Postmortem: Digital Chocolate's *Tower Bloxx*

By Jeferson Valadares,Mikko Kodisoja

Introduction

Digital Chocolate is a mobile games publisher committed to delivering top-notch titles to the market, which in return ensures that here at Sumea, Dchoc's internal development studio in Helsinki (Finland), not a single boring day goes by. Aside from delivering a constant stream of solid titles, we're also proud to say that our company is spearheading the innovation efforts within the mobile games industry with several un-licensed, outside-of-the-box-thinking game concepts such as *Mobile League Sports Network (MLSN)* and *Johnny Crash Does Texas*.

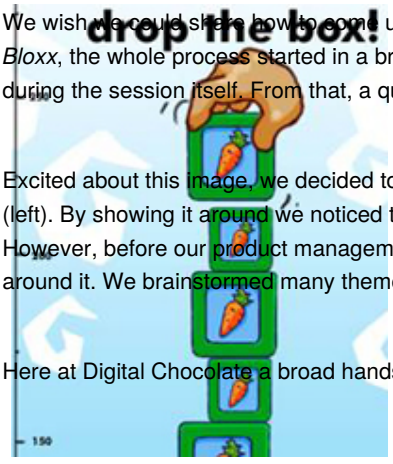
The more recent concept in this family is *Tower Bloxx*, which received a green light for production a bit over a year ago. Now looking at its market acceptance and its slightly bending industry awards shelf (from IGN to the Meffys to the Mobies), it's clear that it has become one of the crown jewels of our studio.

In case you haven't tried it yet (bad, bad you), the game in a nutshell is a mixture of timing-based action and puzzle. To quote some reviewers, it's like "*Tetris* meets *Sim City*." Your ultimate mission is to create a Megalopolis of millions of people (the Tower Toons) by building skyscrapers where they can live. The better you perform in building the towers, the more Tower Toons will be able to move into your city.

In the one-button action mode, you build a skyscraper by stacking floors on top of each other. You do this by dropping them from a crane. It would be easy except for the fact that it's windy, so the floors swing a little bit (well, quite a lot once you get really high up there), and it's crucial to get the timing right.

Tower Toons will be moving in according to how precisely the floors are dropped on top of each other and also by how fast you do it. Drop them too close to the edges of the tower and it might fall down, maybe even bringing some finished floors with it. If you drop floors incorrectly too many times, the construction of the tower will be aborted for safety reasons (even a virtual city can afford only so many engineering screw-ups). To finish a tower, you drop a roof on top of it. Now that the icing is on the cake, it's time to choose where to place the tower in your city.

This is where the puzzle element kicks in. The city mode challenges you to place the towers in the most optimal way to get the highest population count possible out of the city. There are four tower types that you can choose to build (10, 20, 30, and 40 floors) and for each of them there's a strict rule where it can be placed. 10-floor towers can be placed anywhere around the city, while the hardest 40-floor towers have to have all the other building types surrounding them in order to be placed. There are also a limited number of slots where you can place towers.



We wish we could share how to come up with nice concepts easily, but in reality there's no easy way to go about it. In the case of *Tower Bloxx*, the whole process started in a brainstorming session. The idea originally involved a train, but it evolved into something like a tower during the session itself. From that, a quick mock-up image was made by one of our designers where a hand drops a block to form a pile of blocks.

Excited about this image, we decided to develop a simple throw-away prototype with very simple graphics, which was called "drop the box" (left). By showing it around we noticed that we had to pry the phone away from people after they had started playing, which was a good sign. However, before our product management team would let us go off and build this into a game, they asked us to find a good theme to wrap around it. We brainstormed many theme ideas, and in agreement with our product managers we decided to go for the current theme, which was also the studio's favorite.

