|  |
| --- |
| Freddie Jones – fj16315 |
| Primitives Report |
| Animation Production – COMSM0013 |

|  |
| --- |
| Freddie Jones  [Date] |

# Pre-production and Planning

The specification for the animation was for primitives to come together with realistic motions to create a robot. Starting out I had a few ideas, ranging from snow being blown around to make a snowman to a robot factory. The idea I stuck with though was a small Lego robot building a bigger robot out of Lego. I chose this idea for several reasons; Lego lends itself to working with primitives, I have a Lego Mindstorms kit at home making it easy to pre-vis, and the addition of a second character really brings life to the animation.

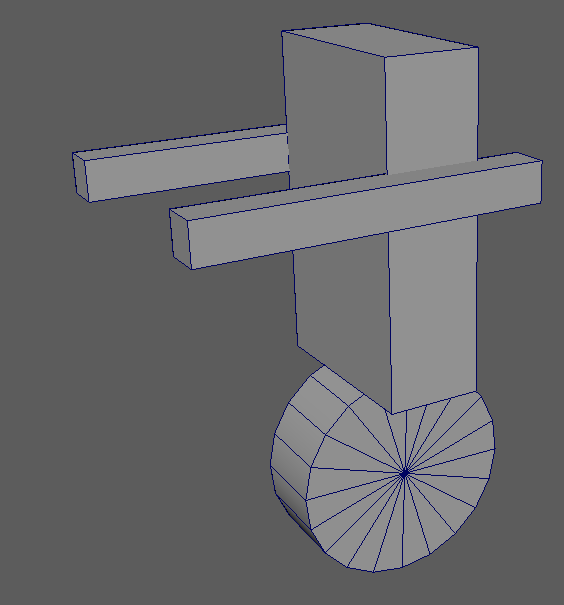
With a concept in place, I started designing the characters. When designing them, I had to keep in mind the big robot and his capability for dancing. As the sketch shows, the little robot would be largely taken up by his control block, with little arms and legs sprouting from it. The big robot however has much more pronounced limbs, with claws for hands and caterpillar tracks for feet.

I then decided to move on to directing an animatic short using stop motion. The reason I used stop motion as opposed to drawn storyboards was predominantly for ease and flexibility. I had a model to hand and a camera. With these I could plan out my shots with more instant visual feedback. If I wanted to change a shot, I simply had to take another picture. Additionally, I’m not much of an artist, so pictures captured much more information than my bad drawings. Before shooting the animatic, I decided to change the little robot’s legs to a single wheel. This offered more character with less modelling and rigging.

With the animatics, I was able to lay out my shots. It gave me an idea for where the camera was going to be and how it would travel. In hindsight, when working with a physical camera, it’s much easier to simply take videos instead for a pre-vis. With how many pictures I was taking, it was difficult in some shots to make out what the camera motion was. However, the animatics also helped me plan how long each shot would be, as well as their pacing.

At the end of planning, I had flushed out my concept. The little robot comes into frame roaming around. He discovers a control block. After a shock, he goes up to it, then looks around. He finds robot pieces near the block. He bends down towards the robot’s head and resolves to put it together. He starts with the legs, dragging them over to the block and plugging them in. As he starts to walk away the tracks turn on and the block zooms away. The little robot frantically gives chase. Cut to him attaching the arms. First the left, then the right. As he goes back over to the left arm, the claws activate and grab his wheel, scaring the little robot. Cut to an overhead shot of the little robot pushing the head piece into place. Then back to a wide shot, he attempts to pull the robot up with some wire. The big robot becomes upright and the little robot celebrates, until the big one starts teetering. Panicking, the little robot pulls the robot back the other way to centre him. Last shot is a zoom in from the little robot to the big robot’s face.

# Challenges and Highlights

After planning out my 20 second sequence, it was time to animate in Maya, starting with blocking out the scene. Blocking out the sequence let me see the pace more clearly, as well as the camera placements. Then I could play around and see what fit. As this was just a rough cut, very little time had to go into modelling. Everything in the scene was built out of transformed cubes and cylinders. However, even with low poly models, I was still faced with several challenges.

