

CHAPTER 8.

A SPACE TO CREATE: TEACHING WITH MINECRAFT

BY JASON WILMOT



Figure 1. GR2 Joey builds his ideal classroom in this screen shot from Minecraft.

I am addicted to the notion that all people are creative. I believe we truly find ourselves in the creative process. I believe we find a deeper, richer understanding of our own aptitudes and capabilities when we stop consuming and start creating.

When I walk into my classroom I see brilliance. I see art and artists of all kinds. I see partnership and problem solving. I see architects and engineers. I see developers, coders, and hackers. I see programmers, designers, and advertisers.

But I also see kids just trying to have fun and be happy at school, to learn about life and grow in self-efficacy. I see kids who love to collaborate in ways that excite and expand their imaginations. I see classmates connecting in completely relevant ways.

What I see is that we need more things like *Minecraft* in school. More things that kids are actually talking about. What we need is less of our own agenda and more of theirs, more of their interests and more of what they find meaningful.

I'm not saying that *Minecraft* is the answer, but I am saying that it might be. It might be. It might be for that kid who has not mastered the art of school. It might be for the kid who is tired of being tested and just wants to be a kid again. It might just be that *Minecraft* is exactly what these kids need—what every kid needs: a space to imagine. A space to explore. A space to dare. A space to fail and fail and fail. A space to be bold. A space for resilience. A space to finally get it right.

A space to craft.

A space to create.

A space to build understanding—one little pixelated block at time.

WHAT IS *MINECRAFT*?

Minecraft is a pixelated powerhouse for digital play. It is a sandbox environment that allows users to build and alter their surrounding world. *Minecraft* can be played on a number of devices, including PC, Xbox, PlayStation, iOS, and Android. *Minecraft* is a game created by Swedish programmer Markus “Notch” Persson, later developed and published by Mojang, and most recently acquired by Microsoft.

Minecraft can be played as a game centering on strategic thinking and deductive reasoning. The user must prioritize, set goals, and problem solve to explore the in-game world. Throw in an optional threat of danger (monsters) and this fosters a need for security and necessitates preservation strategies.

Minecraft can also be played in a “Creative” mode. This mode provides for an endless virtual sandbox of creativity (think digital Legos). Here users can imagine, design, build, engineer, program, develop, code, and simulate almost any type of learning environment. Users can play individually, in small groups, or they can play on servers holding hundreds of players.

On top of all that, *Minecraft* also allows for virtual field trips. Here, a teacher could drop the entire class into a precreated world. You'll find ancient Roman dwellings, China's Forbidden City, and Egyptian pyramids. This feature is called “Adventure” mode. It gives students the ability to visit a world but disables their ability to break anything within it. It's a class trip to the art gallery with no need to worry about oily fingers.

How Do Kids Use *Minecraft* in Class?

I started using *Minecraft* in my classroom in the fall semester of 2013 with the help of Guy Trainin, PhD, of the University of Nebraska, Lincoln. Since then, I've used *Minecraft* in several different ways. I've used it for individual, whole-class, and entire-school builds.

We have learned about digital citizenship, computational thinking, and connected the everyday classroom curriculum into student creativity (see Figure 1). Kids have created and solved math problems (see Figures 2 and 13), combined research projects and digital dioramas to model the Great Wall of China, the Orphan Train (an 1800s orphan-relocation program), our solar system (see Figure 7), water-conservation displays, and everything else you might imagine. We have simulated learning goals from Language Arts, done basic programming, and even used it for our Hour of Code during Computer Science Education Week (CSED). With *Minecraft*, the possibilities are almost endless.

What Are the Educational Benefits?

When people ask about the educational benefits of *Minecraft*, I often ask them to clarify what they mean. Yeah, I'm pedantic like that, but "educational" can mean a lot of different things to different people—and really, that's probably right where we should keep it—amorphous. I subscribe to the idea that education can happen anytime and anywhere.

Maybe I'm "old skool," new school, or something in the realm of other. But as an educator, what I hold onto more than anything else is this: *Learning must be meaningful and it must be relevant*. That means content must be presented in real-world ways. It should make sense on an individual level and whenever possible content should be simulated, not just presented in information-only formats.

Our job as educators is not to provide the dots, but to facilitate the connecting of dots. This means taking student experiences and building *into* that specific knowledge base. It means getting to know the students individually. It is discovering what they find interesting, what they believe about the world, and where they find significance. For all these things—for this affinity space—the best place to start was *Minecraft*.

I say this because *Minecraft* is not an information-only medium. With *Minecraft*, information can be presented and then simulated in hundreds of ways. When students can take new information and build into their schema through experiences or simulation—that's where knowledge, know-how, and learning start driving themselves.

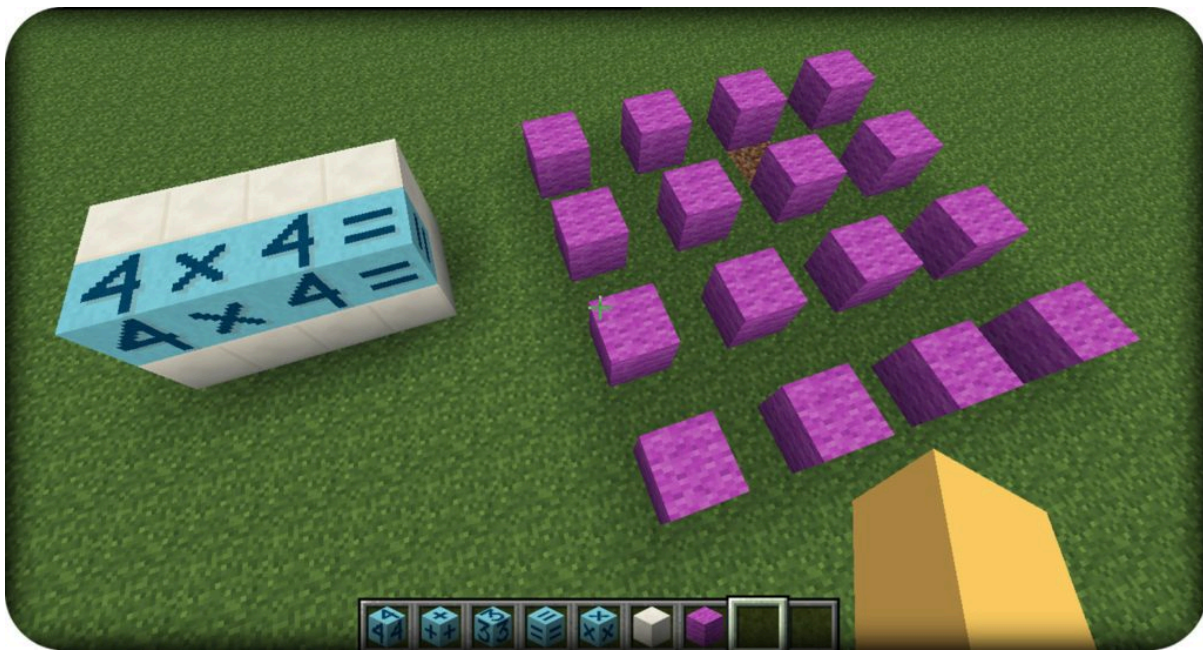


Figure 2. Using Minecraft blocks to demonstrate an understanding of multiplication.

Minecraft Is a Platform

A hurdle I often come across in teaching with *Minecraft* is not merely having to explain what *Minecraft* is, but actually addressing what *it is not*. I start by saying that *Minecraft* is not merely a game, just like Amazon.com is not just merely a website.

Yes, Amazon.com is a website, but you're familiar with Amazon because it is a platform. And that is a very important difference—because a true platform *is a new way of doing something*. Amazon changed the face of commerce, not by inventing commerce, but by creating a new avenue for it to exist. It created a new way for the consumer to interact—almost directly—with the producer. Amazon is a new way of doing something we humans have been doing for a very long time.

