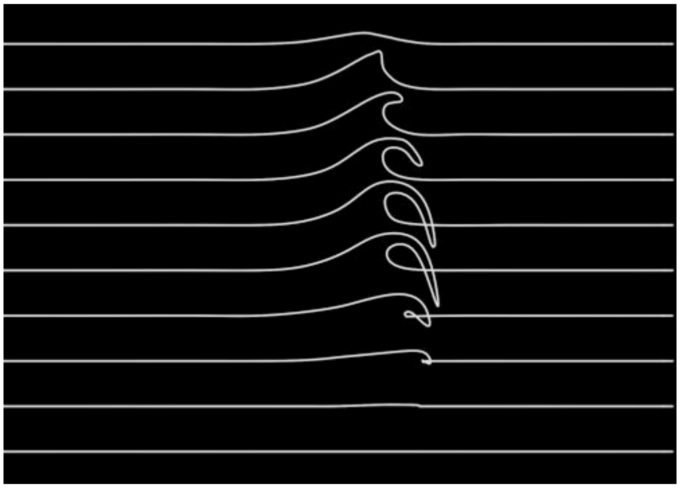
The waves of *Surf's Up* were not simulated, but instead constructed using a procedurally driven animation rig designed by both the Effects and Animation departments. The rig was built in Maya and needed to satisfy specific goals for each department. Animation needed quick feedback, intuitive controls, and to conform to rig standards set up for all character rigs at Imageworks. This allowed the wave to flow through the animation pipeline like any other, more traditional, character using the same methods and tools to import, upgrade, animate and export the wave character. The Effects department had its own set of requirements as the wave surface drove several dynamic and non-dynamic effects such as whitewater, lip spray, and wake trails. The Lighting department relied on consistent topology and secondary attributes to work hand in hand with surface and displacement shaders designed specifically for the wave.

# 4.4.1 WAVE GEOMETRY

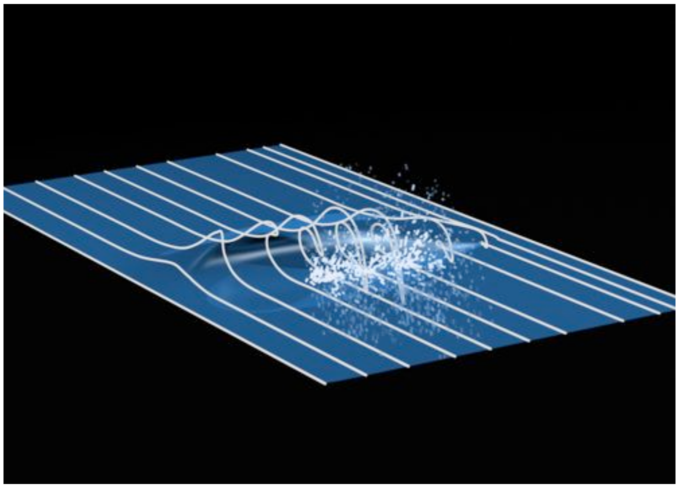
A single cross-section of the wave was modeled by hand using a NURBS curve profile along with several target blendshapes to simulate that profile's life-cycle over time. About eleven blendshape targets approximated the full life of a breaking "pipeline" wave from birth to resolve. A single 0-1 *time* attribute was rigged to drive the wave though all of these targets. In addition to the life-cycle shapes, other blendshape curves were introduced to further shape the wave; controlling aspects like lip thickness, depth of the trough, wave height, slope of the front face, etc. At the heart of the *Surf's Up* wave is this single curve with multiple blendshape targets to mimic various wave shapes and behaviors. Not only was the curve shape important, but the placement and interpolation of the CVs over time was critical for driving how ocean texture moved, stretched, and compressed over the wave surface.

A series of these blendshape cross-sections were placed in a row and lofted to form the single wave surface. Independent control over each cross-section's *time* attribute was maintained allowing different parts of the wave to be more, or less, evolved. In the beginning of the wave's life (*time* = 0 at every cross-section) the geometry was simply a rectangular patch. As *time* along the length of the wave patch was increased the wave shape began to evolve and deform out of the center line of the patch.

Imageworks built a custom lofting plug-in to pass attribute data stored on the cross-sections to the resulting surface. The *time* attribute was passed to each point on the surface as well as an *energy* attribute. Each blendshape target marked a particular moment in the life of the wave from first formation, to spilling over, to closing out. An arbitrary *energy* value was assigned to each stage of life to approximate the amount of power the wave possessed at each moment in time. This attribute was also passed down to the wave surface to be used later by the whitewater system. This fairly simple system was the basis for the wave surface; the next step was to create simple controls to manage and animate the large number of cross-sections in a coherent manner.

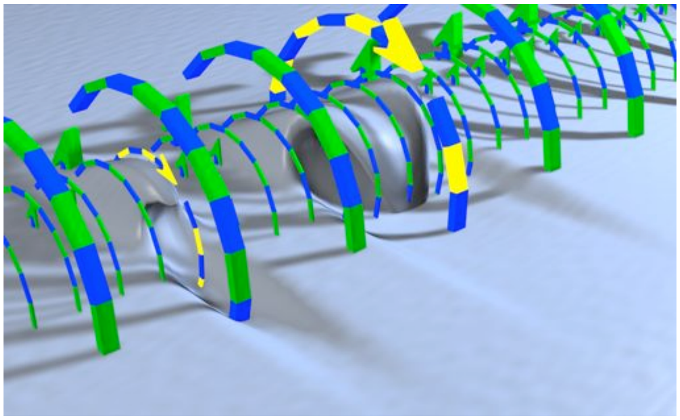


*Wave Curve Profiles*

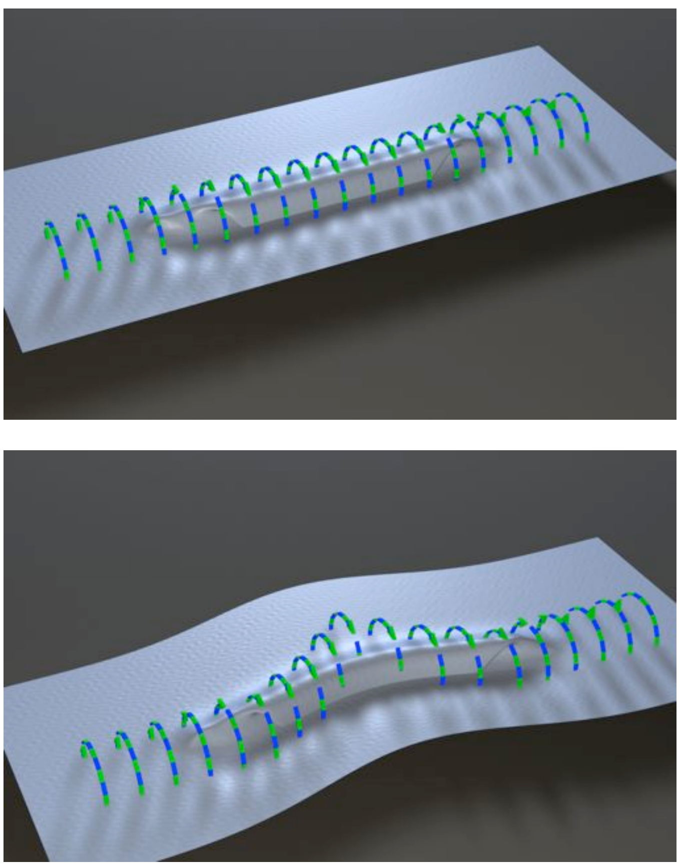


*Lofted Curve Profiles used to make wave*

## 4.4.2 Animation Controls



*Coarse and fine control rings*



*Translation and bending of wave through translation of control rings*

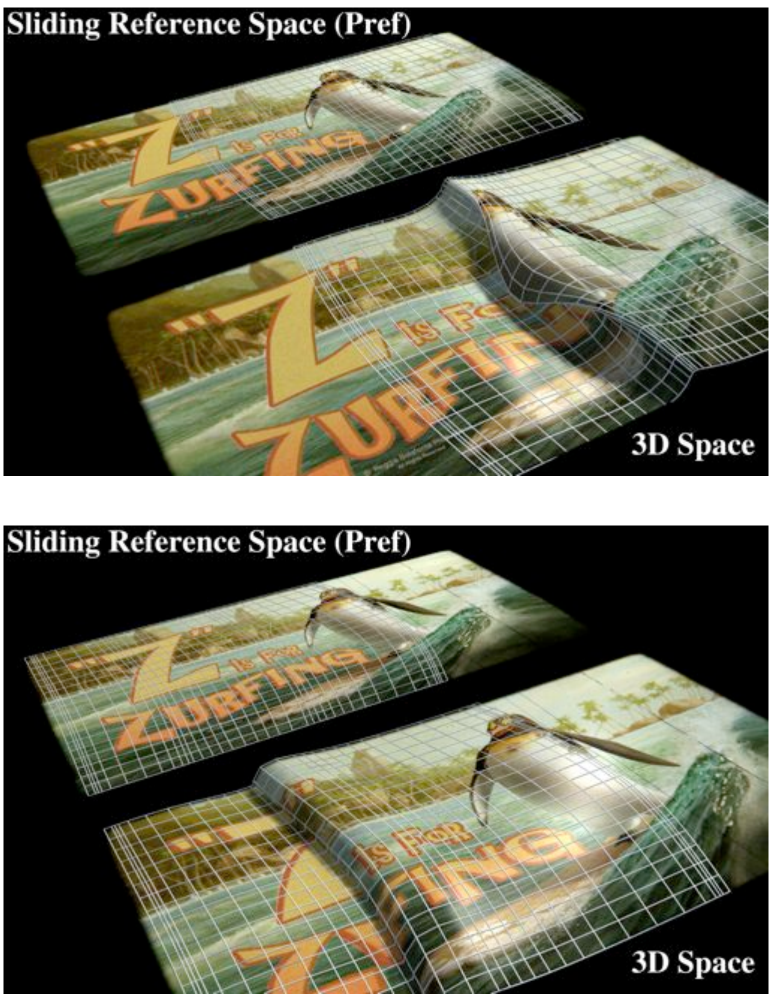
Effects and character TDs designed a hierarchy of control rings to drive *time* at different levels of granularity allowing large areas of the wave to evolve with a single control, or allowing for fine grain tweaking of small areas. The *time* attribute was splined between control rings allowing for a nice transition between crashing regions and newly forming regions. Simply by rotating these rings forward and back an animator could intuitively shape and animate the wave by evolving or devolving various sections. At each control ring other aspects of the wave could be modified as well; secondary blendshape modifiers could be introduced, or noise to procedurally vary the profile of the lip. The wave mesh was dense enough where not every cross-section was directly controlled by *time* or other shaping attributes. Some cross-sections were simply shape interpolations between the nearest two procedural ones. It was found that simple point-based shape interpolation, in combination with edge-length and angle interpolation (as outlined in the SIGGRAPH 2000 paper, *As-Rigid-As-Possible Shape Interpolation* by Marc Alexa, Daniel Cohen-Or and David Levin) worked best to minimize the shape collapsing in areas of extreme twisting, caused by quick changes in *time* between adjacent cross-sections.

The control rings drove the local shape and evolution of the surface but the overall forward motion was simply a parent transform used to move the wave though space. The wave could be bent by translating control rings forward or back relative to this parent transform.

The animation controls did not place constraints on the artist to design plausible motion. The wave could certainly be tweaked in unrealistic ways. The team relied upon a careful testing phase to mimic, as best as possible, how waves move; analyzing the forward speed, rate they break down the line, speed the lip pours over, etc. This testing was done for each wave type, and the result was stored in a library as a starting place for shot animation.

# 4.4.3 Wave Space

Because the *Surf's Up* wave geometry was a single patch moving though space, a special system was set up to show how water surface features moved over the wave. Attaching a simple ocean texture map is not very telling if it is simply locked to the local UV coordinates of the patch. The texture needed to slide over the surface as the wave progressed forward. The team built a secondary animated reference space on a point attribute called *Pref* (position reference). *Pref* for the *Surf's Up* wave was a version of the wave that was completely flat (*time* = 0, devolved) but moved though space at the same rate as the wave. World space point positions of the *Pref* wave were used as UV coordinates to map ocean texture to the primary wave geometry. In this way, the ocean texture appeared static as the wave current moved under it. By flattening the wave in the reference space, streaking was avoided on the Y-axis that would have been seen with a straight world space texture look-up. Any texture, particle effect, or object that was bound to the waves used this *Pref* space to approximate what happens in reality when waves pass under floating objects.