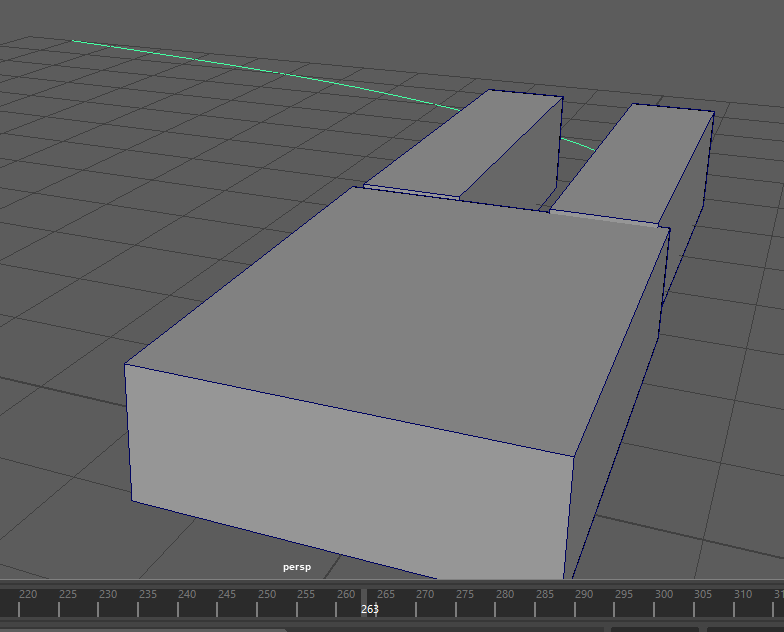
Little robot model made of primitives

The first was making sure I rigged the models correctly. As I had a pre-built character in the scene, I had to make a simple character control rig. I organised the little robot’s geometry into a hierarchy of parents and groups. For example, the arms are grouped together, which is then a child of the torso. With this, children inherit transformations from their parents. So, when the torso rotates, so do the arms, but not the wheel. I also had to move some pivot points, so the geometry rotated around the correct point. The highest parents were then parented to a locator node in order to control the overall movement of the robot from one place. However, parenting does present an issue. Children inherit all transforms from their parent, including scale. This caused weird deforms during rotations. This was a simple fix, I just had to freeze the scale transforms on all the parents.

The next challenge was getting the little robot to pick up objects in the scene. This involved the object following the robot arm’s path convincingly. Initially in tests I tried simply keyframing the object along the same rough path as the arms. This didn’t work however, as it appeared more to be floating in the air.

It then became clear that the solution was context switching. This is the process of turning objects visibility off and on between frames. For the robot to pick up an object, the frame that it was picked up, the object’s visibility should turn off. In the same frame, a duplicate of the object that is parented to the robot’s arms’ visibility should turn on. Assuming the objects occupied the exact same space during the context switch, it will appear seamless. As the duplicate object is parented to the arms, it will inherit their transform perfectly. When the robot needs to put the object down, context switch again. The only issue was making sure the objects were in the same position when the switch happens. This is simple though, as you duplicate and parent the object in the frame he picks it up. I also used context switching for building the big robot model up. All his limbs are invisible until they are connected. Then, as they are in the big robot controller’s hierarchy, the limbs can be controlled like the little robot.

Another challenge was the runaway robot shot. The big robot suddenly sets off and darts left. To do this, I used a motion path. The big robot’s controller follows a motion path, which in turn carries the torso and the legs. The speed of the motion is defined by the start and end points in frame numbers. I also added an additional point near the beginning of the path in order to show the robot accelerating. The direction of the motion is defined by the curve I created. I had to use the cv’s to adjust the curve, so there wasn’t too much wobble at the beginning, and to make sure the path looked natural. The difficult part was pacing and framing the shot. I didn’t want the animation to spend too long looking at it drive off, but also didn’t want it to zoom off then cut too quickly to the next shot. I also had to make sure the robot was still framed even after it zooms away. Once I was happy, I baked the animation into keyframes. This gave me more freedom to change values as I saw fit.