Here at Digital Chocolate a broad handset support is a must (we're not happy just having our games running at high-end phones, so we always shoot for 100% support), so this is always one goal in our projects. Aside from this, the project had three major goals:

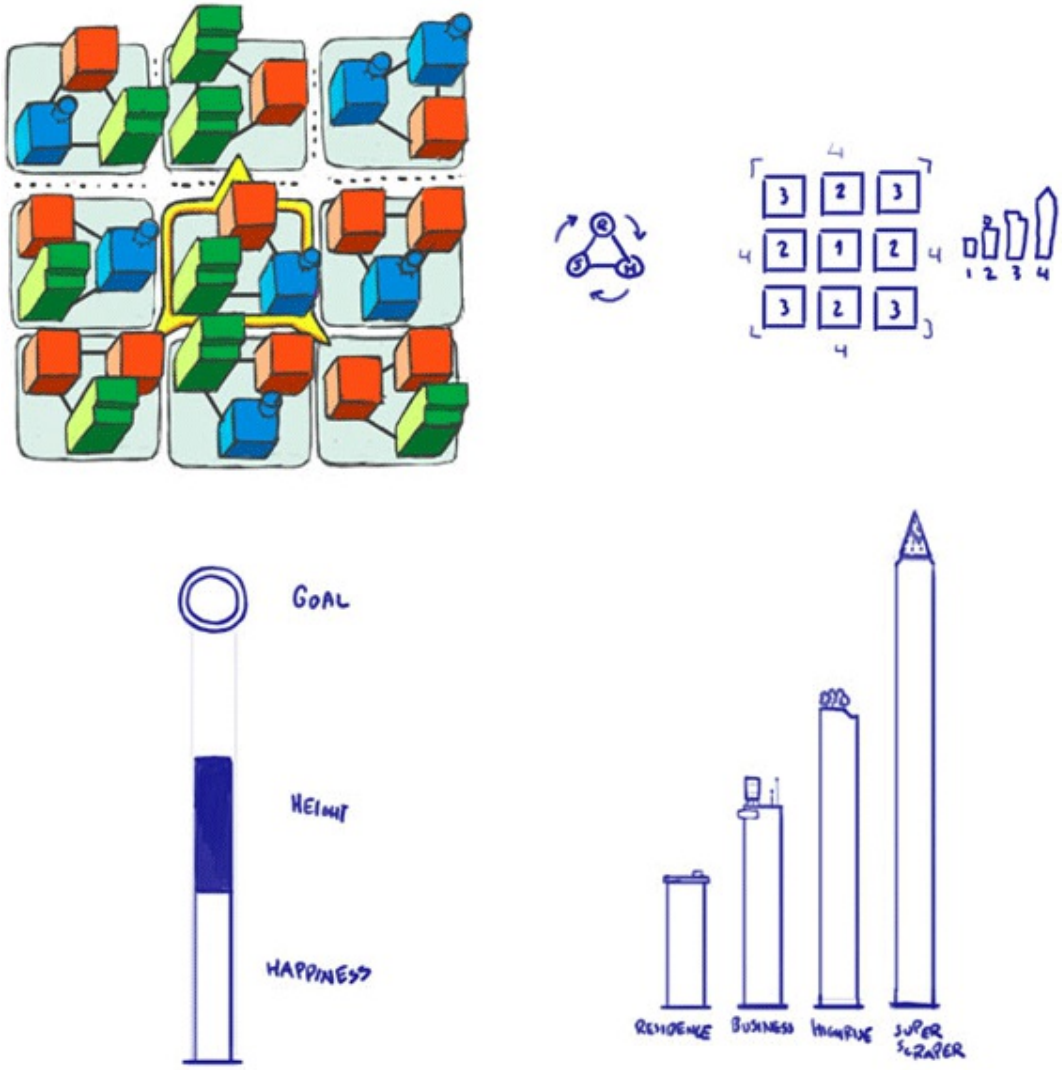
- Create an innovative game that differs from the current market offerings.
- Make the gameplay easy and fun but deep enough to keep the player playing again and again.
- Develop in parallel a 3D version reusing as much as possible from the 2D one.

With these challenges in mind, we set out to solve the four-month puzzle of building Tower Bloxx.

What Went Right

We believed in ourselves. During development, we always send the game to a couple of external reviewers, and also for evaluation inside the company. At this point, we usually get some good reviews and useful feedback that we use to improve the games. However, with *Tower Bloxx*, we got a very mixed response. But we had a great gut feeling about this, so we decided to march on nonetheless and trust our own design views.

Prototyping. As we discussed earlier, the original idea came from a mock-up image which was later turned into a prototype, which led to our core mechanics. After production started, we also needed something to fill the long-term gameplay demand, so we had several ideas of how the rules of the city mode could work, and we prototyped those using both pen-and-paper and Excel. In this way, we could test and tweak our ideas without the need for programming time, which helped us getting the game done faster.