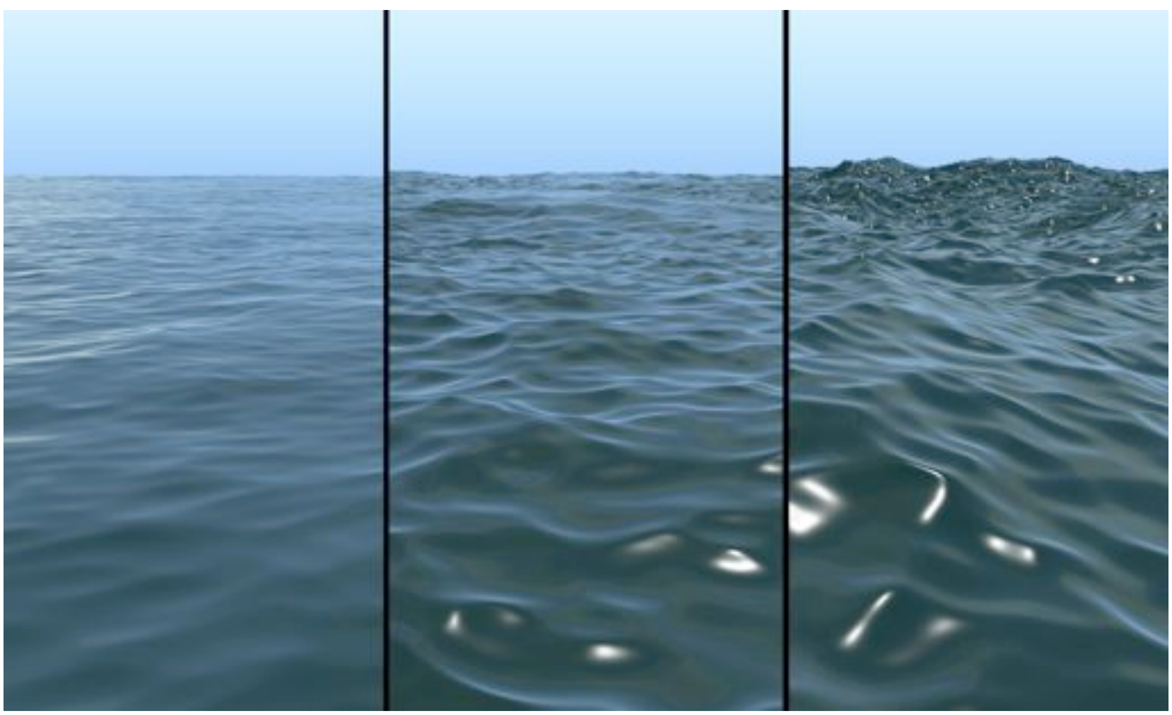



*Pref space on a moving wave*

# 4.4.4 Wave Trains

*Wave Trains* refer to Imageworks' system for generating the open ocean texture. They are the surface patterns that travel over, and displace, the waves and ocean; the details of which are described in the next Section of this document. Wave Trains do play a roll in all departments, beginning with Rough Layout, because they have a significant effect on the ocean and wave displacement. Tools to view Wave Trains were built into Maya to allow the camera operators and animators to see an accurate representation of the surface, in order to make sure surfboards had adequate contact with the ocean surface and that the camera didn't dip below the water line, unless directed to do so.

A custom deformer node was built to access the Wave Train system and was applied to a small, high resolution grid which was displaced on the fly within Maya. This small grid could be placed anywhere on the wave or ocean surface to reveal a section of truly displaced Maya geometry. The Wave Train system spanned RenderMan, Maya and Houdini. The RenderMan plugin was the primary tool used to displace the ocean surface at render time, while the Maya and Houdini visualizers were tools to pre-visualize and match the final rendered surface. Each plugin linked to a common external library to ensure the wave surface position was identical across packages.

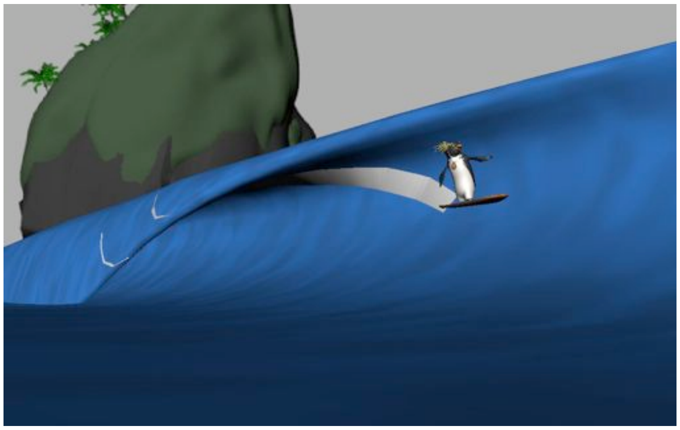


*Wave Train examples*

# 4.4.5 WAVE RIDERS

*Wave Rider* was a term used to describe special locators constrained to the wave patch using Maya's geometry constraint. The geometry constraint allowed the locator to move freely over the surface, but not off the surface, and turned out to be an ideal tool to aid in attaching rough character or camera animation to the wave. This was generally not used to drive final animation but was a good tool to block out animation. Attached to the Wave Rider was a Wave Train patch which was a convenient way to slide the Wave Train surface over the broader wave surface. In addition, a tool to visualize the surfboard's wake was also attached to the Wave Rider locator.

# 4.4.6 Wake Visualizer



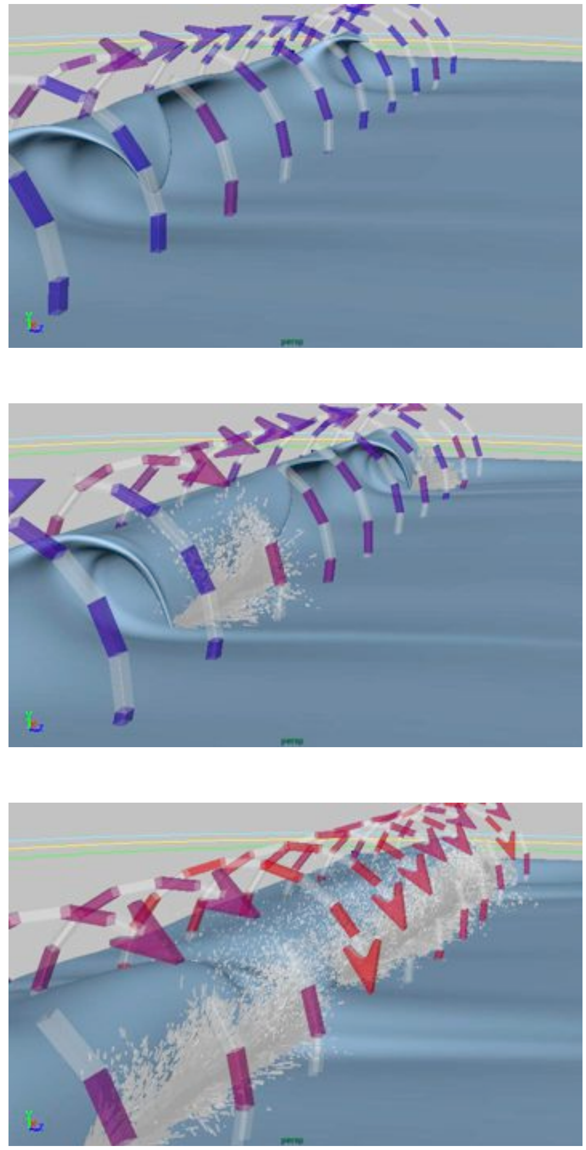
*Wake Visualizer*

The *Wake Visualizer* was a simple dynamic system that dropped a single particle per frame behind the Wave Rider locator node. Each particle ray-intersected to the nearest 3D surface location on the wave patch and anchored itself to the corresponding location in *Pref* space. In this way, each particle slid over the surface of the wave as if the wave were rolling under it. The particles were then lofted into a line to give an accurate geometric representation of how the surfer's board wake and wake history would look. It was important to have this tool when animating the characters in order to gauge the relative speed of the surfer vs. the water surface. If a surfer crossed the wave too slowly the wake trail would travel up the wave very quickly, perpendicular to the board's direction. This was caused when the wave's forward motion was too fast relative to the surfer's lateral motion. If the surfer's speed and wave speed were more in line the wake would trail roughly parallel to the board's direction.

*The Final Render*

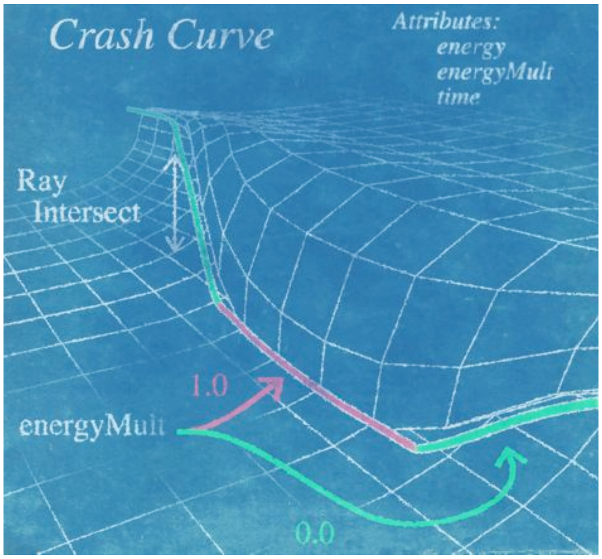
# 4.4.7 Whitewater System

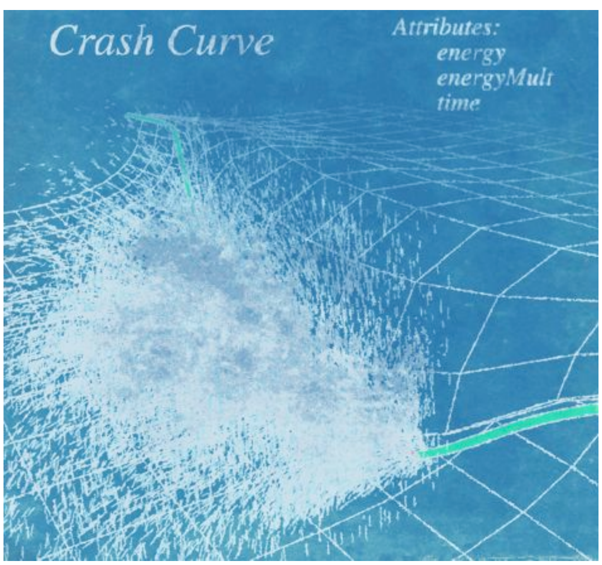
Imageworks felt it was essential to lock down not only the wave geometry, but also various wave features that would dictate camera and surfing motion early in shot production. To achieve this goal, a custom Maya plugin was written to visualize whitewater at the time of wave animation. This allowed for much more complete pre-visualization in the early Rough Layout stage, which resulted in more accurate and dynamic camera moves and compositions. Placing a dynamic simulation in the hands of the Layout or Animation departments was not usual practice; instead, the team built a non-dynamic particle system that responded in real-time to user input without the need to do run-up simulations. The goal was to build a decent approximation of the general volume and motion of final-quality whitewater, while keeping it speedy enough to not hinder animation. The system ended up being not just a good approximation, but particles generated from this plugin were eventually used as a source for final quality whitewater renders.



*Whitewater in Animation Rig*

The Maya plugin was written to draw particle sprites directly to the Maya view-port without creating any geometry. This was achieved by writing a custom locator node in Maya which allowed OpenGL calls to be sent directly to the view-port. All controls to tweak the look and motion of the particles were built into this single plugin node. By handling all the drawing and dynamics inside a single plugin the team was able to maximize the speed of simulating thousands of particles. In order to keep the system fast and interactive, some inputs were cached while others remained dynamic. The two primary inputs to the plugin, a pre-baked **particle clip** and the **crash curve** were cached in memory over time by the whitewater plugin.




*Crash Curve intersection*

The **particle clip** was a simulated cross-section of a larger whitewater system. A clip's size was quite small, only about 500 centimeters across, and its behavior was dialed to have the explosive, clumpy, and staccato nature of whitewater. Clips were created in Houdini at the origin and once approved baked to disk as a time loop-able simulation with gravity forces removed. A clip typically consisted of about 150 frames of animation and somewhere on the order of two thousand particles.

The **crash curve** was a NURBS curve that acted as the emission source for whitewater with special attributes (*energy*, *wave time*) stored at CV locations to control where, and how strongly, particles should be emitted. The crash curve was generally supplied by the wave, running its length and bound to the lip. Its *energy* attribute was controlled by the size and impact locations along the wave, in addition to *energy modifiers* placed on the control rings of the wave rig.

Using these sources, the whitewater plugin cached up this data, over time, in memory. Once cached, the particle clip was instanced end-to-end along the arc of the crash curve, transformed to the position, rotated to the curve's normal at each location, and scaled by the curve's energy. Consequently, particles only appeared where energy was non-zero or areas where the wave was actually crashing. The particle clip was not simply deformed to the curve; each particle contained a *life* attribute which told the system how many frames it had been alive. Using this information the plugin attached each particle to the crash curve's location at the time of that particle's birth, not the curve's position on the current frame. For example, if a particle's *life* was 10 frames old it was transformed to the crash curve's position at time ***current frame - 10***. Since the entire history of the curve's position over time was cached in memory it was relatively quick to look this information up on a per-particle basis. The age of the particle was used to apply other dynamic forces as well: *gravity*, the curve's *velocity*, *drag, wind, energy noise, etc*. All of these attributes could be tweaked at any frame in

# 4.4.7 Whitewater System cont'd

real-time without the need to re-simulate previous frames, all the dynamics were applied analytically. In this way, it was possible to stop on any frame in the simulation, dial the *gravity* or *drag*, for example, and see the affect in near real-time. Another bonus was that baking the particle system out to disk did not have to happen in a linear fashion and could be distributed over several machines.

Under the hood, the system was fairly complex, but in practice the team built the wave rig to handle the details fairly automatically. The animation rig came with a crash curve built in to collect data about the wave's position, energy, velocity, and impact areas. The wave's crash curve derived the *energy* attribute at each location along the wave based primarily on wave *time*. Each blendshape target for the wave was assigned a particular *energy* value; as the wave evolved and crashed so did the *energy* value. This value was multiplied by a second Boolean value which identified areas where the wave actually hit the bottom, or trough, of the wave. This was done by a ray-intersection test of the lip to the trough. As a result, a wave could be animated, the whitewater turned on, and the animator would see a fairly convincing whitewater animation without any adjustment to the plugin parameters.