But when I talk about using *Minecraft* in education, I think most people probably conjure some greaseball trying to suck down their taxpayer dimes playing computer games. And that's frustrating because—yes, of course it is a game—but that is just scratching the surface. Instead, *Minecraft* should really be seen as a new way of doing something: a platform. A platform that provides a basic structure, yet from that structure one can do, literally, thousands of things.

I think Lego fits somewhere into this little assembly of an argument. Lego has a simple concept: Assemble individual blocks to make something awesome. You can follow the directions or you can be as creative and imaginative as you want. There is no wrong way to play with Lego, just as there is no right way or wrong way to play *Minecraft*—they are both sandbox environments designed for play and imagination.

But here is the contradiction that I find. You would be hard pressed to find individuals who would write off Lego or who would criticize it and label it as a foolish distraction. Yet if you put that same

idea on a computer screen, people will fly into fits and pull the old-fashioned “Well, back in the Gilded Age of Learning when I was in school” blasphemy card.

For some reason, creativity, innovation, and imagination don’t come in pixelated form. For some reason, we have convinced ourselves that real learning comes in the form of textbooks and worksheets. For some reason, if you’re playing and gaming, then you’re obviously not learning.

But I think that’s wrong.

Before we had *Minecraft*, we encouraged kids to play and create with Legos. A little before that it was Tinker Toys and Lincoln Logs. Before that it was wooden cubes, sticks and rocks, or whatever kids could get their hands on.

You see, we humans long to play and create. We desire to learn, build, design, and discover. We always have and we always will. *Minecraft* is no different. It is a platform delivering what kids desire. It allows students to construct their own meaning. It paves the way for students to try and fail, or to experiment with their own imagination and creativity. It allows kids to simulate newly acquired information in completely relevant ways. *Minecraft* makes sense. It’s a platform for learning. It’s a platform for play. It’s a platform where students and teachers can actually connect.

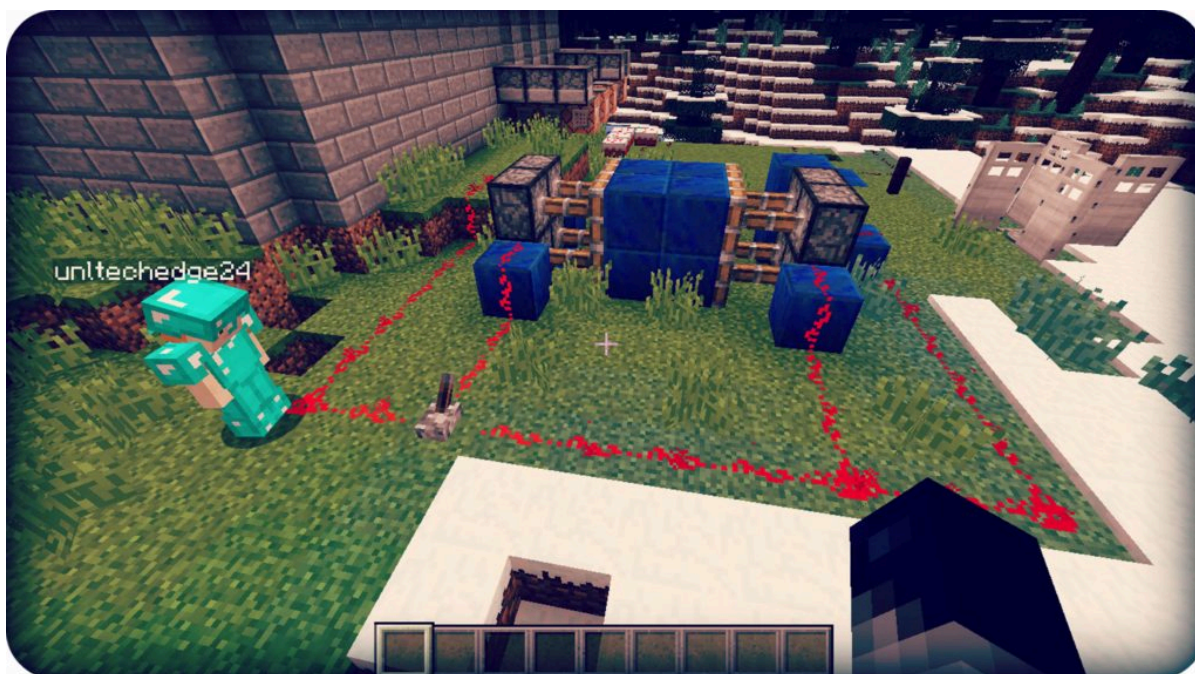


Figure 3. GR5 Imran building a redstone crusher with circuits and a switch to activate the pistons.

Minecraft: Learning by Doing

I recently helped a friend put his IKEA kitchen table together. After half an hour or so of building this table, the ever-so-simple directions walked us through to having a table that stood mostly upright. We beat our chests, reveled in glory, and life was good. Hurray for us. Hurray for dinner. End of anecdote.

Maybe you can’t relate to my built-it-yourself adventure, but here’s the idea: The instructions and the

table are a perfect pair—but separate them and they’re practically useless. Without the directions the table goes unused. On the other hand, if you have all the table’s parts but no directions, the process can easily be stifled.

Author James Paul Gee, PhD, illustrates the same idea with the example of a video game in his book *Good Video Games and Good Learning*.¹ Basically, a prepackaged video game normally comes with two main features: the game itself and the instruction manual. While you can get along just fine without the manual, it does come in handy every now and then when help is sought or needed.

But don’t try giving someone a game manual without giving him or her the actual game. Why? *Because no one wants the manual without the actual game.* What’s the point of reading about the game if we never get to play? What’s the point in memorizing facts in the manual if we never get to simulate or experience it?

You can see where this argument eventually leads when it comes to school. One would argue that we spend far too much time concerning ourselves with content rather than working to blend that information with an experience. In other words, we take the time to read all the manuals but never leave any time to play the game. When we do this, we leave out one of the most essential components of learning: understanding by doing.

If we know that authentic learning happens when we connect experiences and simulations, then we know that methods such as teaching through *Minecraft* make a lot of sense. What if after we read the manual, we could actually play the game? What if we actually gave kids a chance to simulate newly acquired information and turn it into knowledge? What if we could synthesize the information in such a way that kids found it relevant to their lives? What if—rather than just learning about the transcontinental railroad, a Sioux earth lodge, or the Eiffel Tower—what if we could just build them from scratch?

Well, there is a way. It’s called *Minecraft*. It is a game and it is awesome for learning. With *Minecraft*, these kids are teachers, learners, and full-blown artists. Artists who innovate, imagine, and create. Eiffel Tower? We could build that. Earth lodge? Yeah, we could build that, too. We could build a pirate ship, a Pilgrim village, the Colosseum, or whatever else we wanted. We could even build a table.

With *Minecraft*, students aren’t just consuming those manuals either—they are actually making the manuals themselves. Kids are teaching and learning from one another. They are reading wikis that other kids help create. They are starting YouTube channels to share and explore ideas. Kids are creating tutorials and reviewing mods and created worlds to share. They are using mediums they actually find meaningful and applicable. They are creating; they are cultivating. They are inventing and testing, hypothesizing and experimenting. They are learning through inquiry. They are learning through discovery. They are learning by doing.

1. Gee, J. P. (2008). *Good video games and good learning: Collected essays on video games, learning, and literacy*. New York, NY: Peter Lang.

MY MINECRAFT STORY:

Pioneering *Minecraft* in the Elementary Classroom



Figure 4. GR5 Jason builds an entire castle!

I'm that guy who likes his job so much it is annoying. I find meaning in what I do, and for that I'm grateful. I'm lucky enough to spend my day teaching 650 K-5 kids through computers. That's 50 minutes a class, 30 classes a week, and 650 little individual minds.