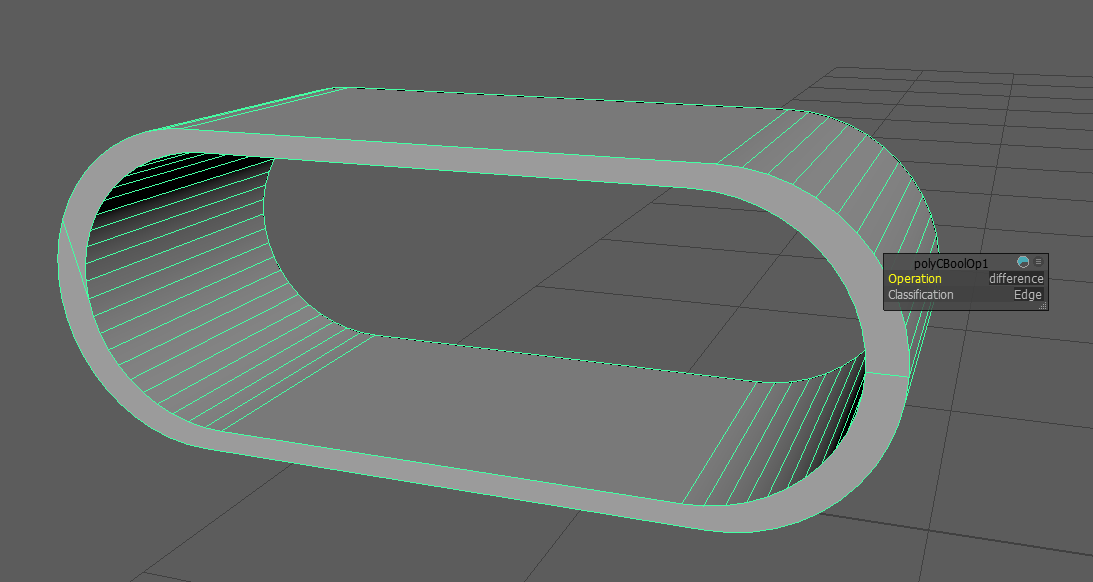
Big robot following a motion path

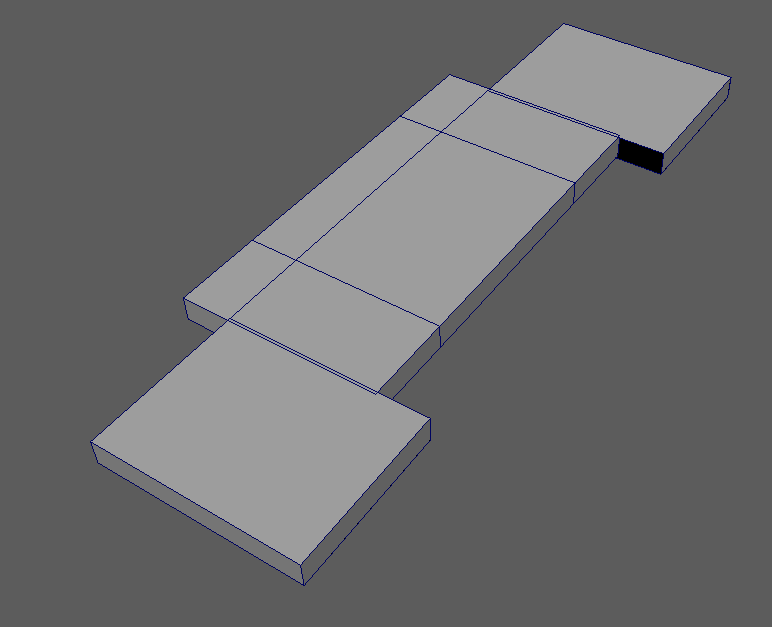
After that, the only real challenge was fitting everything I wanted into the 20 second limit. The original animatic ended up at about 25 seconds long, so it was clear I had to cut some things out. The main cut was the final few shots, with the robot being pulled up. This meant I had to end on the little robot attaching the head. I was happy with this, as you could still see the little robot celebrating in the background, coupled with the zoom in to the head gives it that “the robot is complete” feel. I could also go through the key frames in the graph editor and move them if I needed more time on a section than I originally allotted.