The resulting particles were translated and used directly by the Effects department to produce final quality whitewater. These points acted as seed points to a custom instancing plugin in RenderMan that produced many millions of points.

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

## SURF'S UP

# WAVE EFFECTS

# 5.1 Introduction

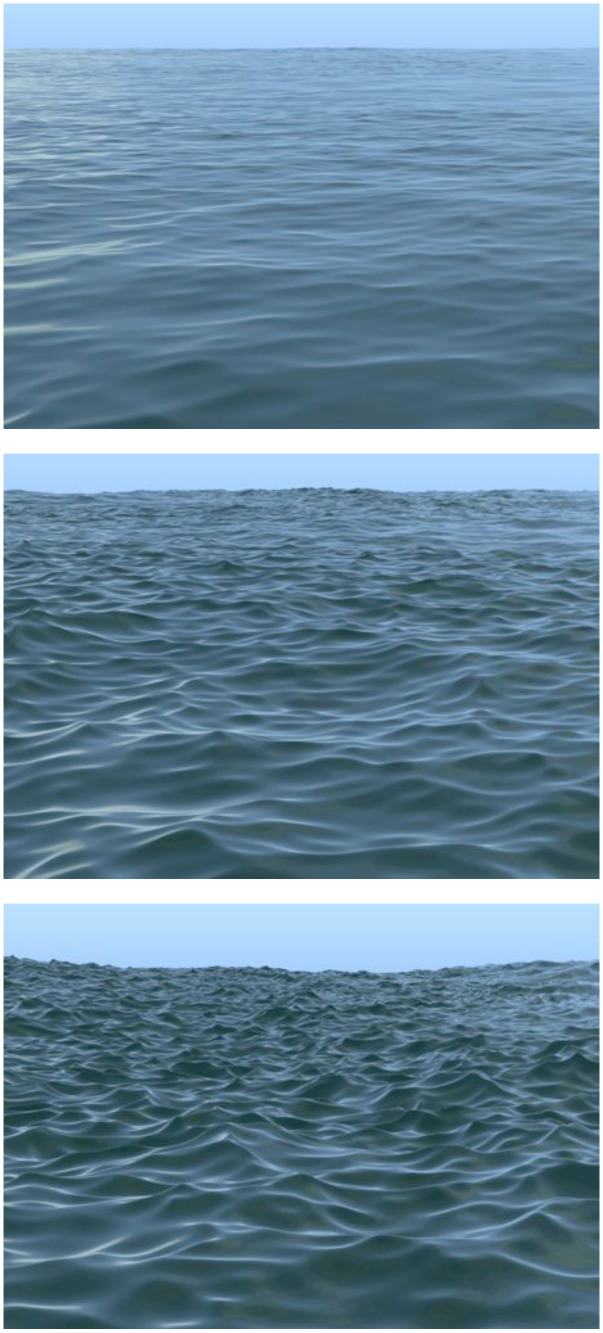
The goal of the Effects team was to create water and waves more realistic than not; and to do so, with regard to the surface motion of the waves, in a non-dynamically simulated manner in order to enable maximum flexibility in the Layout and Animation stages of production. While this approach circumvented the difficult task of figuring out how to physically simulate a tubing wave, it created its own set of technical and aesthetic challenges, especially concerning the creation, synchronization and integration of primary wave features. These features included whitewater, lip spray, the foam ball, surface stretching and the distinction between the wave and the surrounding ocean. What follows is an overview of the key technical strategies the Effects team used to create the surfing waves of *Surf's Up*.

# 5.2 The Wave and the Ocean

The animation wave rig produced a deformed rectangular mesh representing the shape and motion of an isolated surfing wave but not of the entire ocean surface visible in a given shot. One of the first decisions facing the Effects development team on *Surf's Up* was to determine what kind of surface or surfaces would be employed to render the hero waves and surrounding ocean. Did it make more sense to render the wave geometry output from the animation rig directly or to build a method of modifying an ocean surface by that geometry? After some initial testing trying to arbitrarily deform large ocean grids by the wave geometry coming from Animation, it was decided that the best solution would be to consider the hero surfing waves and the surrounding ocean as separate objects that would seam together. Although initially there were fears about problems at the transition between the wave and ocean, it soon became clear that having a rectangular wave patch that was easy to navigate parametrically was highly desirable and that seaming the wave and the ocean together could be achieved relatively simply by rendering an opacity hole in the ocean where the wave was, while insuring a small geometric overlap between the surfaces. This was achieved by rendering, from an orthographic camera, a map of the UV space of the wave patch cast into the flattened space of *Pref* (more on *Pref* later). Included in these tiff-encoded UV projection maps was the transform matrix of the camera which was used in the shader to position the textures correctly back onto the geometry during render time. Using the projected UV space of the map, the shader cut a hole in the ocean surface and managed the opacity overlap between the wave and ocean creating a seamless transition between the two.

In a division of labor that evolved over a period of months, development of the water displacement shader was handled by the Effects department while the Look Development team maintained responsibility for the water surface shader. As a result of this arrangement, patterned wave surface features, such as foam, were defined in the displacement shader and passed along to the surface shader for distinct material property definition.

# 5.3 Wave Trains



*Three styles of wave trains: calm (top), choppy (middle) and choppier (bottom)*

The function of the wave rig was to provide the gross animation of the surfing waves but it was not designed to provide the small, higher frequency waves of the water surface. That was another assignment for the Effects department. Inspired in part by Tessendorf's work, a Houdini and RenderMan based system for simulating open ocean waves was developed for the overall displaced surface of the water. The system employed "wave trains," simply defined as the sum of continuous wave patterns of varying period, amplitude, direction and speed. By creating sets of Gerstner-style wave trains whose speeds, by default, were physically based but whose frequency ranges and angles of propagation were hand-tailored, several water surface "styles" were settled upon. These ranged from almost dead calm to stormy and chaotic. The frequencies of the wave trains were segregated into three ranges: low, medium and high; each with individual control over amplitude, cuspiness, and speed. Provision was made for general noise-based and specific hand-tailored control of areas of amplitude reduction of the wave trains for a more varied and natural look of the ocean surface. The peaks of waves could be determined and isolated in the shader to create areas of aerated water or to be used as the source of emission for particle effects. Data, output from the simulation system, describing the frequency ranges and propagation angles and speeds of the wave trains was input into the water displacement shader for rendering. Alternatively, tileable displacement maps of the wave trains were created and employed for both final rendering and pre-visualizing of the wave trains in Maya during Animation and Layout.

# 5.4 Foam

Foam, ambient or created from crashing waves, splashes, surfboard wakes or shore break, was a critical component of the look of the water in *Surf's Up*. From the start it was important to create methods for general and specific foam placement, erasure, dissolution and animation. Used not only to create a more realistic look, different foam patterns and formations were employed to distinguish wave styles and locations from each other. Three distinct patterns of foam were designed from live action reference and consultation with the Visual Development department: A *patchy foam* used for choppier water and splashes from rocks and characters, a more elegant graphic style referred to as *web foam* taken from specific photographic examples and used with calmer water at the North Beach location and a *convected bubbly foam* used with the beach break system of small waves lapping at the shoreline.

A distinction was made between "standing" foam, foam that was generated with procedural noise functions in the shader, and "interactive" foam, foam that was specifically placed or the result of a specific event like a splash or wake. Interactive foam used the same noise functions as standing foam but was placed on the water using point clouds sampled in the reference space of the water with attributes describing search radius and density. Once enough points, collected in the reference space, crossed a density threshold, foam would appear in the additive space of the points' search radii on the corresponding part of the wave surface. Additionally, and similar to the methods of amplitude reduction of the wave trains, areas of foam could be erased or reduced procedurally with noise fields or specifically with artist-designed maps projected onto the ocean surface.

*Patchy foam generated from the interaction of the water and the bone yard rocks*

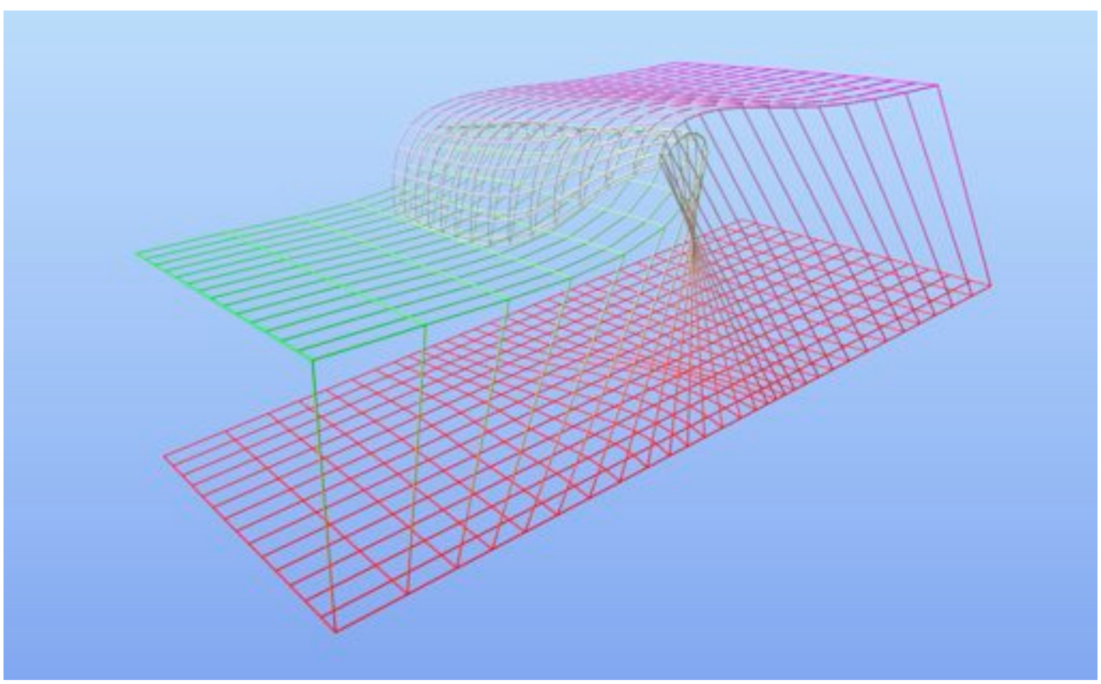
*Standing web foam on a tubing Pipeline wave*

*Convected foam in the beach break*

# 5.5 Wave Geometry Spaces

The surfing wave geometry was essentially a hand animated blend shape solution modeled and moved in such a way as to appear to generally obey a narrow subset of real-world wave characteristics such as forward velocity, lateral break speed, a lip that fell at a speed close to gravity, plausible volume preservation and corresponding surface stretching. A handful of driven parameters output from the wave rig provided essential spaces in the wave geometry for the Effects team to generate whitewater, foam features and surfing effects. These spaces were represented by the following attributes:

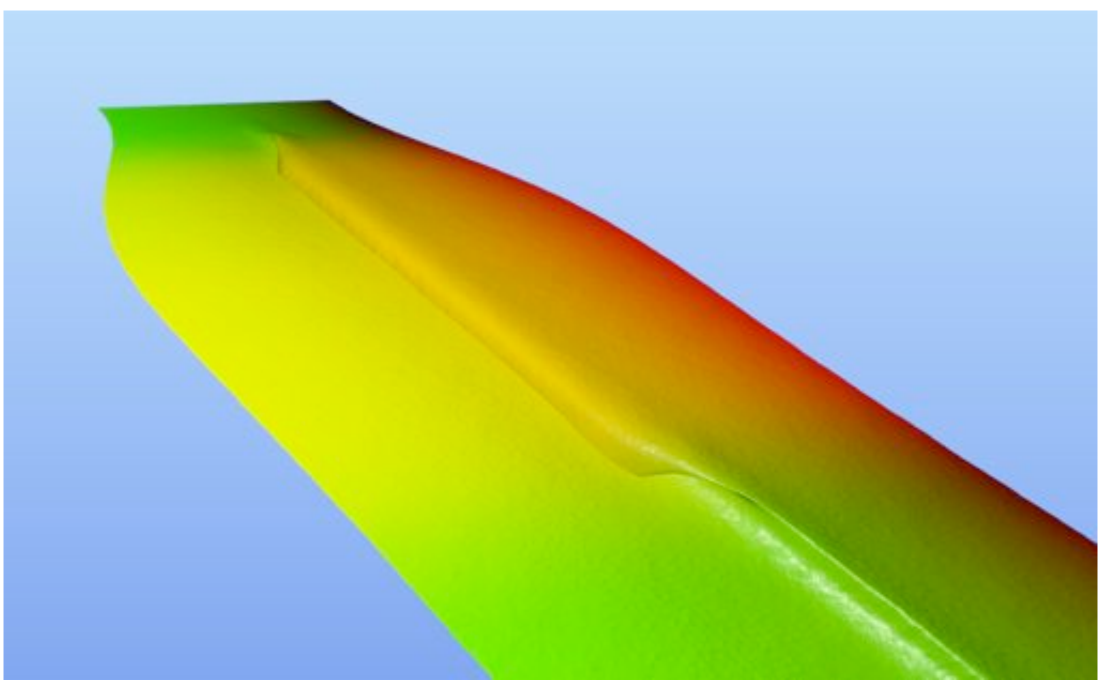
- *Pwave* - the world space coordinates of the wave geometry;
- *Pref* - a flattened reference space that moved along with the wave;
- *Wave Time* - a varying value in the evolution of the wave at any given cross-section parallel to the forward motion of the wave;
- *Wave V* - the V direction in the parametric space of the wave mesh typically perpendicular to the forward motion of the wave;
- *Shoulder* - a value at a given cross-section of the wave parallel to its forward motion representing the portion of the wave patch, at its lateral extremities, that does not break;
- *Energy* - a value assigned to each cross-sectional blend shape in the wave rig and propagated through to the wave mesh representing the force of the crashing wave; this was used primarily for producing whitewater.