Technically speaking, I'm a computer specialist at Campbell Elementary School in Lincoln, Nebraska—home to good people, happy people. I know Nebraska and all these Midwestern states have a pretty lame label as flyover states, but, in actuality, Lincoln is a little gem of hospitality. It is home to an economy probably doing much better than yours, it has a vibrant university, and its diversity and culture are ever growing. Plain and simple, Lincoln is in its own coming-of-age story and I'm glad to be in the thick of it.

On any given day at Campbell, I have kids from all over the world in my computer lab. I can practice Vietnamese, Farsi, Arabic, you name it. Congolese French? Check. Tigrinya? I didn't even know that was a language (Hint: It's spoken in Eritrea and Ethiopia.) We have *hijabs* and *bindis*. We have rich kids and poor kids. We have white kids, black kids, and every shade in between. What we have at Campbell is a rich mix of a whole lot of cultures.

I cannot think of a more ideal classroom. I have these amazing kids from all walks of life and all corners of the globe. And sure, they may look different on the outside, they may speak different languages, and they may have different cultural practices, but they're a lot more similar than you would think. Because there's something kids everywhere have in common: They actually like to learn,

just like the rest of us. Because learning is actually pretty simple. Learning is like breathing. We breathe in oxygen, we grow cells; we take in impulses, we connect dendrites. It is natural, normal, addicting, and never ending.

For kids, *Minecraft* is that never-ending discovery process. It's a digital destination for play and exploration. It's an unbelievably great platform for learning. It encourages kids to be curious and imaginative. It unbridles creativity and promotes originality. It forces kids to think critically and in an orderly way. It creates opportunities for programming, partnerships, and provides all the STEM environments you could want.

Minecraft is a giant platform of adventure and quest. It's a space for strategic, prioritized, and executive thinking. It's an area to create, connect, and collaborate. Throw in a freedom to fail, an ability to set goals, and the opportunities to take risks. Mix them all together and you get a pretty epic environment for learning.

But chances are you already knew that. And it's for all these reasons that you probably started reading this chapter. You've peeked into tomorrow and know what these kids will need. I'm suggesting that *Minecraft* is one way—one really good way—to get them there.

So in this chapter, my hope is to make a case for *Minecraft* and to start spreading this idea. Here is how it is going to work: I have already tried to lay down a little philosophy and pedagogy for *Minecraft*. It's not everyone's cup of tea, but it is the heartbeat of why this matters. It is also the stuff you will need to be able to explain to parents, teachers, and principals along the way.

Next, I'd like to fill you in on my story; it's brief, will help you know the setting of the story, and I hopeful will fill in a little of the background. Last, I'm going to list everything you ought to know if you want to get started, large or small. I even have links and a QR code for example videos and tutorials.



Figure 5. GR4 Joel and Melina build a redstone-powered roller coaster that shoots fireworks when triggered.

Here's Our Story

Before I was a computer specialist at Campbell, I was a fourth-grade teacher for six years. I made the move for a number of reasons, but one of my primary motivations for the switch was to learn how to further embed technology into education. But it was more than that: I didn't just want to sprinkle in techie gadgets or electronic worksheets. I wanted to find where curriculum, creativity, and student desires could intersect.

Upon my arrival at Campbell Elementary, one of the first things I did was try to get to know the kids. Plain and simple, I wanted them to be involved in their own school day. So, I explained, "A lot of times in life, there are things we *have* to do before we do the things we *want* to do." It's in this way I introduced our curriculum for the year. I told the students the curriculum requirements, but I also informed them that I had left plenty of room for their ideas.

All week long I met with 650-plus kids and took inventory of what they wanted to do. I heard great ideas, such as making movies, memes, and music. So we did that. Kids wanted to learn how to edit photos and create digital art. So we did that too. The ideas were fantastic. We grew in our understanding of digital literacies, but we also connected school with real-world experiences and everyday life.

But there was one other thing we talked about. Grade after grade, class after class, what the kids really wanted was to play *Minecraft*. Just like the students at my former school, the kids at Campbell had caught wind of *Minecraft*.

Fortunately, their new computer teacher had heard of *Minecraft* too. Between the transition from my old setting to the new one, I started researching *Minecraft*. My former students had convinced me to download it on the classroom iPads and taught me how to play. It didn't take long to see the potential it had for the classroom.

MINECRAFTEDU TO THE RESCUE

So when my fourth-grade classroom introduced me to *Minecraft*, I couldn't wait to pay it forward and introduce an entire school to it. But as it turns out, *Minecraft* doesn't hand out copies for free (maybe someday it will). What I needed was 30 copies/licenses for my entire computer lab. I did some simple math and it got pricey really quickly. At the time of this writing, one license of *Minecraft* will run you about \$27—that's a little steep when trying to fill an entire computer lab.

Luckily, you and I have a lovely little tool called the Internet. After doing some research and digging around, I came across *MinecraftEdu*. Essentially, *MinecraftEdu* is a modification, or mod, of regular *Minecraft*. Basically, a mod changes what the game looks like or enhances how it is played. Now, loads of mods are out there in the *Minecraft* world, but *MinecraftEdu* puts some pretty handy controls together and makes it easy enough for almost anyone to get started.

And here's the magic sauce: If you buy *MinecraftEdu*, you also get full access to regular *Minecraft*. On top of that, it all comes at half the price. The more copies of *MinecraftEdu* you buy, the cheaper they get. One copy is \$18—not bad; 25+ copies are \$14 each; and 100+ copies are \$9 each. It's affordable, which means it's doable. And it was all the motivation I needed to keep this crazy idea rolling.

\$2 Closer

Upon this delightful discovery of good news, I was all the more convinced that getting *Minecraft* at Campbell was something we could pull off. We needed about \$450 to get started and that seemed pretty realistic. So, I took the idea to the kids to get them in on the conversation. I wanted them to have a stake in this project and that meant being transparent about the cost and about any obstacles along the way. That meant being open about cost, logistics, hurdles, and troubleshooting. It meant asking my kids for help and for input. And that's exactly what I got. Because when I presented these hurdles to hundreds of little kids, I had hundreds of little minds actively trying to solve them.

My favorite idea was from a fifth grader, Nick. He came up with the idea of starting a schoolwide fund-raiser. To Nick, raising \$450 in a school of 650 kids seemed all too easy. Nick explained that if each Campbell student were to bring a dollar we could have enough money to buy *MinecraftEdu* for our entire computer lab.

In fact, he followed it up the next week by stopping by the computer lab with two \$1 bills in hand. He gave me the money and reminded me of his plan.

"But, you have two dollars there," I said, rather astutely.

"Oh," Nick said, "Well, yeah, I brought two dollars. You know, just in case someone can't afford to donate a dollar."

What a champ, that Nick. I didn't take those two dollars. But I should have. It's good for us to contribute to causes we believe in. It's healthy to give. That would have been a teachable moment or something like that. Anyway, I didn't take Nick's money because I didn't need it: Someone had just footed the whole bill.

Guy Trainin, Extraordinaire

My previous year of teaching had been a fantastic one. During my sixth year as a classroom teacher, I started building a relationship with two amazing souls at the University of Nebraska-Lincoln (UNL), Guy Trainin, PhD, and Laurie Friedrich, PhD. I met both via UNL TechEdge, a conference filled with tech-savvy teachers, emerging ideas, and top-shelf speakers such as Dr. James Paul Gee. As with any quality conference, I left filled to the brim with new knowledge, new goals, and two would-be friends.

It was Professor Trainin whom I approached with the idea of using *Minecraft* in the classroom. I told him of the idea, laid out my goals, and told him my plan to get there. More than anything, I just wanted to know his opinion. Had I known walking into the conversation that he would be my benefactor, I probably would have practiced my elevator pitch a little better.

So after I told him my plan and my goal to get *Minecraft* into the classroom, Professor Trainin asked me how much it would cost. After I told him my plan to raise funds, he stopped me in my tracks.