With the blocked-out animation complete, I felt it was important to show it to a new pair of eyes and get some feedback. Someone else may have noticed something I missed or say when something doesn’t look quite right. The feedback I got was to look at the camera placement and how it the watcher’s eye is drawn. The main problem being when the legs run off, they’re on the right side of the screen, but then it cuts to the arms being put on and the legs are on the left side of the screen. It feels like a new robot being built. I therefore decided to change the camera angle. I was also told to consider those impact points and really give them some weight. With that in mind I moved on to refining the animation.

Before I moved onto the impact points, I decided to animate the big robot’s treads. In my design, the big robot has caterpillar tracks for feet. Therefore, the treads must move along the track whenever the robot moves. These were a big challenge to get right and had to go through several iterations.

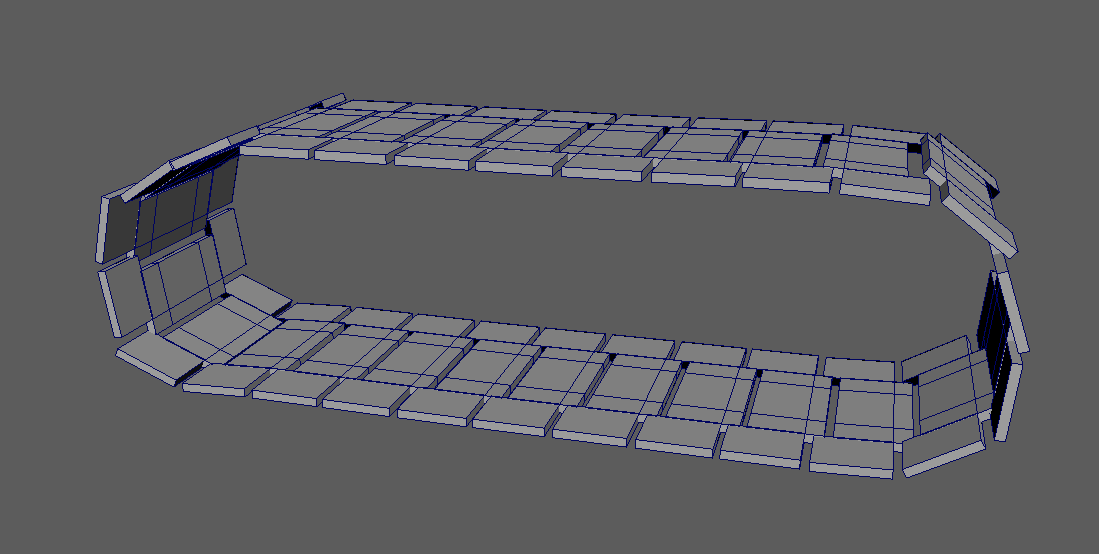
In my first attempt, I bevelled the edges of an oblong to round them off, duplicated it and shrunk it, then difference Boolean the two together. This formed a band polygon. However, with this one polygon it was impossible to have it rotate around two wheels the way tank treads do. For the second attempt, I modelled a small part of the tread. Then I made a closed curve of the path of the treads and made the polygon follow the curve as a motion path. After going into the graph editor to change the motion to a linear tangent, I created an animation snapshot at every frame. This effectively duplicated the tread polygon across the curve. It took some trial and error playing with the start and end points of the motion path, so the treads looked perfectly connected. I was very pleased with the look, but it was still not moving as treads should. This is because the snapshot captured the motion of the object at every frame. So, every frame the animation moved up, the treads would all perfectly replace each other’s positions, making it appear as though nothing had moved at all.