*A mapping of Pwave (the curved wave shape on top) to flattened Pref space below*

Separately and in combination, these spaces were employed in creating a wide variety of effects and surface features. The two dimensional space of *Wave Time* vs. *Wave V* was especially useful for many effects such as creating "lip spray," the spray of water ripped back off the falling lip of the wave. The lip of a wave was a constant value in *Wave V* and would vary in *Wave Time* through the evolution of the wave, making it easy to segregate part of the wave geometry as a source for particle emission. This wave space was also used in creating region specific foam features like "curl foam," one aspect of the forward falling part of wave, and "back foam," the region of foam on the back of the wave after it crashed.

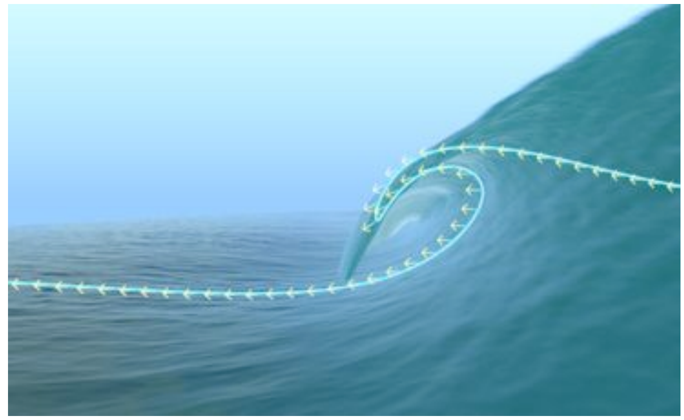
A particularly difficult problem in creating a convincing-looking crashing wave from a series of cross-sectional blend shapes was designing the way surface features, such as wave trains and foam, moved and stretched as the wave traveled through the water. *Pref* solved this problem by providing a reference space where anything cast into it could be projected onto the wave surface and, given plausible wave animation, would result in realistic stretching and movement across and up the deforming wave. In a process similar to the tweaking of UV texture spaces for animated characters, hundreds of hours were spent hand tailoring *Pref* spaces for each wave style (Pipeline, Mavericks, Spilling Breaker and the Swell) to produce realistic motion of the surface features of the wave. In order to achieve a seamless blend between the character wave and the surrounding ocean the *Pref* position at the edge of wave equaled the *Pwave* position of the ocean ensuring that no discontinuities between the two would occur.



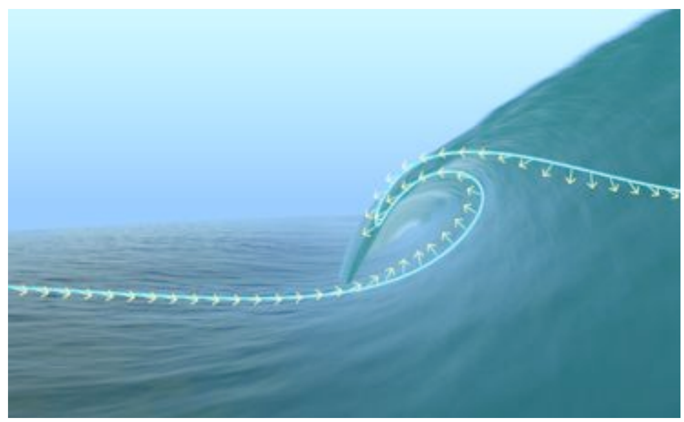
*The space of Wave Time (red) vs Wave V (green)*

# 5.6 Creating Correct Motion Blur

Creating correct motion blur for the wave was also problematic because the motion of the character wave was not physically based and, more importantly, because though the animated geometry represented a plausible shape of the wave it did not represent plausible motion of the surface of the water as the wave moved through it. Traditional deformational blur or simple trailing velocity calculations would not provide correct results in all regions of the wave. The water on the inside vertical face of a tubing wave was moving forwards and upwards but once the wave had tubed frame-to-frame the vertices of that section of the geometry simply moved forwards and thus would not blur correctly. The boundary of the character wave moving forward in space is best thought of as a moving window on a static ocean, so in order to achieve correct motion blur the surface velocity vectors were computed as if the geometry were static with a wave deformation moving through it. Having both current and next frames' wave shapes (the waves' position in *Pwave* space) and undeformed shapes (the waves' position in *Pref* space), the mapping from the next frame's *Pref* shape to its *Pwave* shape was used to deform the current frame's *Pref* shape to the next frame's *Pwave* shape and produced more accurate velocity vectors for the surface motion of the wave. The area of exception was the forward falling face of the wave which, with the technique just outlined, would appear to be blurred in the opposite direction desired, up and backwards over the wave instead of forwards and down. To get around this, for only the vertices of the forward falling face, a vector representing the overall forward motion of the wave was subtracted, the resultant vectors reversed, and the forward wave vector added back in. A similar computation was done in the shader to the texture space on the falling face of the wave to give the appearance of downward motion.



*Incorrect motion blur vectors produced by a deformational blur of the surface*



*Corrected motion blur vectors*

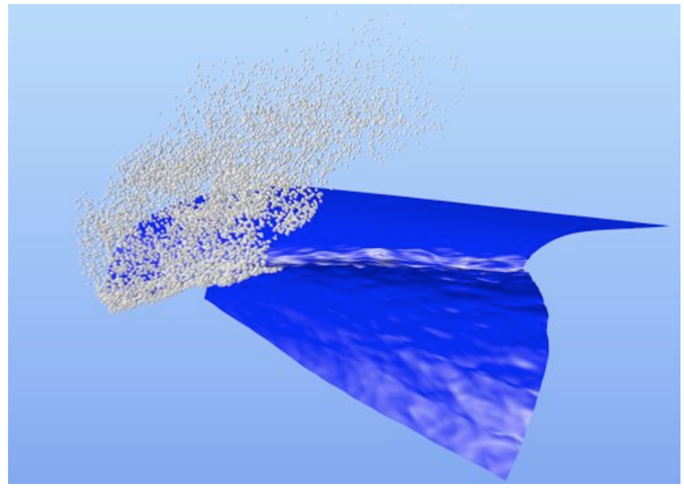
# 5.7 WAVE PARTICLE FEATURES

There were three primary features of aerated water of the tubing waves in *Surf's Up*:

- *whitewater*, the large forward explosion in front of the wave caused by the lip crashing into the trough,
- *lip spray*, the spray ripping back off the lip as it falls,
- *foam ball*, the backwards explosion of the water inside and around the tube.

While each of these features were rendered in similar ways, producing the particle simulations required distinct methods that relied on different sets of the attributes and wave spaces described in the previous two Sections.

# 5.7.1 LIP SPRAY



The lip had an additional wave train displacement, called *lip trains*, introduced at a certain time in the waves' evolution to roughen up the surface as if the lip was beginning to break apart as it fell over the wave face. The peaks of these lip trains' displacements were colored brighter in the shader as if the water was beginning to aerate. These lip train peaks were the source of the lip spray. Because the low resolution wave geometry did not include wave or lip train displacements and the lip spray particles had to align perfectly with the lip surface, increasing the resolution and pre-displacing the source geometry of the lip spray was required. This was accomplished by carving, in *Wave V* space, and "up-rezing" a strip of the wave around the lip, rendering, in the normalized UV space of that carved strip, a map of its final displaced position and then deforming the carved strip by that map. The result was a hi-resolution strip of geometry representing the final wave and lip trains displaced surface that also included a lip trains *peak* attribute. The lip spray particle simulation was birthed from this geometry with emphasis placed on tight, densely clustered particles close to the lip, dispersing into mist as the spray was carried up and over the wave. Integrating the particle simulation with the wave surface so that it appeared as if the wave was actually breaking up into spray was difficult especially on Pipeline waves, the smaller of the tubing wave styles. If the lip spray was too dense right at the wave surface it could appear pasted on, but not having enough particles would give away the fact that the "water" was not really breaking apart. In addition, on many shots, the waves were moving very quickly which made it tricky to get just the right density of lip spray correctly matted by the wave surface at the lip. To help the transition from wave to lip spray, a technique to vary the amount of occlusion of the wave over a short initial span in the life of the particles was implemented.

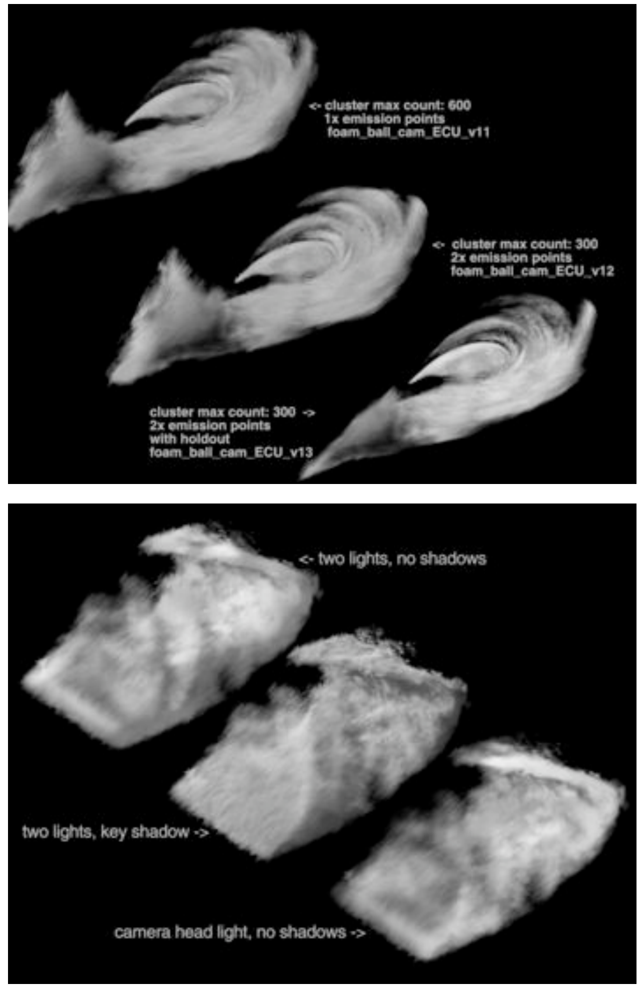
# 5.7.2 WHITEWATER

Because the whitewater explosion was such a large visual feature of the wave, pre-visualizing it in Maya during Layout and Animation, for timing and composition, was a requirement from the beginning of the production. Another requirement, though little utilized, was the ability for character animators and layout artists to affect the behavior of the whitewater, adding or subtracting energy from the explosions. Because of these specifications the development and methodology of creating the whitewater particle simulations was radically different from the lip spray and foam ball, and its particulars are outside the scope of these notes. Some general concepts are worth discussing, however.

The source of the whitewater was a line, carved along the lip of the wave, called the "crash curve." The vertices of the crash curve held an attribute, *energyMult*, which would toggle on wherever the lip had crashed into the trough. Those vertices also held an attribute, *energy*, which represented the potential energy of the wave that was greatest when it first crashed and lessened the further it evolved. Multiplying *energyMult* times *energy* would provide the initial velocity magnitudes for the whitewater simulation. The specific motion of the whitewater, the direction of dispersion and pulsing nature of the simulations, as well as the addition of a second lower section of whitewater called "the skirt," which helped integrate the effect with the wave and ocean, were all artistically directed and not physically based. Gravity, drag and the speed of the whitewater were manipulated in non-real world ways to enhance the size and impact of the waves. An additional element called "whitewater mist" was also created to help diffuse and soften the sometimes hard particulate nature of the whitewater renders. A secondary particle simulation birthed from the whitewater, the whitewater mist was designed to drag and hang in the air above the wave and was rendered using Imageworks' proprietary sprite-based renderer, Splat.

# 5.7.3 The Foam Ball

Similar to the whitewater, the particle simulation for the foam ball was birthed from the crash curve. But because the foam ball swirls around the inside of the tube of the wave, a different approach was taken in its construction. For some of the surfing and wave effects, the forward velocity of the wave made it difficult to have precise control over the particle simulations. One solution was to halt the wave, create the particles, and then add the forward velocity of the wave back into the simulation and adjust the velocities accordingly. This was done with the foam ball with the additional twist that the simulation was done in the flattened space of *Pref* and then warped back into the curved space of the wave. This was done primarily to simplify the collision calculation of particles with the wave, it being easier to collide with a static, flat limit surface than the deforming tube of the wave, but also to have better control over the speed of the foam ball. Moving the crash curve and particle simulation back and forth from wave space to *Pref* was done with some custom barycentric deformation operators.