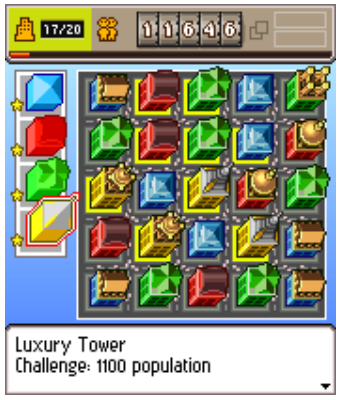


Iterations with core game mechanics. Even though we had a working prototype which was fun to play, we weren't fully satisfied with the overall gameplay, because it wasn't deep enough. After about ten minutes playing it, people started asking what else is there to do. At this point, one could think that the addtional city puzzle mode was the answer for this problem, but the fact is that if we'd have gone that way, it would only have dealt with the side effects of the underlying problem, which in this case was that the block dropping and tower hovering part should be more appealing. Our goal was to make the core game mechanics super-fun so that it alone could have kept players playing the game over and over again.

To make this happen, we put a senior designer and a senior programmer to work together on the core game physics. They were sitting next to each other and having constant communication during the whole iteration time. The total iteration of the core mechanics took three weeks, with one iteration cycle taking something from half an hour to three hours. When a new version was built, the designer would play it and give instant feedback on the physics: what was not right and how the next iteration should feel. If the game mechanics needed any additional graphics or change to the existing ones to make it more fun, this was always put on the fast-track, getting in front of all the other tasks the art team would have had on its table.

The lead designer also took part in testing the core mechanics a couple of times a week. This is always important, since many times people working very closely to the things they love can get blinded to errors, or end up trying to tune and tweak something which will never work.

This working method allowed us to get the core game mechanics to the point where we wanted it to be, and enabled us to more easily drop some features from the core which were loved by team members but that in the end wouldn't have worked so nicely together. However effective, this approach requires some extra attention when planning the project; we will discuss more of this in the "What went wrong" section.



Gameplay value. Even though the action mode at its core is pretty simple, we managed to make it fun and more varied by adding some details, like stronger winds as you go up, roofs and some visual easter eggs. This, coupled with the long-term challenge of the city mode, gave the game a great span of gameplay. Building a tower is satisfying and challenging on its own, which is good for a quick session on the subway, but the city mode adds an extra layer of complexity that players can solve and experience over longer periods of time.

Porting the fun to the low-end phones. Usually it's a huge challenge to make a game that is fun both in high- and low-end phones. Tiny screens and slow performance usually force developers to compromise on the gameplay experience, especially in action games. In Tower Bloxx, we took lots of care with the physics and the collision detection of the action mode to ensure that it would run well in limited environments, and we also made many graphical elements using code (such as the city background), which not only saves precious space but also makes it very easy to scale to different screen sizes. We also planned the puzzle element with this in mind as well. As a result, it was an easy game to port, and the game runs as well on a low-end phone as it does on a top-of-the-line one.