"I'll fund it," he said.

Um. What?

“I’d be willing to fund the project if we could also study the educational benefits,” he explained.

And in that instant, the idea of bringing *Minecraft* into the classroom started looking more like a reality. Instead of focusing on the fund-raising, I could start focusing on getting the plan in place. I walked into that conversation looking for advice, but I left with so much more.

And maybe the takeaway is this: Talk to people about your ideas and ask for help—especially if you think those ideas are really worth it. Going after your ideas may mean looking outside your personal resources. But the reality is this: People will always be interested in good ideas. Plenty of individuals and organizations want to invest in young minds, in their education, and into the hopes and dreams of tomorrow.



Figure 6. GR4Hamza creates a fire maze for his friends and classmates.

The Spinning, Spinning, Spinning Beach Ball of Death

It’s wise to remember that things move slowly down the bureaucratic assemblage line. After getting the great news from Professor Trainin, I ended up with months of waiting. We had to get all the yes sirs, yes ma’ams, and signatures from an assortment of folks. I had to talk with my principal, parents, district administrators, and university partners. And while the process took way more time than I had anticipated, in the end I had all the OKs I needed. The green light was finally given. *Minecraft* at Campbell was finally a go.

I Got Lemons. That’s What We’ll Call It.

When life gives you lemons in little blocks and pixels you must right-click and squeeze. But when those lemons look and sound a lot like grad students, you smile, shake their hands, and offer those poor kids some coffee.

When Professor Trainin and I had originally discussed this project, he asked one of his graduate students, Ji, to help out. Ji had come from China looking to pursue his doctorate in technology and education. He was interested in digital citizenship, collaboration, and educational gaming.

Ji and I had connected after hearing the good news and started setting up my computer lab to get things going. He arrived and we started talking servers, IPs, and RAM. I immediately started feeling overwhelmed, a bit ignorant, and even a little embarrassed. Here I was, trying to implement this game played over servers when I didn't even know how to set up a server myself.

But this is what learning looks like: We ask for help, we acknowledge our own shortcomings, and look to others for resources. Fortunately, there are tons of people like Ji—people who are patient enough and humble enough to show even a computer teacher (of all people) a little thing or two about computers.

Ji and I discussed other issues, too. We had to manage user profiles, maneuver directories, and find out the specs for 25+ kids on the same server at the same time without crashing everything. Yet little by little, Ji and I deciphered this little Swedish computer game. He researched server IPs while I registered accounts. We worked. We tested. We fixed. Rinse and repeat. By working together, in a matter of a few days we had everything we needed to push the “Go” button. At least we thought we did.

Advice: Get to Know the Folks You Don't

Then a brand-new wave of problems occurred. Basically, I had trouble accessing the *MinecraftEdu* launcher on our computer-lab student accounts. This meant kids couldn't all connect in whole-class settings. Instead, if our whole class wanted to connect, we had to use regular *Minecraft*. That, in turn, meant having TNT, lava, and fire as common occurrences. And although the mass destruction led to great discussions of governance, property, and digital citizenship, the students were having a hard time buying into the creation of digital art only to have it blown to pieces moments later.

This is when I really got to know and appreciate our tech gurus at Lincoln Public Schools (LPS). Just as advertised, these computer ninjas got me all the help I needed. They customized our app launcher and even surprised us with a super-stable virtual server. This meant I could start a *MinecraftEdu* server and use the in-game controls to stamp out fire, TNT, and all those little urges to grieve and destroy.

These tech masterminds never get the credit they deserve. They're not teachers or principals or department heads, but they are so essential to our schools. They problem solve, they troubleshoot, and they think outside the box. They are the people who make our modern age work. These behind-the-scene-ers deserve awards, our applause, and troves of Bitcoin.

While I was lucky to have some much-needed help, I also consulted a lot of forums online. Even with no tech support, help was always only a few clicks away. We live in a day and age in which affinity spaces have self-supporting resources. *Minecraft* has an absolutely huge web presence. You will find wikis, YouTube tutorials, Twitter, and a Google community dedicated to all things *MinecraftEdu*.

The Paradox of Choice

After months of waiting, planning, hair pulling, mouse slamming, and everything in between, the kids at Campbell finally had *Minecraft*. The major issues were tackled, the servers held stable, and the kids knew the high expectations. The time to play, create, and craft had finally come.

And I think this is where the project's vastness hit me all at once. All of the prep work, all of the defending, convincing, and pursuing—all of that was just the first step. Now, I had to actually implement it and prove it useful.

This is where the work of an entrepreneur, a visionary, a pioneer—this is where work can get frustrating and isolating. I think it's because we are so apt to envision a new way and so inclined to welcome fresh ideas, but actually bringing that change to fruition is a really difficult feat. We can get so caught up in the lofty vision that when it's time to implement the idea, we find ourselves facing an impasse of our own unrealistic expectations.

And that's where I found myself regularly: treading water in this ocean of ideas, not sure where to go. There were so many different ways to implement *Minecraft* that I didn't know where to start. It was the Paradox of Choice: You know, there are so many options that you're unsure which one is best. The worst part is struggling to actually embrace any game plan because it may not be the best one. So, here I speak to you, astute reader and aspiring *Minecraft* teacher: Don't get caught in this mind-set. Trust your gut. Make a plan and stick to it. Ask for help. Don't be a hero or a go-it-aloner.

Ultimately, I decided to just go. And what that looked like was just my stepping back. I wanted this to grow organically, not in a contrived or conditioned manner. I didn't try to rush in my own agenda or attempt to shoehorn *Minecraft* into some standardized curriculum. So, at first we just played. We learned the controls, we stretched our imaginations, and we discovered what we could make this game do.



Figure 7. Crossing curriculum: Fourth-grade girls team up to build a representation of the solar system.

Collaborative. Creative.

I want to leave you, reader, with a taste of what you can actually do with this game. It's hard to know where to start, so I want to run through a few highlights to get your imagination going. We've engaged in whole-class projects of giant roller coasters (see Figures 5 and 20), and we have built arenas, stadiums, and fire mazes.

The biggest project we have pulled off was a schoolwide build properly named Campbell City. The city project was actually done pretty early on in our adoption of *MinecraftEdu*. To be honest, I wanted to do something schoolwide and I wanted to make it showy. I wanted to show skeptical parents, doubtful teachers, and any onlooker what these kids were capable of doing.

Campbell City is a feat. It's a staggering collection of thousands upon thousands of textured blocks. Comprising 500+ individual plots with 500+ brilliant, individual ideas, each grade holds its own subsection of the community. Within that grade-level community, each classroom has its own street. Alongside that street, all students had their own plot of land where they were allowed to imagine, design, and build whatever they could dream up.

Campbell City includes mini-theme parks (see Figure 21), movie theaters, pools, restaurants, galleries, and shops. Students conjured pyramids, giant tree houses, castles (see Figures 4 and 18), statues, hotels, parks, and fountains. There were skyscrapers, mansions, train stations, museums, and loads and loads more.

Overall, it was an all-consuming, collaborative, self-governing adventure. We talked about governance, consensus, and digital citizenship. Kids worked together, laid ground rules together, and inspired one another to be innovative and original. We built together, learned together, and saw how much potential this game really had.

Continuous, schoolwide builds aren't always all that practical, though. I still had commitments to curriculum and standards to meet. So we started working in small groups for smaller projects. Within these small groups, the greatest variable I didn't see coming was the collaboration: real, genuine, cooperative discovery. The kids learn and work with one another. They team up, they set goals, and they even arrive at class with plans in hand.

Students have done collaborative research projects, biography builds, Google Earth re-creations (see Figure 19), social studies connections, digital dioramas, and history lessons. They have mapped our solar system (see Figure 7) with glowing planets and surrounding asteroids. They have built the Orphan Train, Pompeii, the Great Wall of China, pyramids, functioning lighthouses, petting zoos, malls, hotels, theme parks, bomber planes, and pirate ships (see Figure 11). Students have programmed turtles to mine tunnels for them and have written code to build structures automatically. They have created incredible command block and redstone projects that contain so much ingenuity that it baffles the hired “expert” in the room.