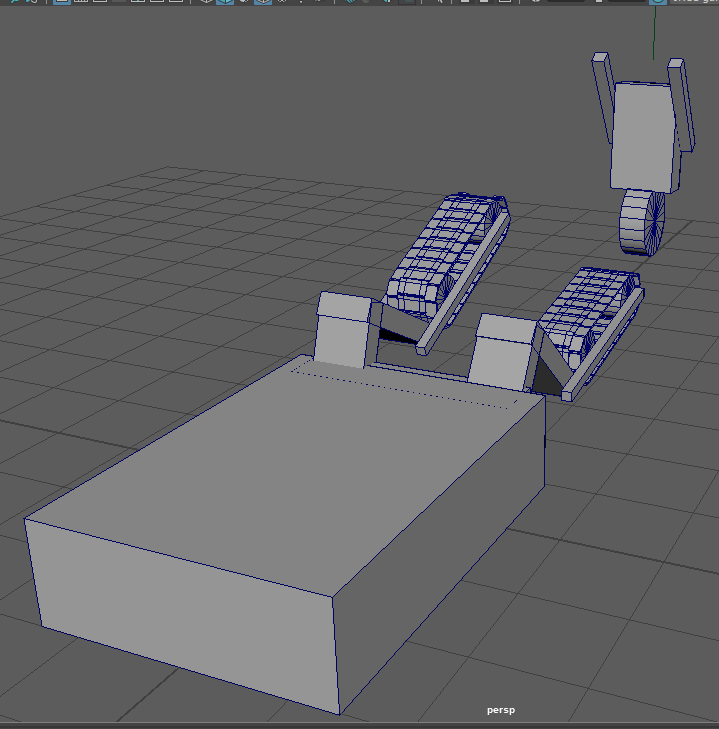
First attempt at tracks | Single tread piece

The next attempt started the same as the previous one, a small tread polygon attached to a motion path. However, this time, instead of creating an animation snapshot, I duplicated the tread and applied that to the motion path too. Although this causes both treads to start and end in the same place, I got around this with set driven key. I made a new attribute in the big robot controller to control the treads. Then I keyed the u-value of both the treads to the new attribute. So, when tread attribute is 0, u-value is 0. When tread attribute is 1, so is the u-value. However, for the second tread, I offset the u-value. This caused the second tread to lag behind the first in perfect sync. I then just had to repeat this for every tread on both feet. The final touch was in the graph editor. I changed the tangents for all the treads to linear, cycled pre and post infinity curves, and deleted the keyframes from the motion path. With this, I can now animate the treads.