What Went Wrong

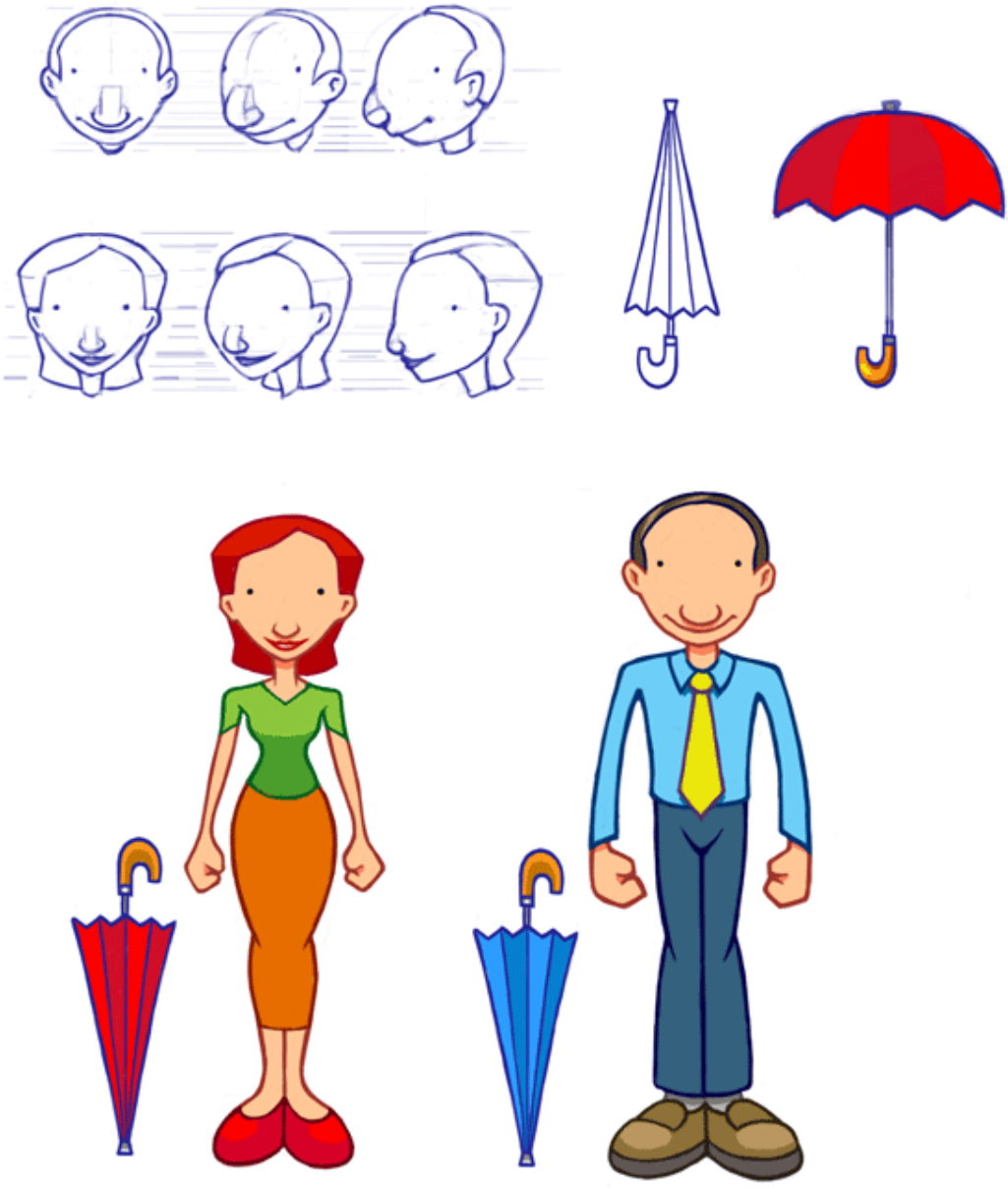
Prototyping graphical UI elements. Even though we did prototype the core game play and the rules of the city mode, we didn't pay too much attention on how the actions like placing the building into the city would work graphically. Our artists made many of animation frames and sprites but we didn't test them before (for example, using Flash) nor did we do any GIF-animations to see how they would work together on top of the cityscape.

This cost us too much time as our city mode programmer had to implement the graphics and iterate the code multiple times before the rest of the team was happy with the results, and all this time was taken away from implementing more features into the city mode. In the end we had to drop many nice-to-have features from the city mode and from the game flow due to this.

Scheduling. The project was scheduled as a project where you have a good idea of what you were doing. This was not the first time Sumea worked on one-button game mechanics: two years ago we developed a game called *Johnny Crash* and its sequel, *Johnny Crash Does Texas*. Before the production of Tower Bloxx we carefully went through our internal postmortems of those projects trying to internalize all the learnings from them.

This worked well, and we managed to dodge the major problems of both *Johnny Crashes.* But now, looking back, we can say that *Johnny Crash* was not the best case to base your planning from a producer's perspective, especially because in that case we got the tuning right fairly quickly.

Tower Bloxx was an innovative title and still we treated and scheduled it like it was a sequel or a game where you can have at least a good guess on how much each task will take. For example, the iteration cycles of tuning the tower physics were not initially forecasted to be as long as they ended up being. Another example is that initially we planned the city mode to be just a graphical high score table, so our estimates were based on that. When we decided to change it, it put the schedule in risk, though in the end we did manage to keep the schedule.