*Early foamball testing*

*Final rendered frame*

# 5.8 RENDERING A LOT OF POINTS

All the spray effects in *Surf's Up*, the whitewater, lip spray, foam balls, surfboard sprays, character splashes, rock splashes, etc., were rendered, entirely or in part, as dense clouds of RiPoints calculated at render time in RenderMan. To accomplish this, a proximity-based particle instancing scheme, called Cluster, was developed as a RenderMan DSO. The instancing algorithm produced new points along and around the vectors between pairs of seed points from sparse particle simulations with many parameters controlling point size, distribution, density, opacity fall off and attribute blending. Because the final particle counts required for whitewater or lip spray in a given shot would most often exceed the memory limitations of the machines on the render farm, methods for rendering subsections of the elements had to be developed. A scheme for slicing Cluster renders into layers based on distance from the camera plane and managing the compositing of those layers was implemented. Shadow renders also required the slicing scheme, despite the use of techniques for reducing the point counts, and increasing the point size. However, as they were rendered without motion blur twice as many points could be generated. A screen space dicing scheme, to increase the point count of a single render, was introduced later in the show and added an optional approach that would, under certain circumstances, allow passes to be rendered without slicing. The trade off of dicing versus slicing was that although dicing could render a frame in one pass and often faster than the cumulative render times of the sliced renders, since it was a singular process it couldn't be parallelized. Throughout the show the Cluster DSO was optimized in an attempt to render as many points as possible and ultimately was capable of rendering without dicing or slicing about 45 million motion blurred points. For a big Mavericks wave shot in *Surf's Up,* the combined point count of the whitewater, lip spray and foam ball could easily reach 500 million points.

The clustered effects were lit using deep shadows which were rendered from each light; typically a key, rim and fill. The final beauty render of the element was a "utility" pass with equal contributions of the key, rim and fill lights segregated into the RGB channels of the image to be balanced and color corrected into the shot during the compositing stage (see Section 6.4). Extra passes for specular glints, particle life and density variation were also provided to increase the detail of the element. To save time, especially during sliced renders, matting of other objects was handled by rendering deep shadows of the occluding geometry from the shot camera and sourcing them into the whitewater and spray shader for opacity variance.

# 5.9 Summary

In retrospect it seems odd that a computer generated movie with so much water in it would have been made without the use of a fluid solver at any point. But that fact underscores the overall methodology used by the Effects and Animation teams on *Surf's Up* which was initiated in response to the following quandary: how to efficiently create a lot of realistic looking surfing waves in a production pipeline where the primary animation of the waves occurs during Layout and key features of the wave need to be pre-visualized and altered during Animation. As more sophisticated ocean simulation techniques become available, CPUs become faster and memory more expansive future answers to this question may not rely on many of the strategies outlined above. However, given the demands of an animated feature, where keeping creative and technical options open for as long as possible throughout the pipeline is strongly desired, the approach of layering linked, yet discreet solutions to the primary wave features proved highly successful.

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

**SURF'S UP**

# WAVE SHADING

# 6.1 INTRODUCTION

Water and waves were an integral part of *Surf's Up*. Waves were not merely a part of the environment, but almost characters themselves. This chapter focuses on the role of the Look Development department and the techniques used to render the wide variety of wave and water styles. The rendering and compositing techniques used to achieve the documentary style of filming the surfing action will also be presented.

# 6.2 Artistic Goals: Realistic vs. Stylized

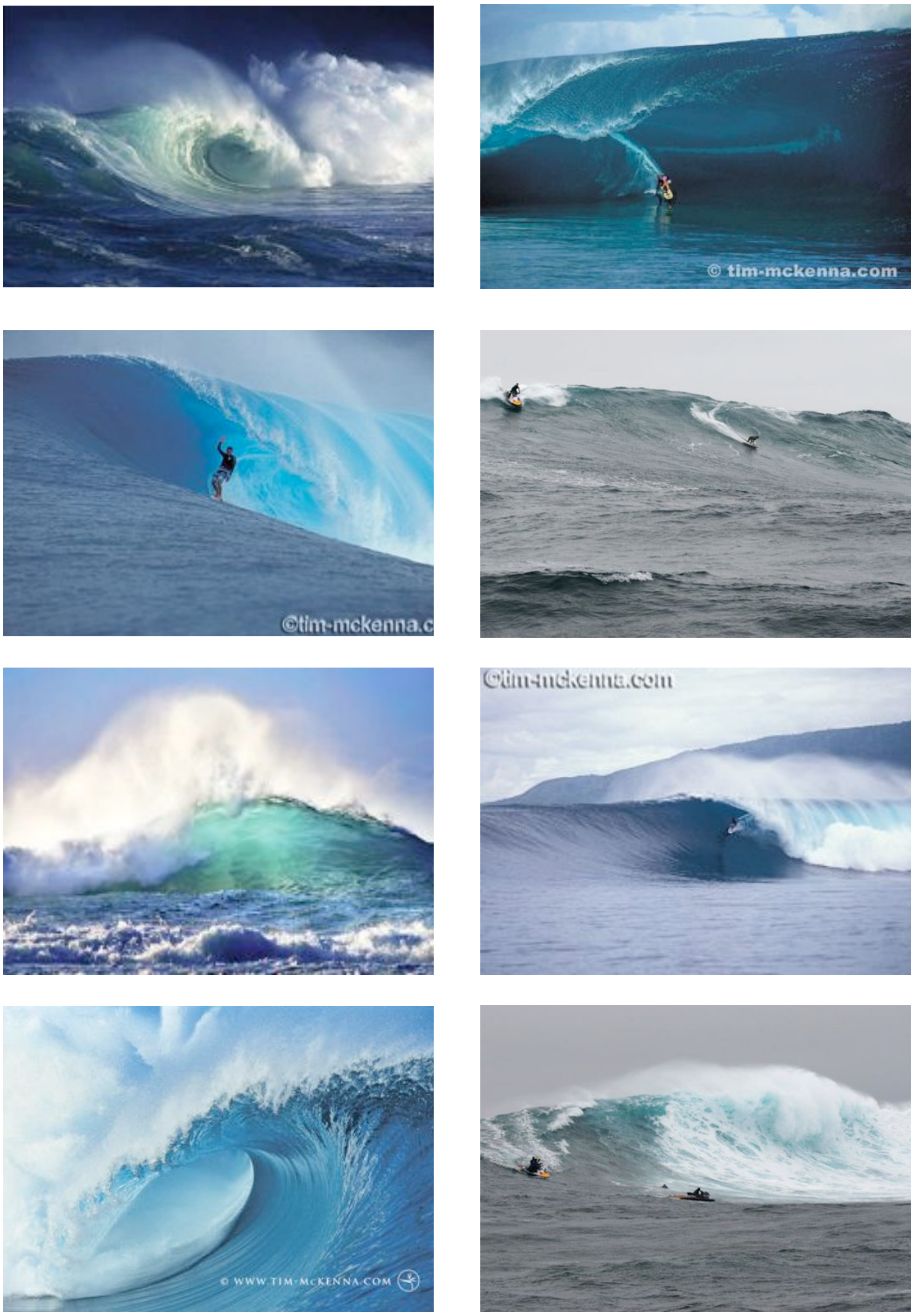
Although the characters in *Surf's Up* are quite stylized, they live in a realistic and believable world. In order to capture the excitement of surfing it was important to achieve a fairly high level of realism. The viewer is in the action, surfing with our characters, and experiences the beauty of being in the perfect tube as well as the horror of wiping out on a monster wave.

In addition to this realism, the movie contained a wide variety of waves, water and locations. The wave styles ranged from the cold dark waters of Antarctica to the tropical waters of Pen Gu Island and finally to the stormy and ominous waves of competition day. These waves were also filmed in many different lighting situations.

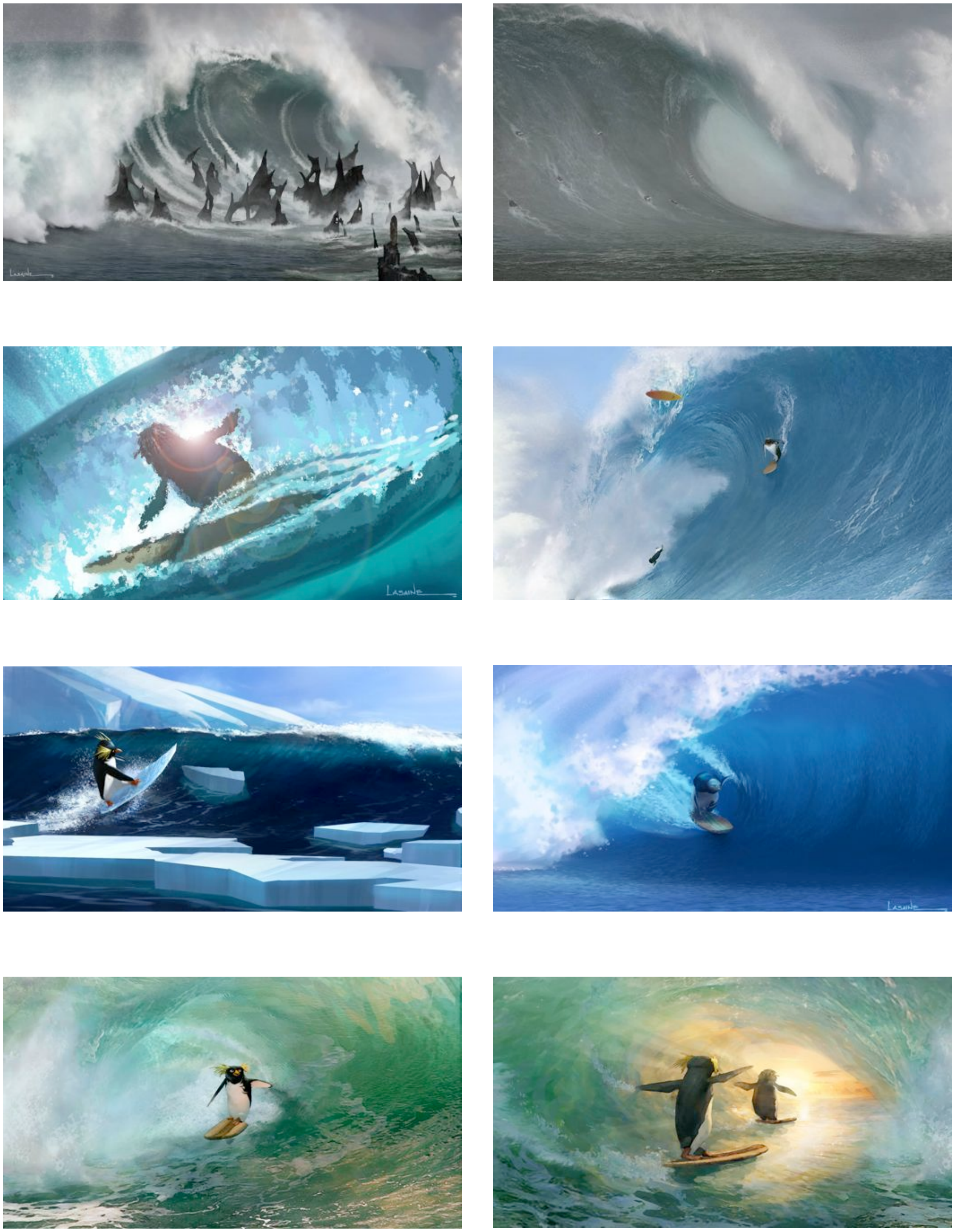
The style of the waves was inspired by surfing and outdoor photography and the conceptual art created by the Visual Development team. The following images represent a sample of this reference imagery.

# 6.2 Artistic Goals: Realistic vs. Stylized cont'd

*Examples of reference photography*



*Examples of Visual Development artwork*



# 6.3 Rendering the Water: Wave Zones

Various physically based rendering techniques were considered but it quickly became obvious that the film required a more flexible way to render these waves in order to achieve the visual style and variety required.

A shading technique was developed that made it possible to render realistic waves with a high degree of art direction. This technique employed a local coordinate system in which a series of wave zones were calculated using the cylindrical coordinate system of the wave and the parameterization of the wave surface. These zones were separated by angle and represented the various parts of the wave during its life cycle. Each zone could be shaded separately and all zones could be animated and blended. Lighters were able to manipulate these zones in a way that allowed them to almost paint the color of the waves. The image below illustrates these zones.

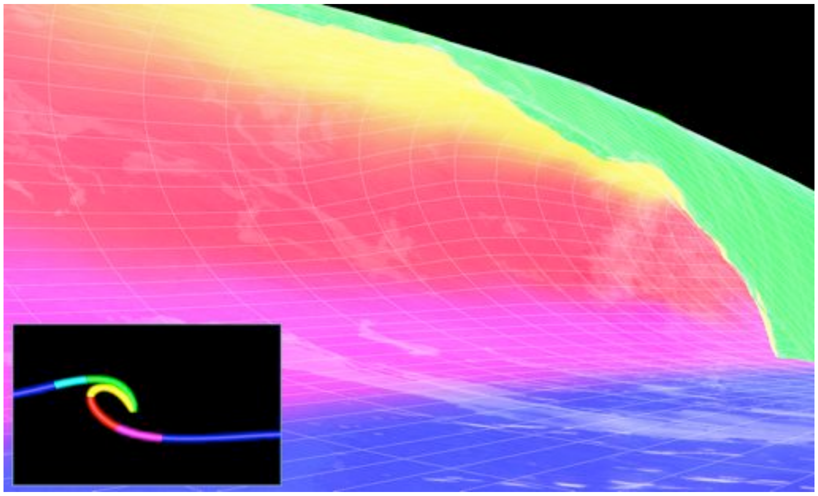


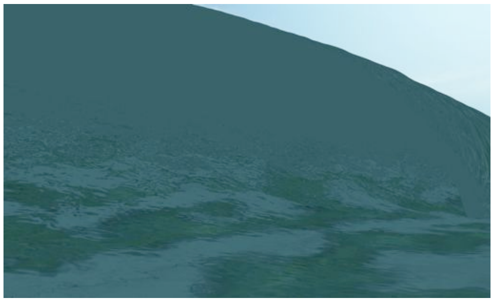
*Diagram illustrating wave zones*

This method was developed based on the observation that waves break in a similar way and that they have a similar shape when considered from a profile. The angles formed by the normal of a surface point and the Positive Y axis are similar across the waves types. Wave zones were created based on this angle. The parameterization of the wave surface was normalized with the 0.5 value being equal to the leading edge of the breaking wave. Values less than 0.5 represented the bottom of the wave and those greater than 0.5 represented the top.