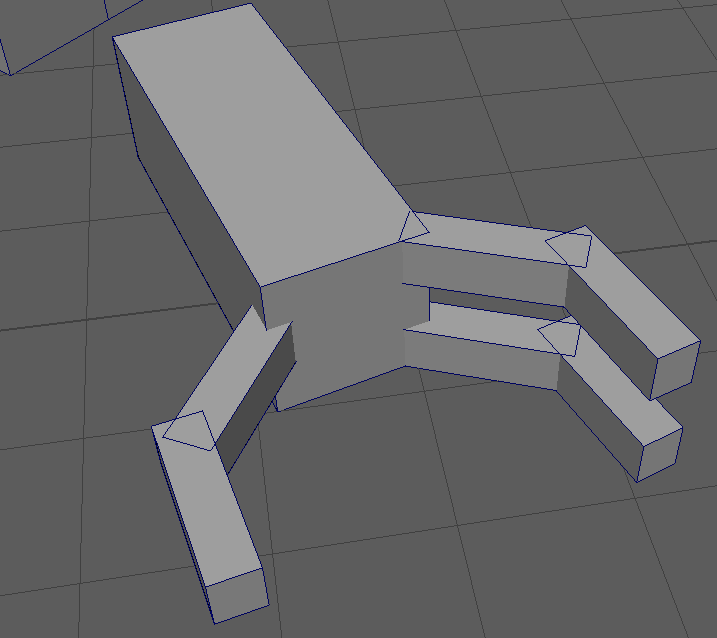
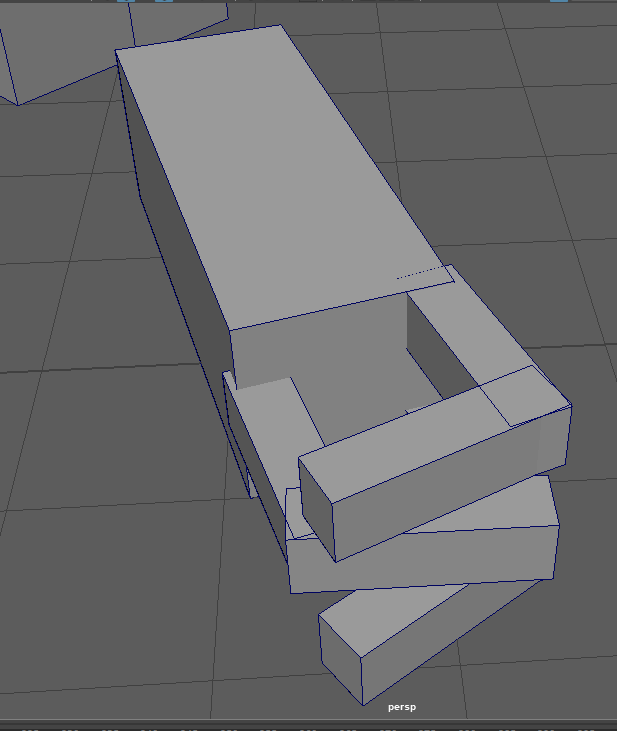
26 tread pieces put together to form the full tank tread

Using set driven key on the treads was very useful, as it let me keyframe a single attribute to control the speed of the treads as they move perfectly in sync. I used this power to increase the tread speed slowly as it accelerates, then the treads get faster as the big robot zooms further away. The only problem is it becomes hard to see the treads moving in the shot. It’s not too big a problem though, as it will become more useful in the dance assignment when we get to see the big robot move around more. On top of the tread’s animation, I added knee joints to the legs, as well as a small improvement to the impact of the legs connecting with the body. The body now recoils back, the legs flail a bit out of phase, and the treads whir around.

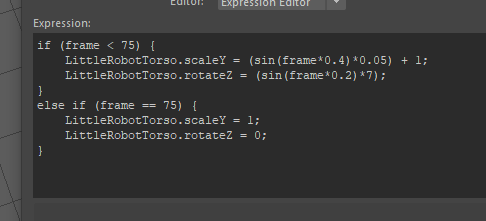


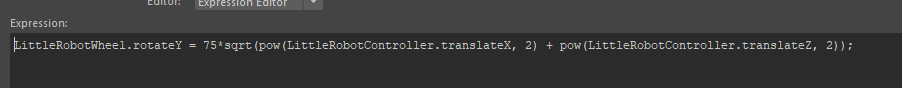
Plugging the legs in scared him!

After the legs, I wanted to move on to the next modelling improvement; the claw hand. As only one of the arms is grabbing the little robot, I only spent time improving that arm. The model was simple, two oblongs for each finger. The arm is then the parent of the fingers, which is in turn the parent of their respective finger tip oblong. In order to make it easier to control the hand, I added a new attribute in the big robot controller to control the claw opening and closing. The attribute, open, is between 0 and 1. 0 is closed and 1 is a fully open claw. To achieve this, I used set driven key again, where the new attribute drives the rotations of the fingers. With this I could easily keyframe the hand opening in preparation, then closing to grab the little robot. Without making the arm clip through the robot however, I did need to use a lot of squash and stretch. While it does come off as a funny cartoonish style hand launching for the robot’s wheel, I worry it does seem a bit out of place. I also improved the impact animation of the left arm connecting. The main body tilts back and recoils momentarily. I decided to leave the right arm’s animation for contrast and to show the audience both arms only work when they’re both connected.