That's just a taste. The list grows every day we fire up *Minecraft*. You see, there is no right or wrong way to use *Minecraft*. It's a platform, a medium, a digital meeting place. So if you don't know where to start, just stop and listen to your students—they will give you all the ideas you need. Sure, you can weave in your pedagogy and your philosophy, as you must. But for now, just let the kids show you what they can do. Sit with them. Build with them. Create. Play. Enjoy.

One Last Bit

So what I am going to do is encourage you to take the next step: Just get started. No more waiting, no more “maybe tomorrows,” no more excuses. In the next part of this chapter, I have provided everything you need to get started. I will give you the tips, tricks, and all the key info you need to know in order to start teaching with *Minecraft*, whether it is just a handful of kids or an entire classroom. On top of that, I will provide you with links to videos I have created that show you exactly how to set up a server, back up your digital worlds, and even import new worlds into your classroom.

So this is it; this is where I leave you. This is where my story ends and where your story begins.

But first I want to thank you. I want to thank you for taking a risk, for doing something fresh and new. I want to thank you for inspiring minds and encouraging hearts; your classroom is lucky to have you. You are the reason students come to school smiling: You instill dreams, incept ideas, grow minds. You are exactly who your students need. Thank you for who you are and all you do.

Now, go have some fun.

MAKE MINECRAFT HAPPEN IN YOUR CLASSROOM:

A How-To and FAQ

This last section comes from a resource I created to help anyone get started teaching with *Minecraft*. By no means is it an exhaustive how-to, but it should cover all of the basics you come across. I would also recommend visiting the QR code (see Figure 8) for video tutorials, a Parental Permission Form, a printout of the controls, blog posts, as well as any updates that may have come after this publication.



Figure 8. JasonWilmot.com/minecraft

Your Need-to-Knows:

1. Learn how to create and customize a world.
2. Learn how to open a LAN connection.
3. Learn how to set up a server.



Figure 9. GR3 Cian builds a refrigerator that shoots meat!

Getting Started: Singleplayer and Multiplayer

Singleplayer

Select this to start a solo campaign in either Creative or Survival game mode. Singleplayer is to be used for solo playing, but it also allows the user to open up his or her game to a small group of players using the local area network (LAN) feature.

Multiplayer

Go here to join an open LAN world or to connect to a server address, or IP.

LAN worlds will automatically be detected upon selecting “Multiplayer.” To join an open server, users will Direct Connect or add a server by inputting the known IP address.

LAN Connect

A local area network can be created to host a small group of players on one local network. The LAN world is hosted on the computer that starts the Singleplayer game. With a LAN world there is no need to set up a server. Servers are only for groups of kids greater than a LAN connection can handle. Ideally, I wouldn't be running any more than five to six kids per LAN, but this depends on your hardware.

Tip: A LAN connection is a great place to start playing with others if you do not understand how to set up a server).

Tip: Up to eight players can play in a LAN world. Remember, the game performance has greater lag with every additional player.

Tip: If you are having trouble detecting LAN worlds, check your security settings to allow Java through your firewall.

LAN Step-by-Step

Want to open a LAN world for small-group gaming? Start a Singleplayer world and press the “esc” key to pull up your menu options. From there, select “Open LAN.” This next screen determines the settings for those who join. To launch your LAN world, simply press “Start LAN World” when finished.

Game mode determines type of play, while “cheats” refer to operator abilities. If you want all players to have operator abilities, turn cheats to “On.” I would recommend cheats on for Creative but off for Survival.



Figure 10. GR5 Reagan builds a redstone-powered circuit to open and close a sliding door.

Game Modes

Survival Mode

In this mode players must gather and mine resources in order to craft items. A sense of discovery and strategy are prerequisites. Throw in the optional threat of danger (i.e., monsters), and this fosters a need for security and necessitates preservation strategies.

Tip: The most productive way to use Survival mode is to have students preplan their goals for the day. Think project-/purposed-based learning.

Creative Mode

Creative mode is the mode in which players can access all blocks and are limited only by their imaginations. This mode is great for creative builds, programming endeavors, and redstone designing.

Tip: You can actually switch between game modes at any time. I've provided the actual game commands under the "Commands" section a bit further below.

Adventure Mode

This is yet another crazy, amazing aspect of this game. Adventure mode is a mode for virtual field trips. People all around the world have created *Minecraft* worlds that you can visit. There are ancient Roman dwellings, China's Forbidden City, puzzle worlds, coding-based worlds, and even creations such as the Wonderful World of Humanities—a vast, virtual world for exploring ancient civilizations. Adventure mode gives students the ability to visit a world but disables their ability to break blocks.

MinecraftEdu Mode

This is another mode that comes within the bundle *MinecraftEdu* sends your way. In *MinecraftEdu* mode, users will have Survival settings but will not need to worry about health, hunger, or PVP (player versus player), as these can be disabled.



Figure 11. Second graders build themselves a pirate world.

World Types

You have a number of options to customize your world in *Minecraft*, but if you're just getting started, then keep it simple: Stick with either a default or superflat world. A default world includes various landscapes and biomes. A superflat world is a flat world without biomes or landscapes.

Default World

I recommend starting within a default world the first time anyone plays *Minecraft*. There is something about being dropped into a huge gaming environment that only the default world can provide. The sheer vastness of the world is incredible. There are biomes of grass plains, swamps, mountains, and jungles, just to name a few. Want to wow new players? Start them in a default world.

Flat Worlds

If you want to do a creative build, then a superflat world is a better way to go. These superflat worlds are perfectly flat and seem to expand forever. Another plus about these worlds is that there are no mountains of blocks to weigh down your server.

How to Select Your World

After selecting "Singleplayer," you have the ability to customize your new game by pressing "More World Options. ..." Next, you have the ability to turn on cheats, structures, and bonus chests. You can

customize your world type by selecting “World Type.” From here, my advice is just to explore your world options. Again, if you just want to get started, keep it “default” or change it to “superflat.”

Tip: A huge bummer about a superflat world is that it is generated with a very shallow digging depth. A lot of new players will fall through the world just by digging. If you want a deep world with a flat surface, try using the Redstone Ready or the Tunneler’s Dream presents found under Create New World > More World Options ... > World Type > Customize > Presets.