Each wave zone was treated as a separate (albeit simplified) material. These material zones were combined in the shader through blending regions. The surface color derived by these zones ended up being a good approximation of the diffuse and ambient lighting components of the wave. This was combined with other more physically based techniques for the reflection and refraction to achieve a high degree of realism.

The foam patterns were created in the displacement shader by the Effects team (see Section 5.4). This procedural foam pattern was "message passed" to the surface shader. In the surface shader, the top foam (above water) and under foam (below water) were treated as separate and unique materials.

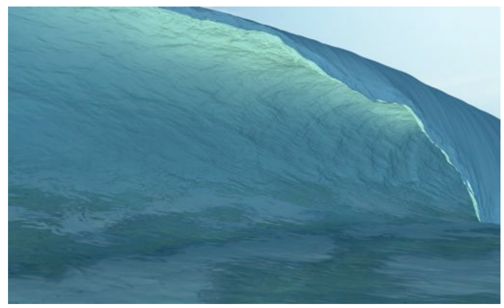
*Composite of beauty renders adjusted using wave zones as control maps.*



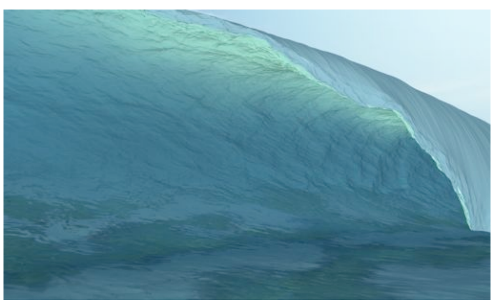
*Bottom map pass*

*Diffuse pass*

*Incandescence pass*



*Translucence pass*



*Color correction pass*

*Specular pass*

*Cast shadow pass*

*All layers composited*

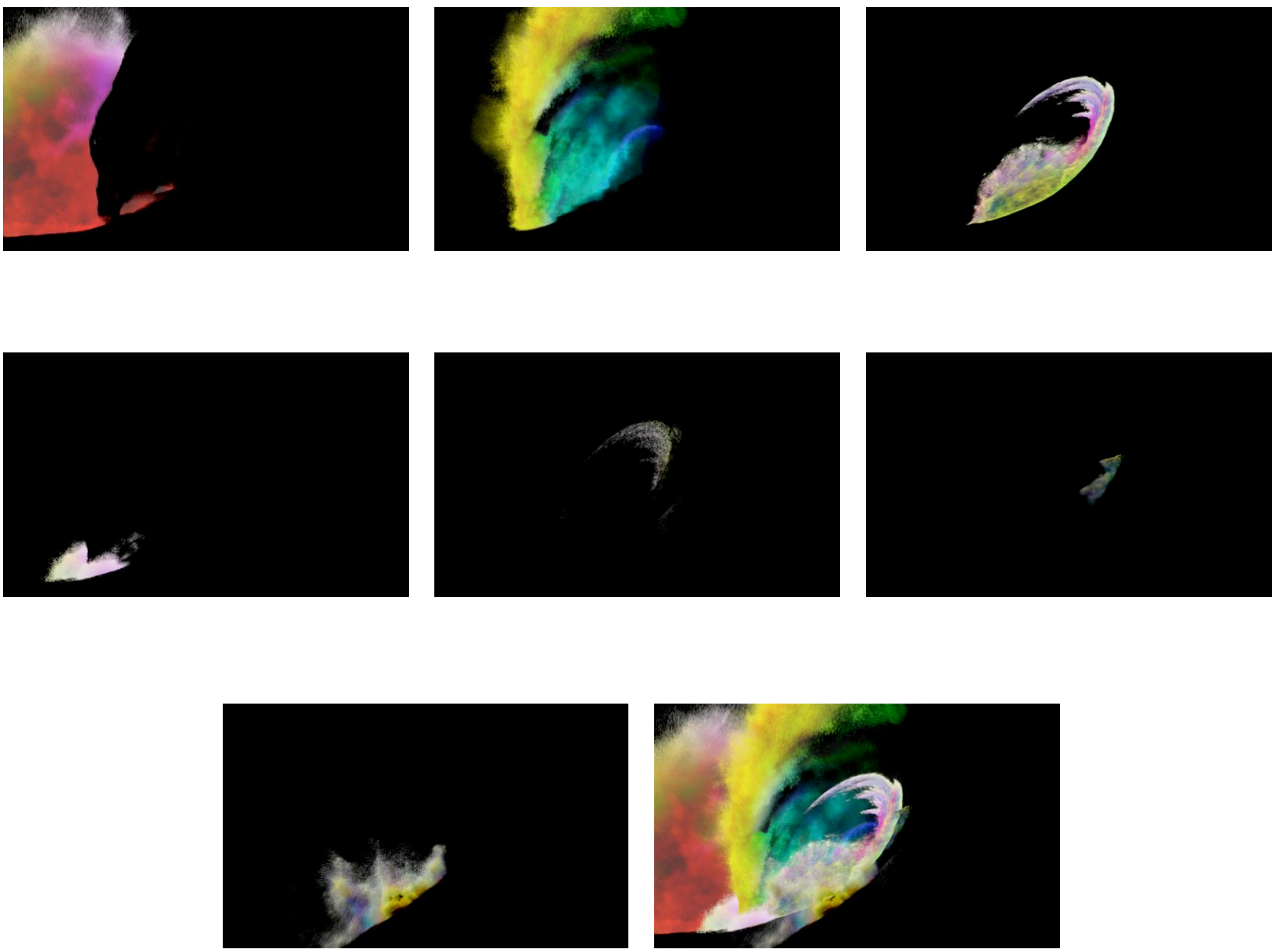
# 6.4 Rendering the Particles

Particles were rendered for all spray and whitewater elements including the following:

- Lip Spray
- Surf Board Spray
- Wave Whitewater
- Paddle Splashes

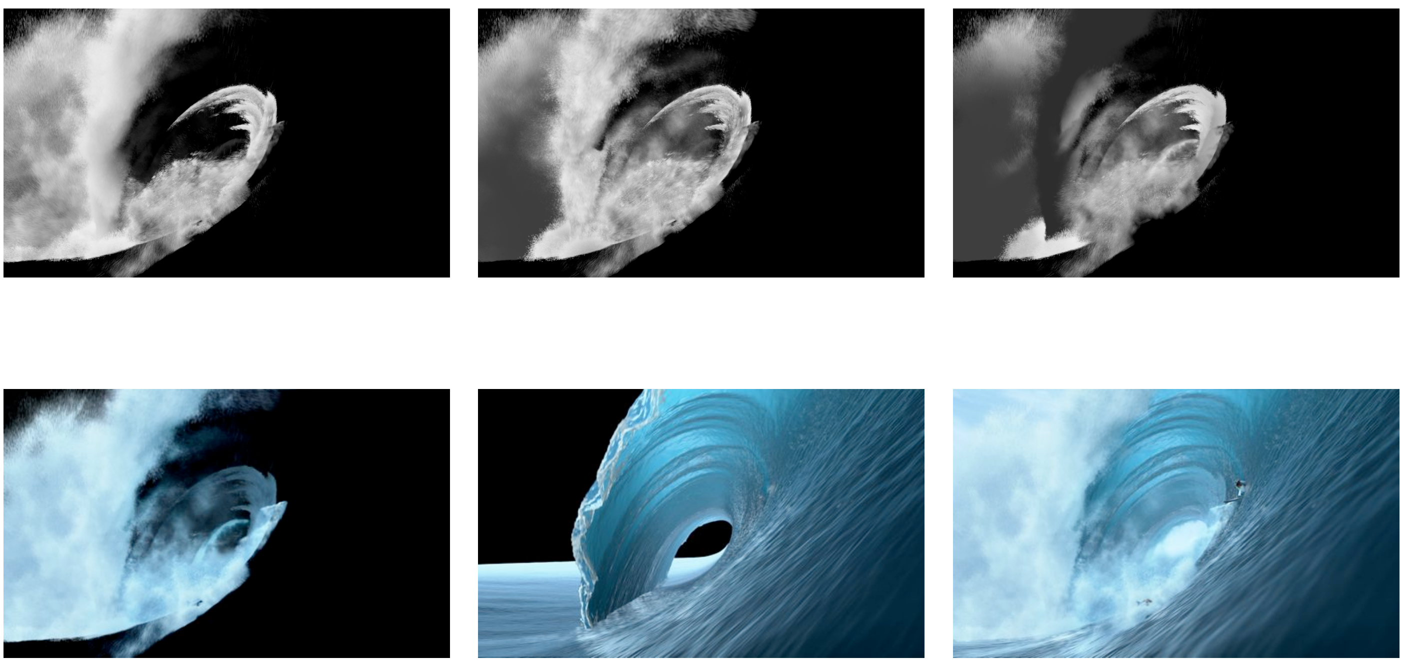
These elements were rendered as RiPoints in very large numbers as described earlier in Section 5.8. Many of these particle elements were rendered by the Effects department using Z-depth images as holdouts and then passed along to Lighting for final integration.

*Individual particle renders lit by 3 lights – each light rendered to a separate R, G, or B channel*



*Proximity* and *time* attributes for the particles were used to transition the particles from clear refractive water to a white aerated appearance. This was especially useful for the interactive splashes and sprays from the characters and their surf boards. The images below show the elements that were used by the artist in the composite to create these splashes.

*Composite of all particle elements*



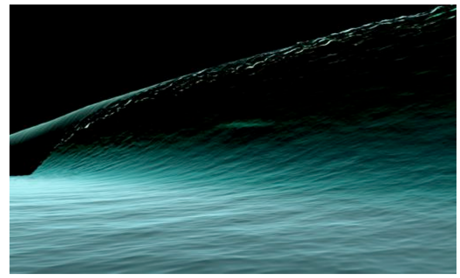
An additional finer and softer mist element was also often rendered using Splat, Imageworks' proprietary sprite renderer. These elements were used for fine wave spray or fog.

# 6.5 Compositing: Putting It All Together

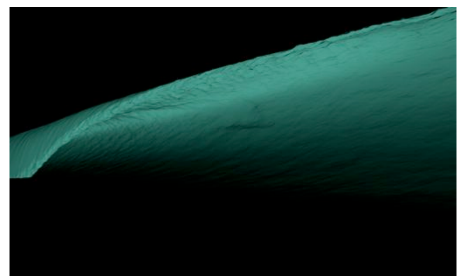
A great deal of integration of the wave elements was done during the compositing stage. RenderMan's Arbitrary Output Variables (AOVs) were used to render utility passes at the same time as the primary (or beauty) render. These utility passes typically consisted of masks and lighting components including the following:

- **Normals**: Surface point normal. This was often used in the composite to add additional specular highlights or reflections and for the refraction of the bottom map render.
- **Z-Depth:** Distance in Z away from the camera plane, used for atmosphere and depth-of-field simulation.
- **World P:** Position in world space of the surface point.
- **Fresnel:** Used to mix the ratio of reflection and the amount that we see into the water (refracted bottom map).
- **Reflection Occlusion:** Render pass used to hold out the reflection pass.
- **Top-Foam Mask:** Used to color correct the foam on top of the water surface.
- **Under-Foam Mask:** Used to color correct the foam under the water surface.
- **Bottom Map:** Render of the bottom of the ocean.