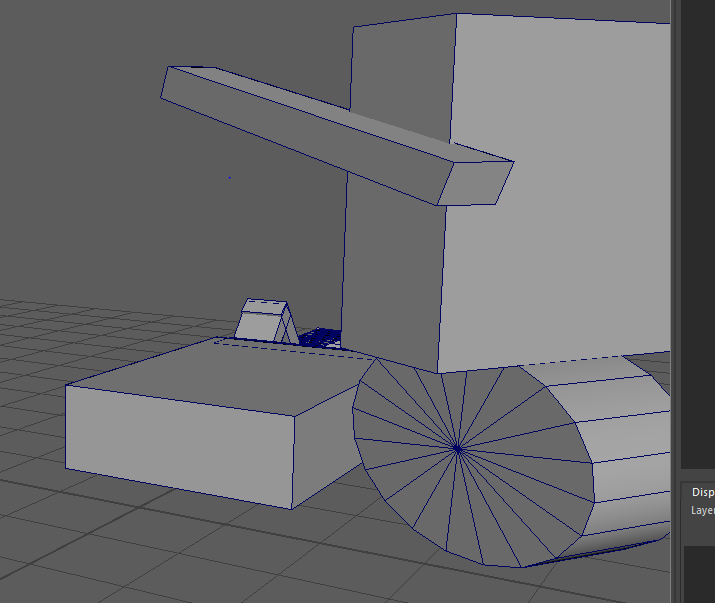
Hand open and close

After getting some more feedback, I decided to improve the little robot’s expressions. At this point the little robot’s movement was looking too robotic. To add some organic movement, I used MEL expressions. By using sin functions on the current frame number, I made a small oscillation in the scale of the torso and its rotation left and right of the wheel. I also matched the rotation of the wheel to the distance travelled by translate x and z. Altogether this causes the little robot to sway and bob a little as he travels, as well as the wheel rotating in time.



MEL expressions

I also wanted to add more expression to the robot’s faster movements. To do this I used one of the animation principles; anticipation. The robot now winds up for a big move by leaning back and moving back slightly. This works well to prepare the audience for his move. I used a similar technique to improve the little robot’s movement when carrying the legs. When he bends down to pick up the legs, his arms squash down. When pulls up to carry the leg, his arms now stretch out and his torso goes straight. This shows the audience he’s putting his back into it and that the legs have real weight. I also added a little wobble to the legs as he drops them to further show their weight. Additionally, for the body driving away, when the little robot realises the body is going away, I added a Donald duck style pop frame for more emphasis. I’m not entirely sure it works but it does add some urgency to the moment.



Donald duck style pop frame

# Conclusion

Overall, the final animation has some life to it. It’s entertaining and funny. However, given more time, there’s definitely more to improve upon.

If I could increase the length of the animation I would, as 20 seconds feels too small to introduce a meaningful story. Everything rushes by too fast without giving the audience any time to settle on any one thing. For example, with more time I could added a part in the beginning where the little robot connects two smaller bits together. This would be that lightbulb moment for both the little robot and the audience where they figure out how to put the big robot together.

However, I do love how the tracks came out. They look well modelled and the animation is smooth. Although for the time spent on them, they aren’t in shot too much, which may have been a mistake. But I believe they will be seen more in the dance animation, when we can see the big robot move around more.

If I were to redo the animation from scratch, I wouldn’t change the pre-production process. I believe I planned the shout out well. This helped me a lot when it came to blocking out the animation. I would certainly spend more time on the little robot and less time on the tracks however. As he is, the little robot isn’t expressive enough. I would use more of those animation principles to play around with him more. Meanwhile the time spent on the tracks compared to their screen time was not worth it in this assignment, despite how well they came out. I would also spend more time focusing on the impact points, make them look a bit more natural.

# Appendix – Sketches