Producer was wearing two hats. In our projects we have a lead designer who is responsible for the design of the game and a producer who takes care of the whole project. In this particular project, the same person was the Producer and the Lead designer, which created some conflicts within the project. As a designer, you want to tweak the game to perfection, and as a producer you want to do it as on time/budget as possible.

This is a healthy tension in any game, but the lines are a bit blurred when you have the same person making both calls. This can also be very frustrating to the team, who might come up with a new design solution and they can't know which hat the producer/lead designer was wearing when deciding to drop a feature.

3D version could be better. Although we did get a 3D version of the game out at the same time and in a cost-effective manner as we had planned, we only managed to do it by making it as similar as possible with the 2D one. We had a few other cool features planned, but implementing those would require time we ended up not having.

Naming. One of the hardest things in the mobile landscape when you are working on unlicensed titles is the name of the title. Our industry has the smallest advertisement window in the world, meaning that in most cases the user will buy the game while browsing a WAP page and seeing only a list of names, without any screenshots or extra information.

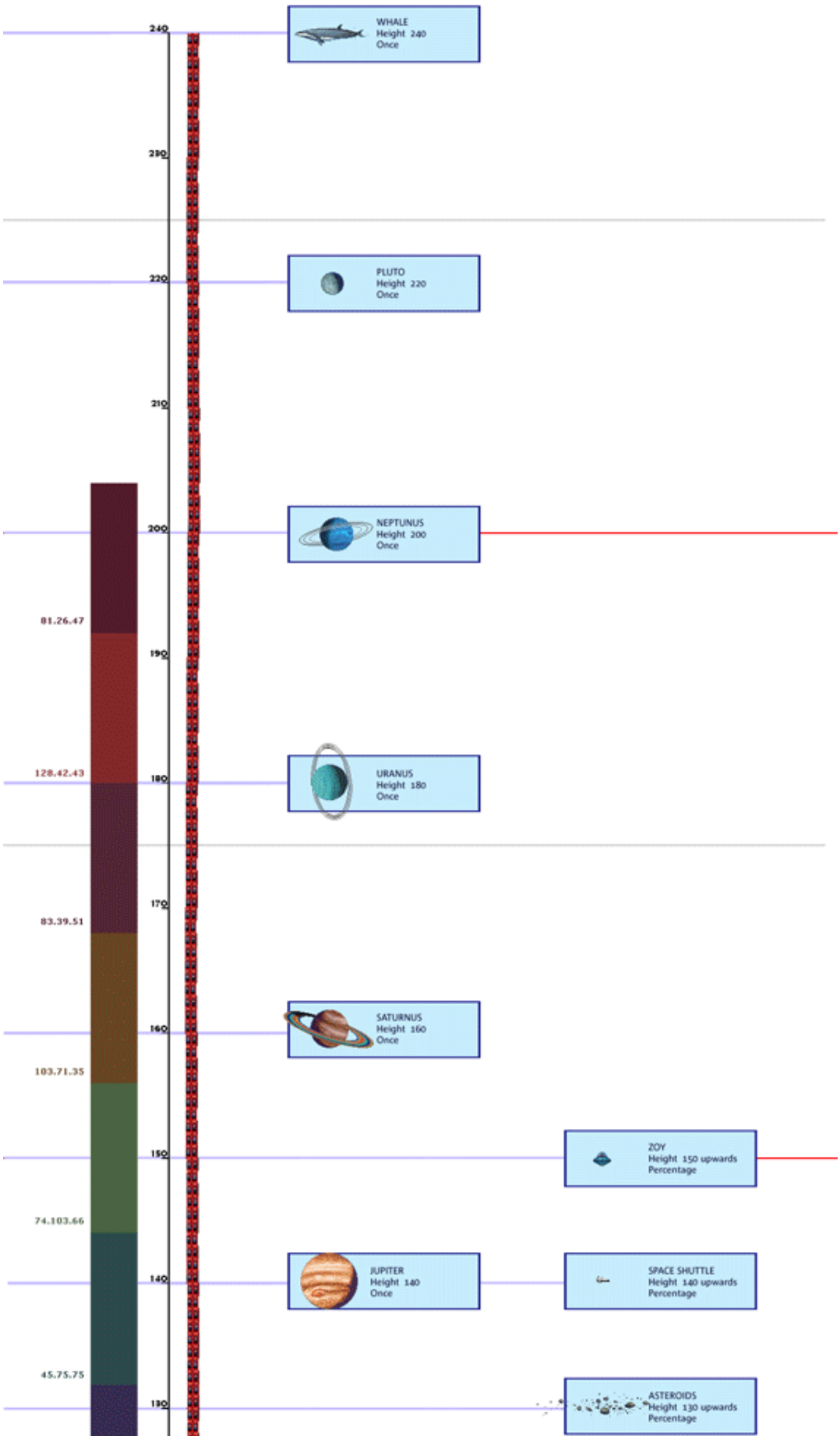
So in many cases the end-user has to make a decision based only on the name of the game. As a result, the name has to be descriptive and appealing on its own and also when compared against games with licenses. This problem gets even worse when you're trying to come up with something new – how do you describe a game where you pile floors up to build towers that you want to place in a city so you can attract people to it?