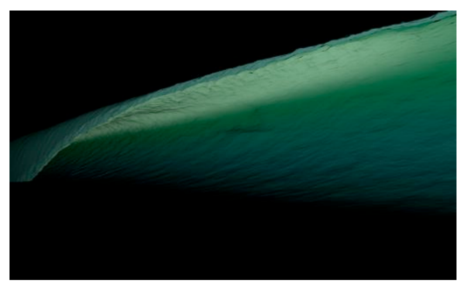
The following images demonstrate the individual layers that make up a typical wave:



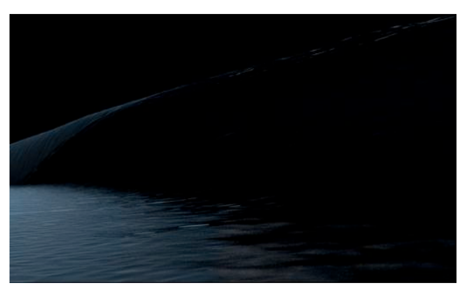
*Diffuse*



*Incandescence*

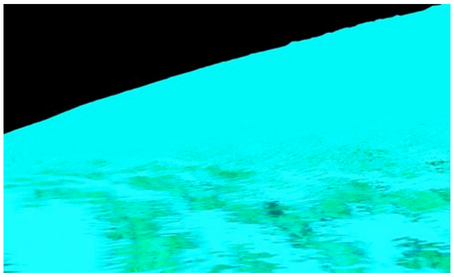


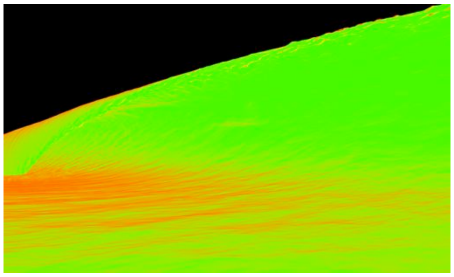
*Translucence*

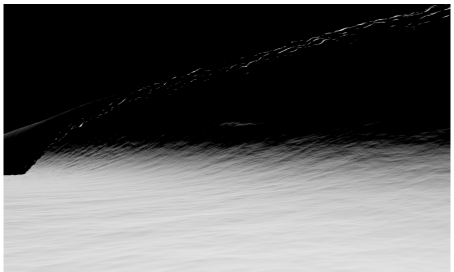


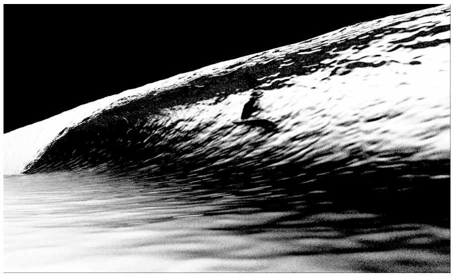
*Reflection*

*Individual layers that make up a typical wave composite (rendered as AOVs)*

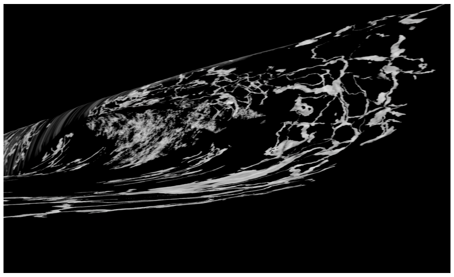

*Bottom Map*

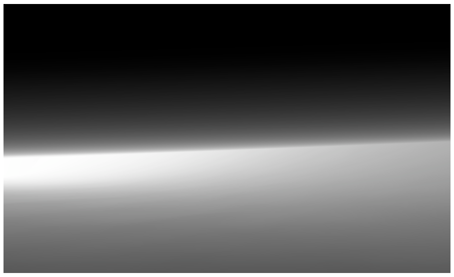

*Fresnel*


*Utility Zone*


*Reflection Occlusion*


*Under Foam*


*Top Foam*


*World Space Position*


*Final Composite*

# 6.6 Filming Waves Documentary Style

## 6.6.1 Underwater Camera Housing

The well worn adage that "invention is the child of necessity" was true for rendering the water and led to the visual concept of an underwater camera housing.

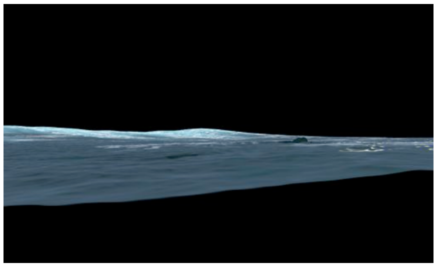
In the quest for an authentic documentary feel, there were very few restrictions placed on the camera and filming of the action. The Layout department was free to do anything with the camera in order to attain a documentary feel and to simulate how a crew might actually film in a given environment, like water. Allowing the camera to be partially submerged however caused problems when rendering the ocean and waves because the water surface intersected the camera plane. This situation resulted in long render times with very large memory requirements.

One typical solution is to push the rendering clipping plane forward and away from the camera. This rendering requirement led to the idea of simulating an underwater camera housing in which the glass of the camera housing is further away from the film plane than the camera lens. By simulating a distant plane of glass this allowed the artist rendering the water to push the camera clipping plane far enough to prevent the rendering problems. This turned out to be an aesthetically desirable choice as well. The split screen imagery created by this camera housing felt more intimate and emphasized the documentary style by the fact that the characters where being filmed with a "real" or physical camera.

The following are several of the elements that went into creating the look of the underwater camera:

- **Water Split-Screen:** Separate underwater and above water renders divided by the water line**.**
- **Underwater Look:** Various elements such as light rays, particulate material and murk (loss of light underwater) were used for the portion of the image that was underwater.
- **Lens scratches:** Including lens flaring based on light angle.
- **Lens splashes:** Water drops and sheeting on lens.
- **Underwater Housing:** Light falloff from imperfect camera housing.

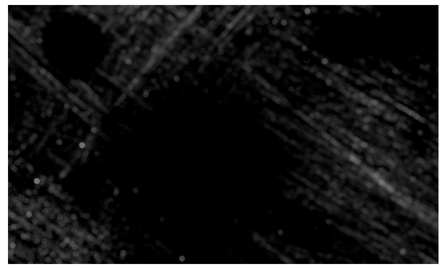
The following shot build demonstrates how these elements were composited together to create the final image:



*Above water pass*

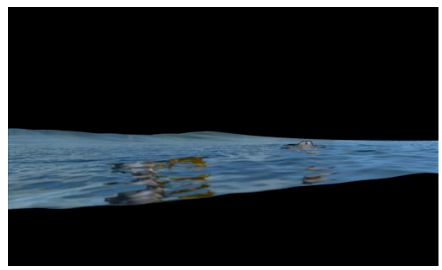
*Caustics*

*Characters*



*Lens scratches*

*Water drips on lens*



*Reflection*

*Underwater particles*

*Underwater light rays*

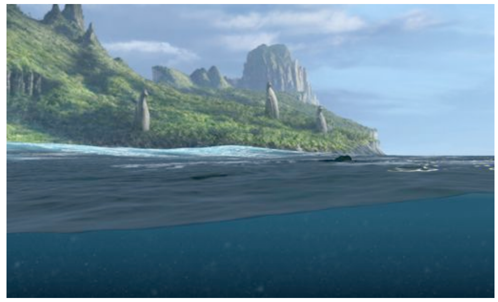
*Refraction*

*Z depth*

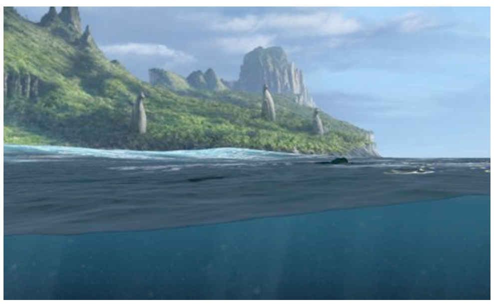
# 6.6.1 Underwater Camera Housing cont'd

*Composite of underwater camera elements*

*Water surface*



*Under water*



*Under water light rays*

*Characters*

*Characters underwater*

*Caustics*

*Depth of Field*

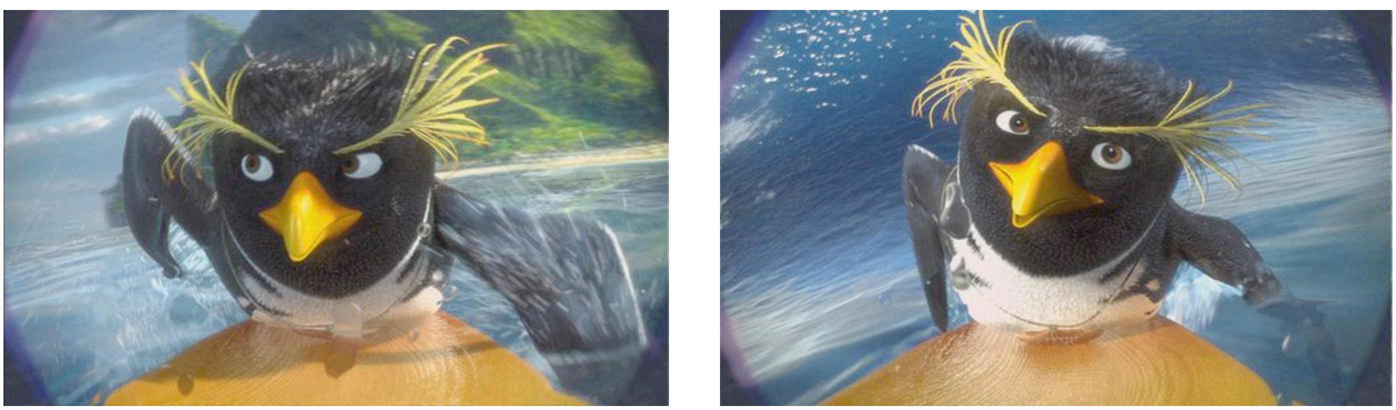
*Lens vignette*

# 6.6.2 Board Mounted Video Camera

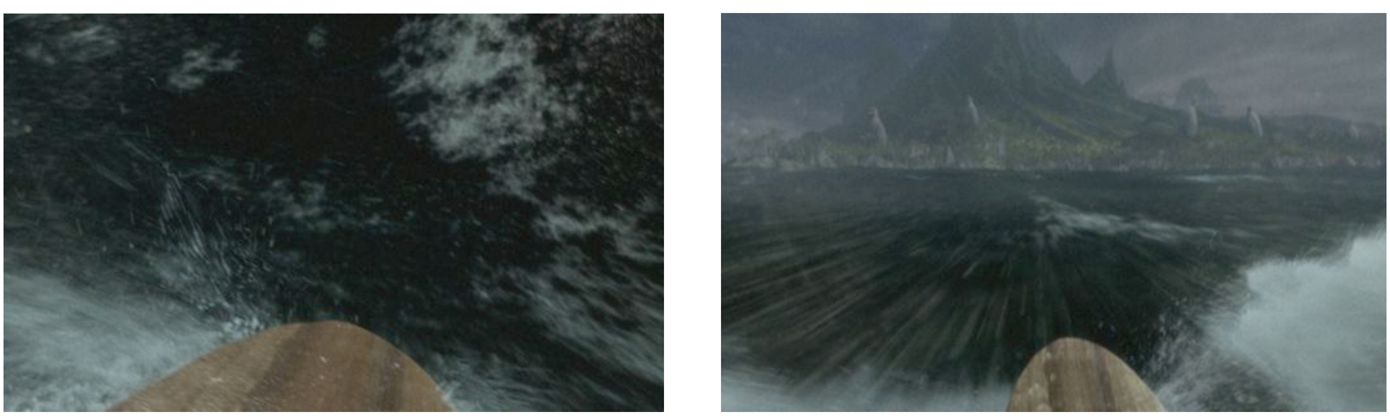
The concept of an underwater camera housing was extended to create the look of a surfboard camera, similar to that used in sports footage. In this case the idea was to visually simulate the immediacy and excitement that is characteristic of filming with a small camera mounted to the surfboard. The elements in the images on this page were composited to create the look and feel of a cheap servo operated video camera. As Dolly Parton once said: "It takes a lot of money to look this cheap."

Some of the individual elements that were characteristic of this look are listed below.

- Little motion blur (step printing for slow motion and skip printing to speed up action)
- Larger depth-of-field to simulate small, cheap lens
- Video lines
- Video dropouts
- Simulated servo controlled movement
- Video color
- Vignette from cheap camera housing with imperfect fit
- Splashes and water drops on lens



*Rear facing camera*



*Forward facing camera*

# 6.7 Examples of Waves

Variations of the techniques described in this Section were used to light and render all the waves in *Surf's Up*. The next few pages contain several of the more memorable wave moments in the film.

# 6.7 Examples of Waves cont'd

# 6.7 Examples of Waves cont'd

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

## BIOGRAPHIES

# 7.1 Presenters

## 7.1.1 Rob Bredow

Rob Bredow is a Visual Effects Supervisor at Sony Pictures Imageworks who recently completed work on *Surf's Up*. He is currently supervising the upcoming animated movie *Cloudy with a Chance of Meatballs*. While at Sony, Rob has been involved in creating many of the complex visuals featured in *The Polar Express*, *Cast Away* and both *Stuart Little* films. Rob's other credits include the feature films *Megiddo*, *Independence Day*, *Godzilla* and others. Rob is experienced in the field of effects animation, shading and rendering and has presented his work in SIGGRAPH Courses in 2000, 2002, 2005.

## 7.1.2 David Schaub

David Schaub is an Animation Director at Sony Pictures Imageworks who recently completed work on *Surf's Up*. He was also the Animation Director at Imageworks for last year's *The Chronicles of Narnia*. Schaub was the Animation Supervisor on *The Polar Express*, receiving a VES nomination for "Steamer" (one of the traditionally animated characters) and received a VES award in 2002 for his animation work on *Stuart Little 2*. Other film credits include *Stuart Little (1)*, *Cast Away*, *Evolution*, *Patch Adams*, *Hollow Man*, *Godzilla* and *The Craft*. Schaub has also acquired producer and director credits on several award-winning independent video projects.

## 7.1.3 Daniel Kramer

Daniel Kramer is a CG Supervisor at Sony Pictures Imageworks. Prior to *Surf's Up* Daniel was Effects Animation Supervisor for *The Polar Express* and presented his work at SIGGRAPH in 2005. Daniel first joined Imageworks in 2000 to work on *Spider-Man®* where he acted as Lead Effects Animator. Other projects include visuals for feature films and television including *Godzilla*, *Star Trek: First Contact* and *Independence Day*.