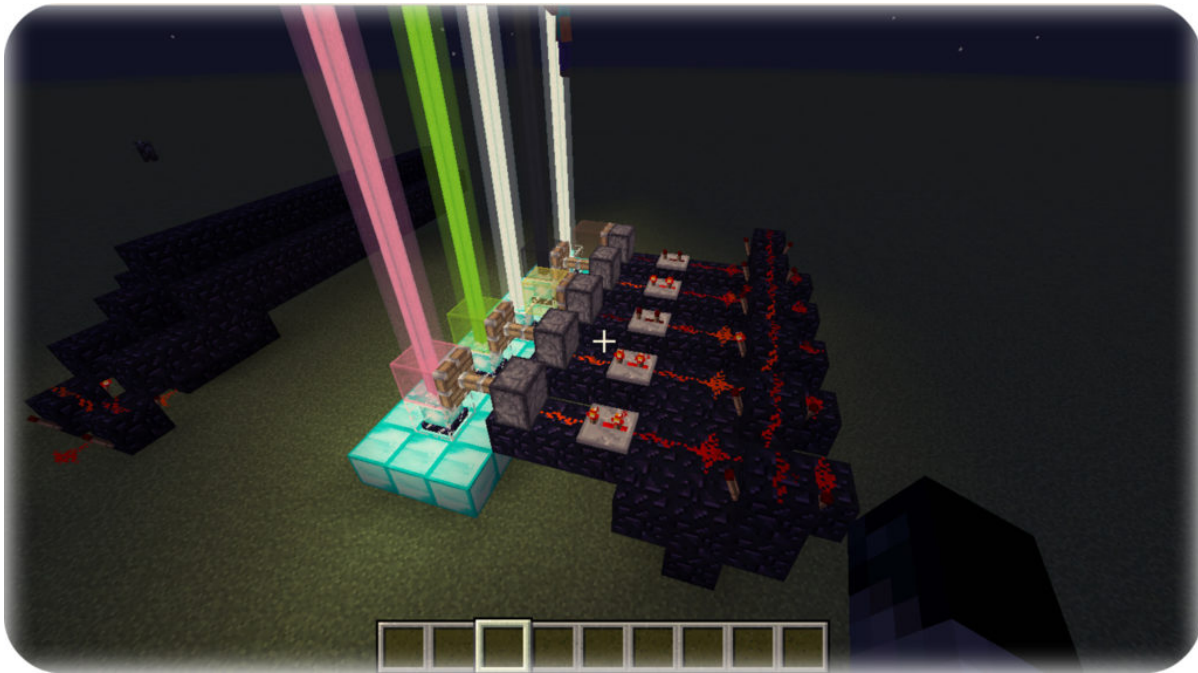


Figure 12. Repeating redstone-powered pistons change color and light up the night sky.

Commands

Make Minecraft Do What You Want

One of the best features of *Minecraft* is the ability to pull up the command prompt and tell the game what you want it to do. For beginners these are commands you should probably know:

`/gamemode 0` (Changes game to Survival mode)

`/gamemode 1` (Changes the game to Creative mode)

`/time set 0` (Changes it to daytime)

`/time set 13000` (Changes to dusk)

`/weather clear` (Stops the rain)

`/tp <username> <username>` (Teleports one player *to* another)

Teachers should encourage students to learn these controls, too. The more you enable kids to do, the

less you need to be the Operator, or OP. Plus, it's way more fun to build and interact with the students than being the continual source of daylight. Pawn that off. You've got better things to do.

You Need to Know About TP

TP stands for "teleport." An operator can teleport any one player to another, and you'll probably need to know how to do this on Day 1. Here's how it works: Pull up the command prompt by pressing the / key and then type in the letters "tp" (for teleport). The teleport feature always transports the first player to the second. Here is an example: `/tp playername1 playername2`. In this case, Playername1 will be transported to Playername 2.

Tip: Instead of writing out the entire name of the user, type in the letters "tp" and then press "tab"; it should pull up the names of users on your server. You can also start typing in the first letters of any username and press "tab" to finish the name.

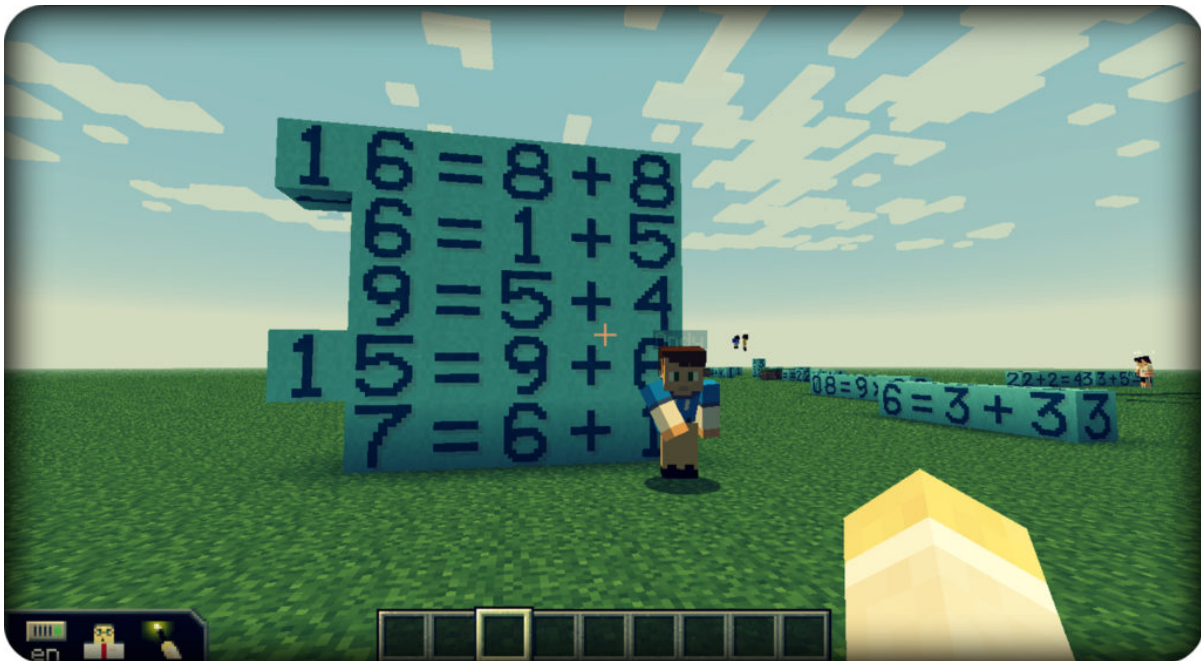


Figure 13. GR3 Andy shows off his math skills.

Servers

A server is a node on the World Wide Web. It exists as a digital space where users can connect. Every server needs an address to exist. This address, or space, is designated by its Internet protocol (or IP) address.

You don't need anything fancy for a server—you can even use your own computer. If it's a newer computer and has at least 4-8 GB memory (RAM), you could probably run about 15 users on a default world server.

I started out using two servers, each running on 2008 iMacs with 1GB RAM. In other words, you don't need much. These computers would run 8-10 kids and would crash about two to three times a period. Obviously, a server's crashing isn't ideal, but it's not the end of the world because it takes only a

few seconds to start things back up. If and when those servers go down, remember—it gives excellent teaching moments about servers, IPs, RAM—all that stuff you should be teaching kids anyway.

Tip: The server/computer must monitor everything you generate in Minecraft. The more blocks you lay, the more animals you spawn, the more TNT you blast away—all of that takes up memory. Everything taking place determines the need of memory, and this determines the quality of the environment. (A default world takes more computer memory than a superflat world does.)

Setting Up a Regular Minecraft Server

There is an initial learning curve when setting up a regular *Minecraft* server. You can customize your server within your *server.properties* file. It may take a while to figure out everything you can do, but I'll give you the basics to get you started. Also, rather than writing you through the process, I would recommend viewing my video tutorial. Check it out with the QR code or link I provided.

- **IP address.** If you are directing students toward your own server, make sure the IP you provide is the same as the IP within your *server.properties* file.
- **Difficulty.** Difficulty refers to the level of intensity for mobs and monsters. If you want to get rid of them all together, set your difficulty to Peaceful.
- **PVP.** This means “player versus player.” If you do not want kids to be able to attack one another, set PVP to =false. If you're using regular *Minecraft*, this will be found in the *server.properties* file in your server folder.
- **Game mode.** The Game mode determines the type of environment for all players. Game mode = 0 is Survival; = 1 is Creative; and = 2 is Adventure.
- **Stopping a server.** If you are running a regular *Minecraft* server, don't just close out of it when you're ready to stop it. Instead type in the command “Stop.” This will save your game and bring things to a close, all right and proper. If you're running a *MinecraftEdu* server, the software allows you to save your world and everything created within it.