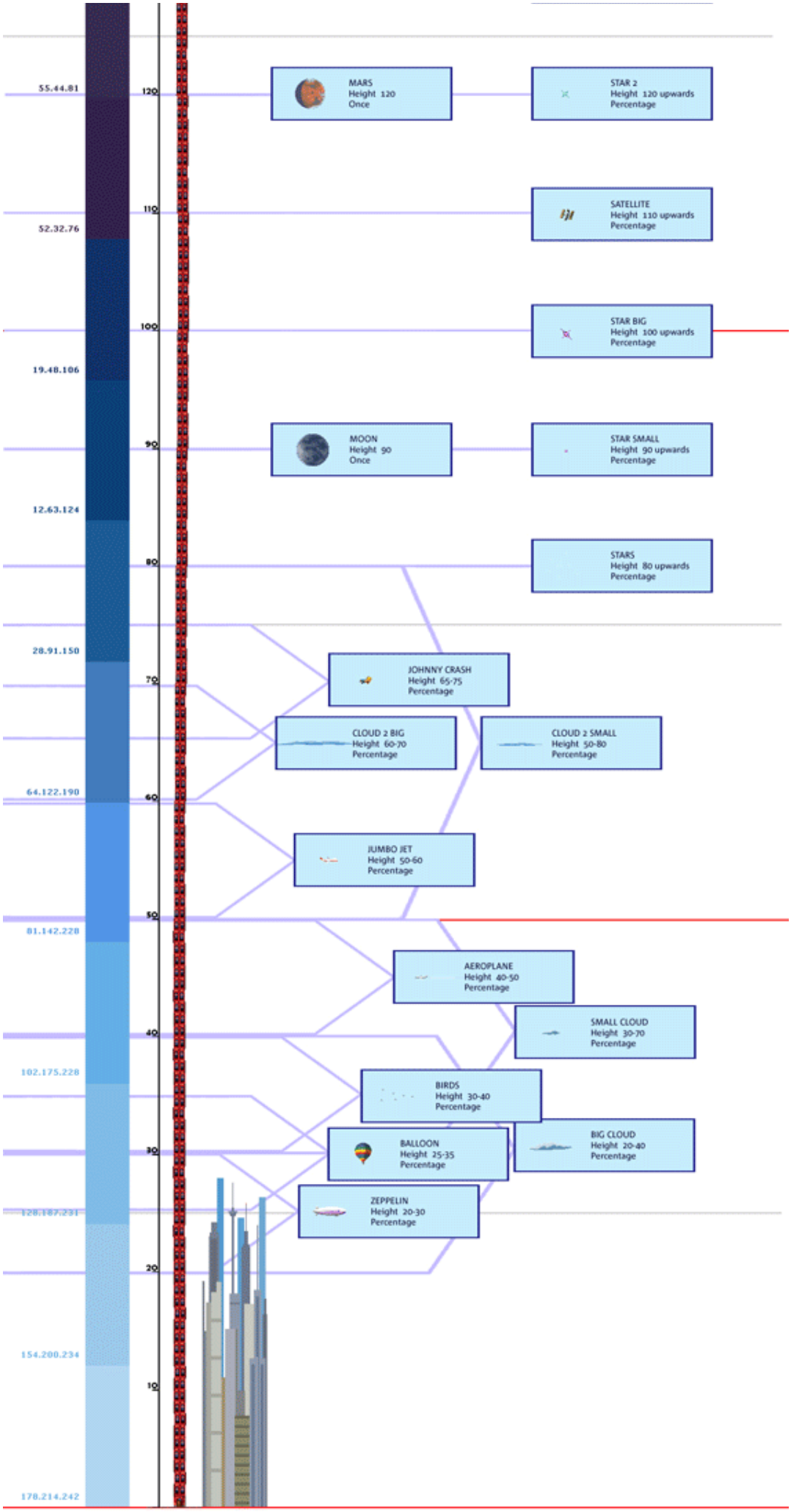
Final Thoughts

We have a couple of points to share with developers out there who are trying to make games that break the mold. First, don't start without having a proof of concept stage; write a few throw-away prototypes to test the core game mechanics, and if you have time, prototype the

graphical UI as well to save some extra work.

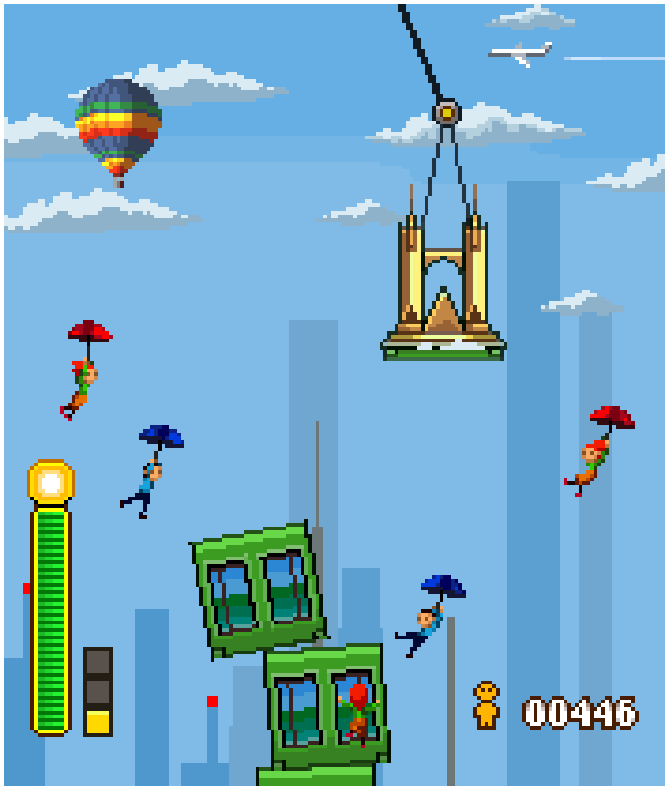
Second, either add some slack to the schedule or consider using change-friendly project management methods such as Scrum, because in these types of games there are always a lot of new things to be discovered and worked out.




55.44.81
52.32.76
19.48.106
12.63.124
28.91.150
64.122.190
81.142.228
102.175.228
128.187.231
154.200.234
178.214.242
120
110
100
90
80
70
60
50
40
30
20
10
MARS
Height 120
Once
STAR 2
Height 120 upwards
Percentage
SATELLITE
Height 110 upwards
Percentage
STAR BIG
Height 100 upwards
Percentage
MOON
Height 90
Once
STAR SMALL
Height 90 upwards
Percentage
STARS
Height 80 upwards
Percentage
JOHNNY CRASH
Height 65-75
Percentage
CLOUD 2 BIG
Height 60-70
Percentage
CLOUD 2 SMALL
Height 50-80
Percentage
JUMBO JET
Height 50-60
Percentage
AEROPLANE
Height 40-50
Percentage
SMALL CLOUD
Height 30-70
Percentage
BIRDS
Height 30-40
Percentage
BALLOON
Height 25-35
Percentage
BIG CLOUD
Height 20-40
Percentage
ZEPPELIN
Height 20-30
Percentage

We firmly believe that mobile games can be a lot more than arcade remakes or just a quick profit opportunity for license owners, and we put our money where our minds are. Building *Tower Bloxx* was a long and challenging journey, but as we were finishing the game we hoped that we had stacked enough blocks and put them in the right order to attract and please a big audience. Judging by its critical reception and its commercial success (it's been our best-selling game in many markets over the past few months), it seems we got some of it right.

And now, it's time to start it all over again!



Return to the full version of this article