### 7.1.4 Danny Dimian

Danny Dimian is a CG Supervisor at Sony Pictures Imageworks and has recently completed work on Imageworks' second animated feature *Surf's Up*. Danny started at Imageworks as a shader writer on the Academy Award® nominated *Hollow Man*. Other Imageworks projects include *Spider-Man®* , *Stuart Little 2*, *The Polar Express* and *Monster House*.

### 7.1.5 Matt Hausman

Matt Hausman is a CG Supervisor at Sony Pictures Imageworks where he recently completed work on *Surf's Up* as Effects Animation Supervisor. His previous credits at Imageworks include *Stuart Little*, *Spider-Man®* , *Cast Away*, *What Lies Beneath* and *The Polar Express*.

## 7.2 Additional Contributor

### 7.2.1 R. Stirling Duguid

CG Supervisor Stirling Duguid joined Sony Pictures Imageworks in 2000 as an FX Technical Director on Robert Zemeckis' *Cast Away* and *What Lies Beneath*. He recently completed work on *Surf's Up*, and prior to that on the Academy Award® nominated film *The Chronicles of Narnia: The Lion, the Witch and the Wardrobe*. Some of Stirling's other film credits include *Dr. Doolittle*, *Stuart Little 2*, *The Polar Express*, and *The Matrix: Reloaded*.

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

# 8.1 ACKNOWLEDGEMENTS

A special thanks to all those who helped in the writing, rendering, compiling and correcting of these Course Notes (in no particular order):

Deborah Carlson
Michael Muir
Nicola Lavender
Bert Van Brande
Mike Ford
Sande Scoredos
Lu Kondor
Susie Oh
Jordan Soles
Lauren Matheson
all the SUR PSTs
Noé Padilla

*and all those involved in creating the images here in...*

# 8.2 IMAGEWORKS CREDIT ROLL

**Co-Producer** Lydia Bottegoni

**Visual Effects Supervisor** Rob Bredow

**Senior Animation Supervisor** David Schaub

**Digital Producer** Chris Juen

**Supervising Animators** Renato Dos Anjos, Chris Hurtt, Peter Nash, Chad Stewart

**CG Supervisors** Danny Dimian, R. Stirling Duguid, Daniel Kramer, Bert Van Brande

**Executive Producer, Imageworks** Jenny Fulle

**Digital Production Managers**
David Kalbeitzer
Garrick McLaughlin

**Pipeline Supervisor** Henry Vera

## MODELING

**Modeling Supervisor** Marvin Kim

**Lead Character Modeler** Moon Jun Kang

**Character Modelers**
John Butiu
Luis Labrador

**Lead Environmental Modeler** Greg Galliani

**Environmental Modelers**
Justin Diamond
Richard Lee
Justin Marshall
Gastón Ugarte

## CHARACTER SETUP

**Character Setup Supervisor** Michael Ford

**Wave Setup Lead** Erick Miller

**Character Setup Leads**
Stephen Candell
Keridan Elliott

**Character Setup TDs**
Andrew Anderson
Kyudon Choi
Ramiro Gomez
Michael Kuehn
Peter Samuel Tieryas

## LAYOUT

**Layout APM** Lauren Prince

**Lead Rough Layout Artist** Lisa Suzuki

**Wave Animation Lead** John Clark

**Rough Layout Artists**
Stephen Childers
Allen Foster
Harald Kraut
Hiroshi Mori
Dani Morrow
Carlos Pedroza
Christopher Poplin

**Lead Layout Camera Operator** Michael Condro

**Final Layout Artists**
Juan Gonzalez
Andres Martinez
Joseph Thomas
Pete Upson

**Final Layout Coordinator** Michael Killoren

## ANIMATION

**Animation APM** Patrick G. Ramos

**Animators**
Anders J. L. Beer
Josh Beveridge
Jamaal Bradley
Benjamin I. Cinelli
Jeffrey Croke
James Crossley

# 8.2 IMAGEWORKS CREDIT ROLL CONT'D

| | |
|---|---|
| Robert Fox | Kevin Freeman |
| Derek Friesenborg | Rick Glenn |
| Steven Pierre Gordon | Bill Haller |
| Dave Hardin | Jordan Harris |
| Alan Hawkins | Nicole Herr |
| Seth Hippen | Ethan Hurd |
| Michael Kimmel | Eric Lees Jeff Lin |
| Joe Mandia | Gavin Moran |
| Matthew J. Munn | Patrick Osborne |
| Keith Paciello | Brett Paton |
| Jayson Price | Andreas Procopiou |
| Tim Ranck | Gwendelyn Robson |
| Sandra Ryan-Moran | Denis Samoilov |
| Henry Sato, Jr. | Jeff Schu |
| Brian Scott | Sue Hyunsook Shim |
| Keith A. Sintay | Ron Smith |
| David Stodolny | Jeff Vacanti |
| Pepe Valencia | Roger Vizard |
| Michael C. Walling | Christopher Walsh |
| Larry White | Alex Whitney |
| Dougg Williams | John Wong |

**Animation Coordinator** Jack Finlin

## CHARACTER PIPELINE

**Character Pipeline Supervisor** Sumit Das

**Animation Support Lead** Brian Coffee

**Animation Support**

| | |
|---|---|
| Stefaan Contreras | Jason Doss |
| Chris Gallagher | Dae-Ho Han |
| Bryan Lentenbrink | Raj Naiksatam |

**Character Pipeline**

| | |
|---|---|
| Frank Müller | Terrence Robertson-Fall |
| Carolyn Oros | |

**Animation Support Coordinator** Amy R. Gordon

## LIGHTING & COMPOSITING

**Lighting APMs**

| | |
|---|---|
| Lea Lambert | Evangeline Monroy |
| Miguel A. Oaxaca | Paula Tudorof |

**Lead Lighting Artists**

| | |
|---|---|
| Karl Herbst | Michael Muir |
| Michael Sandrik | Douglas James Smith |

**Compositing Supervisor** JD Cowles

**Lighting & Compositing**

| | |
|---|---|
| Jesse Andrewartha | Leslie Baker |
| Joachim Bärnreuther | Lynn Basas |
| Jean-Paul Beaulieu | Jorge Bobadilla |
| Scott Bogoniewski | Erik J. Borzi |
| Grady Campbell | Georgia Cano |
| Onny Carr | Alan Chen |
| Amy Christensen | David Conlon |
| Bertrand Cordier | Robert W. Crain |
| Frederick Durand | Patrick Finley |
| Darren Fisher | Jason Galeon |
| Shane Glading | David Godwin |
| Svetla Gressak | Kevin T. Hahn |
| Todd T. Hara | Dave R. Howe |
| Ian Jenkins | Kurt Judson |
| Miku Kayama | Kenny Lam |
| Michael Lasker | Dan Lavender |
| Nicola Lavender | John J. Lee |
| Daniel Leung | Kimberley Liptrap |
| Tim Llewellyn | Christine J. Lo |
| Shannon McGee | Enrique Muñoz |
| Erik Ober | Winston Quitasol |
| Matthew Rampias | Priyes Shah |
| Kenji Sweeney | Eric Tablada |
| Daniele Tosti | Mike Webster |
| Craig Welsh | Mark Wendell |
| Douglas Yoshida | Melva Young |

**Motion Graphics** Nori Kaneko

**Additional CG Supervision** Walter F. Hyneman

## EFFECTS ANIMATION

**Effects Animation Supervisors**
Matt Hausman
Carl Hooper

**Effects Animation Leads**
Steve Blakey
Deborah Carlson
Matt Cordner
Tom Kluyskens
Stephen Marshall
David R. Davies

**Effects Animators**
Steve Avoujageli
Todd Boyce
Dipankar Goswamy
Victor Grant
John Han
Andrew Hofman
Jeffrey Kember
Seunghyuk Kim
Zsolt Krajcsik
E. J. Lee
Carlos Donis Lemus
Fredrik Limsater
Franklin Londin
Gustav Melich
Daniel Patrick Naulin
Siegfried Ostertag
Tom Pushpathadam
Gregory Wade Reynolds
Jeremy Squires
Mark Story
Chris Wachter
Jeff Willette
Aaron Wilson
Matthew Kiyoshi Wong

**Effects Animation Coordinators**
Keiko Koyama
Beth Tyszkiewicz

## HAIR & FEATHER

**Hair & Feather Supervisor** Rodrigo Ibanez

**Hair & Feather TDs**
Chris Bolwyn
Brian Casper
Oscar Castillo
Joong Choi
Saba Roufchaie
Sean White

## MATTE PAINTING

**Matte Paint Supervisor** David R. Bleich

**Matte Painters**
John Bevelheimer
Joseph DiCesare
Allen Gonzales
Nick Hiatt
Steve Matson
Noelle Triaureau

## LOOK DEVELOPMENT

**Look Development Leads**
Onny Carr
Karl Herbst
Michael Muir
Michael Sandrik
Douglas James Smith
Kenji Sweeney
Daniele Tosti

**Look Development TDs**
Jean-Paul Beaulieu
Alan Chen
Kevin T. Hahn
Piotrek Krawczyk
Erik Ober

**Shader Lead** Laurent Charbonnel

**Shaders**
John B. Anderson
Ryan S. Heniser
Aaron Lawn

**Texture Paint Lead** Nori Kaneko

**Texture Painters**
Suzy Brown
Matt Hollingsworth
Kerry Nordquist
Brian F. Ripley
Mike Yazijian

# 8.2 Imageworks Credit Roll Cont'd

## PRODUCTION

**Digital Resource APM** Jordan Soles

**Production Office Coordinators**
Lia Abbate
Gredel Berrios-Calladine

**Production Accountant** Marc Ostroff

**Assistant Accountant** Nicole Koop

**Assistant to Directors** Jason Cooper

**Production Assistant** Miguel Gardo

**Production Infrastructure**
Suzanne Labrie
Regaye Fulcher

## RESOURCE MANAGEMENT

**Lead PST** Lauren Matheson

**PST Night Supervisor** Troy Moore

**Production Service Technicians**
Jeffrey Benjamin
Sean Corzo
Noel Vaughan Eaton
Jonathan F. Garcia
Jason Madsen
Jonathan S. Swartz

**Development Specialist** Jennifer Cranfill

**Manager** Andrea Lackey Pace

**Digital Asset APM** Jeffrey Pohl

## SOFTWARE

**Sr. Pipeline Producer** Brian Keeney

**Sr. Software / R&D Supervisor** Bill Villarreal

**Software Supervisor** Leslie Picardo

**Pipeline Producer** Tok Braun

**Pipeline Supervisors**
Rene Limberger
John McLaughlin
Sam Richards

**Lead Pipeline Developer** Steve Lavietes

**Technical Production Managers**
Mitch Dobrowner
Brian Hamblin
Erik Strauss

**Pipeline TDs**
Ron Bublitz
Amy Hronek

**Team Leads**
Armin Bruderlin
Moti Cohen
Brian Hall
Bruce Navsky
Michael Frederick Wilson
Blair Zajac

**Engineering**
Michael Ball
Christopher Burdorf
Matthew Chambers
François Chardavoine
Clint Chua
Pablo Collins
John Flynn
Albert Golembiowski
Jeff Hill
Junko Igarashi
Manson Jones
Cottalango Loorthu
Ken McCloskey
Lucas A. Miller
Sosh Mirsepassi
Hiroyuki Miyoshi
Kishore Mulchandani
Dhasarath Ravindranath
J Robert Ray
James Salter
Jeremy Selan
Mathew Selby
Peter Shinners

Joseph Slomka
Andrea Solis
Bond-Jay Ting
Dan Weston
Geo Snelling
Nicole Tester
Victor Wang
Brian Wong

**Coordinator** Leslie MacIver

## TRAINING & ARTIST DEVELOPMENT

Sande Scoredos
Adriana Jaroszewicz
J.C. Cornwell
Samantha N. Brown

## DIGITAL PRODUCTION MANAGEMENT

Ken Maruyama
Bill Anastas
Jill Shane Butler
Camille Eden
Rachael Phillips
Dana Pettit-Vanhove
Diane St. Clair
Sharon Berlin
Jana Day
Wendy Mashburn
Karen Sickles

## ANIMATION PRODUCTION MANAGEMENT

Lauren Ann Littleton
Anett Gough
Chad Hellmuth
Robin Alan Linn
Mary Rachel Thompson
Charlene Maryland

## SYSTEMS ENGINEERING

**Supervisor** Alberto Velez

Ted Alexandre
Garrett Cox
Tyler Franks
Alex Lostaunau
Jeremy Simpson
Daniel Villarreal
Scott Williams
Charlie Boster
Gerardo De La Cruz
Olin Kimberly
David Perry
Michael Trujillo
Ken Williams
Peter Wright

## SYSTEMS R&D

Dean E Miya
Suri Denduluri
Stephen Kowalski
David Simas
Nicholas Bali
Bruce Dobrin
Daniel Lake

## SONY PICTURES IMAGEWORKS SENIOR MANAGEMENT

Tim Sarnoff
Tom Hershey
Don Levy
Barry Weiss
Debbie Denise
George H. Joblove
Stan Szymanski

Course # 13, SIGGRAPH 2007
San Diego, CA, USA

# SURF'S UP

**Sony Pictures Imageworks, Culver City, CA, USA**

*Surf's Up* images courtesy of Sony Pictures Imageworks and Sony Pictures Animation



*Body: Adobe Garamond™; Headlines; Confidential™;*
*Layed out and designed by Noé Padilla in Quark™ 6.5 on Apple Macintosh G5*