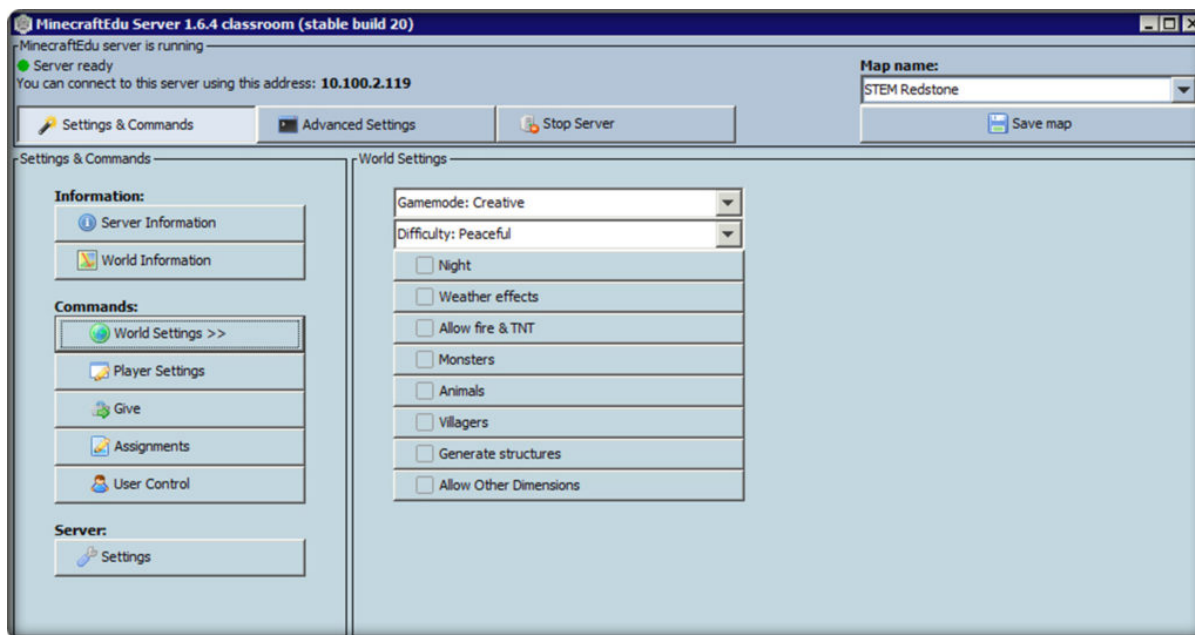


Figure 14. The *MinecraftEdu* server is easy to set up and even easier to manage.

The *MinecraftEdu* Server

The *MinecraftEdu* server tool (see Figure 14) makes starting a server, managing users, and inputting commands super simple. There are many customizable features built within the software and it allows for a quick and easy start. Whether you're running your server on a computer or virtual IP, the controls and options on the *MinecraftEdu* server tool aren't too difficult to learn. Within this server you can change your world settings, control the command prompt, provide assignment or building information, manage users settings, give users items, allow and disallow users to join, and even start players off with a message of the day. Within the *MinecraftEdu* server, you can even launch your old worlds and save any updates. Again, I would recommend viewing my video tutorials for a more detailed explanation.



Figure 15. *MinecraftEdu* comes with a unique and useful set of blocks.

MinecraftEdu Teacher Menu, Commands, and Blocks

Eventually, you may want to learn how to use the teacher menu, “copy,” “cut,” and “paste” commands, as well as *MinecraftEdu* specialty blocks (see Figure 15). Again, this is not something that you need to know to get started, but learning the *MinecraftEdu* teacher tools will make your life easier as the facilitator. I’ve made tutorials explaining the teacher menu and commands, and I have compiled a playlist on the *MinecraftEdu* specialty blocks. Information about the *MinecraftEdu* blocks is given below.

MinecraftEdu Blocks

MinecraftEdu comes with nine additional specialty blocks and 19 number blocks. Really, I see two ways of figuring out what these specialty blocks do: Try the do-it-yourself (DIY) trial-and-error approach, or explore a world from the *MinecraftEdu* World Library and see how other educators have used them. This will give you a better understanding of how these specialty blocks can be used.

According to *MinecraftEdu*, the additional blocks and their descriptions are:

- **Build Allow block.** Used to set up areas where students can build even if student building is disabled in the world.
- **Build Disallow block.** Used to set up areas where students cannot build even if student building is enabled in the world.
- **Border block.** Used to set borders.
- **Information block.** Stores messages inside it. Used to show information, messages, links ...
- **Information sign.** Used to inform students that there’s something important near this block.

- **Foundation block.** Foundation blocks can be helpful when you are building large, symmetrical buildings. They measure distances to other foundation blocks in three-dimensional space.
- **Spawn block.** Defines the starting point where all new users enter the world.
- **Home block.** The home block can be placed anywhere within a world by teachers or students. Once placed, the player may teleport back to the home block any time he or she wishes by accessing the student menu. Home blocks can be broken only by teachers and the students who originally placed them.



Figure 16. GR5 Ethan programs an automatic TNT launcher.

Redstone

Want to grow scientists, programmers, and engineers? Use redstone. Redstone is used to create circuits that perform tasks. Circuits can be created by using redstone dust and redstone component blocks. Redstone can be used to turn on lights, open doors (see Figure 10), program pistons (see Figures 3 and 12), power mine carts, power repeating TNT launchers or dispensers (see Figures 16 and 22), create mini-games, puzzles (see Figure 6), and anything else you can conjure up (see Figure 9). This is a major STEM connection.

Like anything else in *Minecraft*, it takes a little time experimenting with redstone to get the hang of it. I have created a few video tutorials demonstrating pistons, pressure plates, repeaters, dispensers, command blocks, roller coasters, and more. The key to using redstone is first seeing what it can do. I suggest watching some YouTube videos or having experienced students show one another. There are videos showing basic first steps for redstone and others that go a step beyond to incorporate signal repeaters, power supplies, and programmed dispensers.

If you really want to hone computer-science skills, try incorporating coding and redstone at the same

time. I have used the mod ComputerCraftEdu and combined it with pressure plates, pistons, signal neutralizers, and dispensers full of goodies.



Figure 17. ComputerCraftEdu mod: Minecraft gone Logo.

Mods

A “mod,” or modification, is an alteration of the program code that causes the game to operate in a way different from its original version. Mods vary in what they can do—from changing the look of all your blocks to pulling in new items, abilities, and game scenarios. Chances are that kids will ask for mods immediately, but really, you don’t need mods to get started. However, mods are something to look into after a while. Mods offer more options, and more options mean more possibilities.

Importing mods is easy using *MinecraftEdu*. With a few clicks, the mods can be downloaded and installed. A favorite mod I use is ComputerCraftEdu (see Figure 17). The mod finds its roots back in the days of the Logo turtle, but it blends this logic-based programming into the learning-friendly medium of *Minecraft*. With this turtle mod, students use drag-and-drop coding to program the turtle to move, dig, drop, build, and more.

Students can build stairways into the sky or dig a series of holes, only to have the turtle turn around and fill them back in. Students can program a hide-and-go-seek turtle, house-building turtles, or even sync up their turtle to drop materials onto a pressure plate—activating redstone circuits to launch fireworks into the air.



Figure 18. Fifth-grade boys team up on their semester-long building project.

Backing Up and Importing Worlds

You are bound to run into hurdles, obstacles, and unexpected events when teaching with *Minecraft*, but you can ensure your worlds and creations are safe with just a few clicks. It's always wise to back up (or save and store) your worlds. As mentioned above, I have tutorial videos showing you how to do this. The tutorial videos walk through how to save your worlds and how to load previously saved or imported worlds.

If you're using *MinecraftEdu*, there is an alternative way to import worlds. *MinecraftEdu* hosts a large collection of worlds to use in your classroom, absolutely free. Within its World Library, there are worlds for coding, programming, puzzles, and problem solving, and worlds for historical and cultural exploration. The library continues to grow as more users load more educational resources. I've created a tutorial on how to access and download worlds from the *MinecraftEdu* World Library.

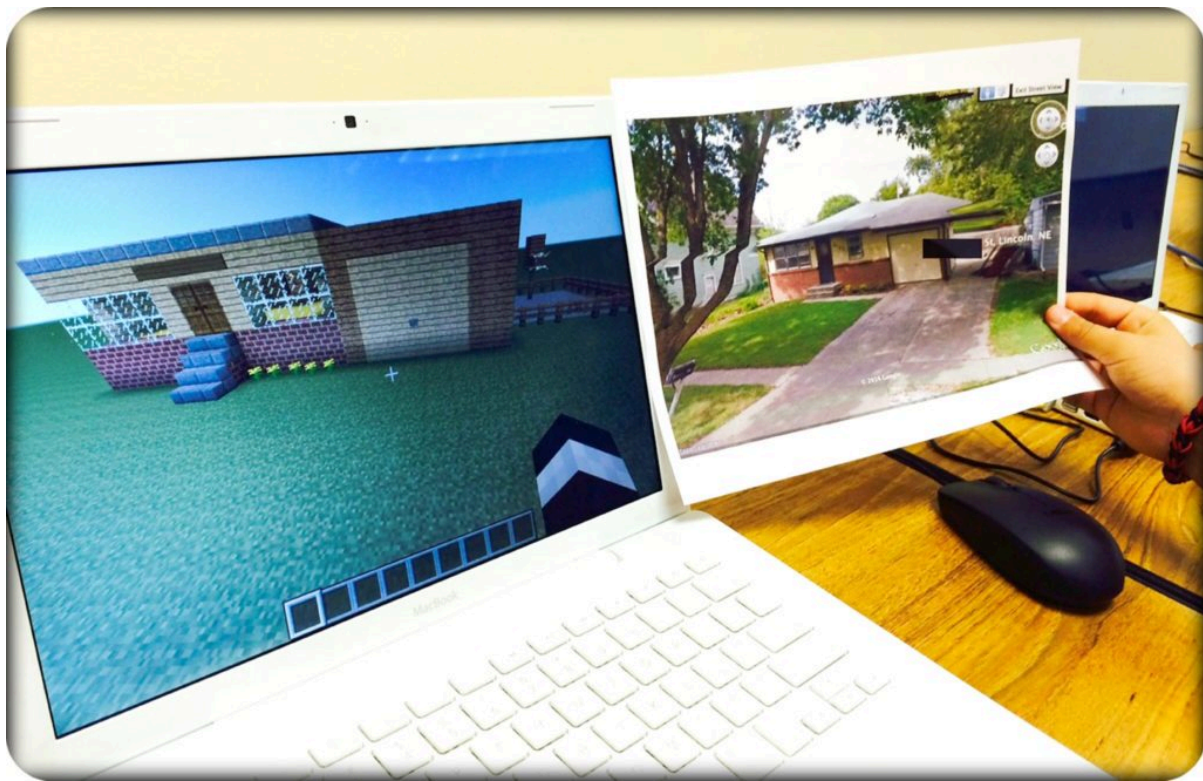


Figure 19. Google Earth and Minecraft: GR5 Nick builds his own house.

Your First Minecraft Lessons

One of the most common questions I get asked is what to do for the first lesson. Obviously, it depends on your goals and outcomes. However, there are a few things I would suggest. First, just let the kids play. The experts will naturally find leadership roles in mentoring the newcomers. Sometimes the best thing you can do is just to let the kids learn from one another.

Another idea would be to have students build a house (or any structure) in Creative mode. This will help students to explore the inventory and learn the blocks. This will also allow them to fly around and explore the game controls.

The last idea for a first lesson would be to use *MinecraftEdu* Tutorial World. This tutorial world guides users through the controls, basic mining and crafting, and has some really cool elements to explore. Best of all, this world is free. Simply press the *MinecraftEdu* “Server Launcher,” and then select the option that reads “Start Server with Tutorial World.”



Figure 20. Kindergarten coaster time!

Collaboration

Collaborative Projects

There is no written rule on how to best go about collaborative projects, but I want to throw some more unsolicited advice your way. I have found whole-class environments are hard to facilitate day in and day out. Granted, we have created some pretty amazing projects, but I don't make whole-group projects a routine thing. It obviously depends on your objectives and learning outcomes. Do what seems right for you.

The space where I have found most success for collaboration and digital citizenship is within small-group LAN worlds. When kids are grouped or self-group into teams of three to four, you see some pretty amazing things. You'll hear dialogue of planning, problem solving, delegating, and goal setting. All in all, if you want to get started feeling successful, stick with small groups at first.

Minecraft Mentoring

One of the things I love most about teaching with *Minecraft* is seeing students mentor one another. If at all possible, try to get kids teaching one another. I regularly have students spending time in my room mentoring players younger than they are. It makes for amazing learning opportunities, student leadership, and self-discovery. Last, it makes the kids feel like pros. And who doesn't like feeling like a pro?



Figure 21. Kindergarten Chris builds an entire theme park for his classmates.

Connect

Teachers should collaborate, too. With the ability to connect over servers, your classroom just grew worldwide. With *Minecraft*, we can collaborate with other classrooms, schools, and teachers across the globe. Ask around and look for opportunities to connect with schools regardless of their geographical location.

How Can I Connect With Others Teaching Minecraft?

The best spot to start is on *MinecraftEdu's* community page,² or in the Minecraft Teachers Google group.³

Twitter

There are loads of teachers who tweet out regularly on the hashtag #MinecraftEdu. If you're looking for some quick advice or help, start there. My handle is @MrJasonWilmot.

Use Google Hangout

You can screen share, chat, and collaborate—and with so many built-in tools, you'll be amazed that it comes absolutely free.

Last Random Tips

YouTube Is Your New Best Friend

2. <http://minecrafteu.com/community>

3. <https://groups.google.com/forum/#!forum/minecraft-teachers>

YouTube is your absolute best resource for all things *Minecraft*. There are tutorials for building, for updating, for troubleshooting, and so forth. In fact, it is a great idea to have students start creating and producing their own videos and uploading them to the ever-growing community.

Learn the Controls

Learning the controls is one of the most fundamental aspects of *Minecraft*. It enables quicker play and gives a sense of mastery over the game. If you're right-handed, keep the movement buttons on W, A, S, and D—you will want to have left-hand control over the keyboard because the right hand is busy on the mouse.

Tip: You can grab a PDF of the controls from MinecraftEdu or check out my QR code (see Figure 8), as I have a copy there as well.

Do I Really Need a Three-Button Mouse?

Yes. Well, no, but yes. Why try to ride a three-legged horse just to prove you can do it? If you can't afford to buy any, ask around for people to donate them. A three-button mouse is absolutely worth it.

Learn the Lingo

Trolling. Griefing. Mobs. Mooshoom. A whole new lexicon awaits you, noob.

Your First In-Game Enemy: TNT

If you're using *MinecraftEdu*, you are in luck because you can disable TNT. On your *MinecraftEdu* server, select "World Settings" to discover your ability to enable or disable TNT. Lava and fire are still active but will not spread.

Water, Water Everywhere

Having problems with kids pouring water everywhere? Me too. Good luck with that.

Slow Gameplay?

Go to your video settings and turn "Fancy" to "Fast." This should speed up gameplay. There are a lot more things you can do to speed up your gameplay, so for an exhaustive list I would suggest the World Wide Web.

Screen Shots!

Teach your kids how to take screen shots. You can save them, print them, and send them home. You can also teach kids to upload their screen shots to Edmodo, Google Classroom, or online communities such as Creatubbles in order to share.⁴

What Age Range Can I Use With Minecraft?

4. <https://www.creatubbles.com>

I've used *Minecraft* with Grades K-5 with absolutely no trouble. You will obviously need to tailor the lessons or the learning objectives to the appropriate levels, but in my experience "too young" has never been a problem teaching with *Minecraft*.

Can I 3D-Print My Minecraft Creations?

Yes. In the age of 3D printers, even your *Minecraft* creations can become tangible objects. If you're interested in printing your projects, plenty of websites offer their services.⁵

5. <http://www.printcraft.org> and <http://www.minecraftprint.com>