



Above: Example building content by vrbn studios, for games, VFX and virtual production

Below: LA in full 3D, based on GIS data





INDUSTRY INSIGHT

Procedural Cities series #3

# Strategies for procedural modelling of 3D cities

Matthias Buehler, founder of 3D urban environment specialists vrbn solutions, breaks down the creation process and strategies for building believable cities

**I**n this third and final article in our series, we will tackle procedural cities on a more global and strategic level. We'll discuss what elements make up a believable city, and how its creation can be broken down into more granular levels of procedural control – but the most important aspect is not losing control during the planning and implementation phases.

Strategy is half the success. Learning by doing is a good strategy when undertaking tasks like learning how to ride a bike. Tackling other challenges like how to plan and create a procedural 3D city needs a different approach though, as there is a lot of implicit knowledge required. Trying to automate certain tasks needs an excellent understanding of the underlying craftsmanship and processes. Let's dive in.

## 1. ELEMENTS OF A CITY

“What does a city consist of?” I guess we all know most of the typical urban elements like water bodies, parks, streets, and buildings. But we need to dig deeper. So let me rephrase: “What are the elements and processes that make up the nature and identity of a city?” The answer to this fundamental question is always derived from the story to be told.

So, what are these elements?

- Physics (gravity)
- The inhabitant's control of energy

and technology (see, for example, the ‘Kardashev scale’)

- Natural environment (geography and geology of the planet – unless your city is on a space station)
- Weather and atmosphere
- Artificial structures and urban layout
- Politics and social climate

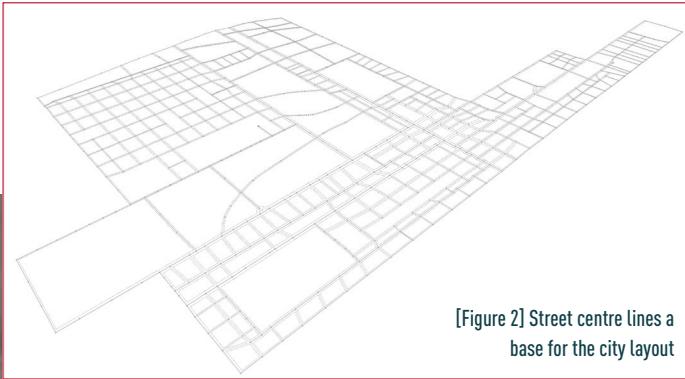
All of these details need to be taken into account when planning your project.

One of my favourite examples of coherent visual storytelling in the creation of a city is the dystopian ‘Mega-City One’ in the Hollywood sci-fi film *Dredd*, where Mega-Blocks (almost vertical cities with thousands of residents) cast large parts of their surrounding urban fabric into its endless shadow. This creates an extremely inhospitable environment for its inhabitants [see Figure 1].

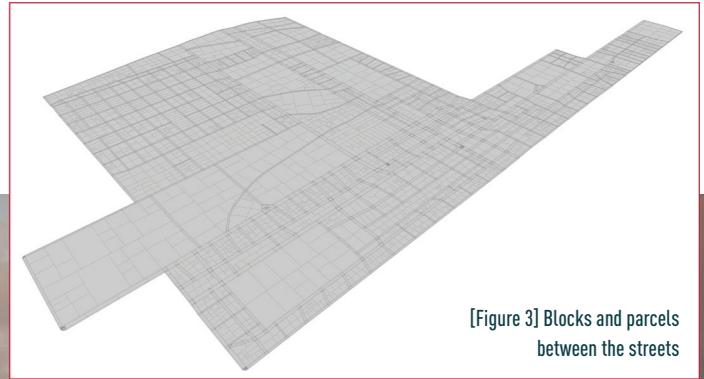
“CONSIDER THE ELEMENTS AND PROCESSES THAT MAKE UP THE IDENTITY OF A CITY”



[Figure 1] Mega-City One art by VFX art director Neil Miller for *Dredd* © DNA Films



[Figure 2] Street centre lines a base for the city layout



[Figure 3] Blocks and parcels between the streets



A fully procedural 3D favela

## IMPLICATIONS OF AUTOMATION AS A CONCEPT

Automation by a computer implies the process is fully informed about the whole scope of decisions. In the case of procedural modelling of architecture, it is the same. Think about it: you're trying to develop a system that creates a city in less time (as time is money) than it would take to design and model every single building. Therefore it then becomes clear that this requires the introduction of some shortcuts to reaching a balance or a compromise. And those compromises – that's the art – have to be decided for each project and cannot be automated.

## 2. GRANULARITY OF PROCEDURALISM

Every project you tackle will probably have different needs and a different budget. Therefore, the strategy will probably need to be fine-tuned individually. But what are the 'global' strategies you can leverage when diving into a procedural 3D city project?

Over the last few years, I have isolated these five typical high-level approaches:

- a) Procedural layout + fully procedural buildings
- b) Procedural layout + procedural buildings based on building part assets
- c) Procedural layout + asset buildings
- d) Procedural buildings on GIS footprints
- e) Instance asset buildings on GIS footprints

Each approach has its distinct pros and cons and may or may not apply to a specific project. But how far do you go? Do you even want variation in every plant in your city? For example, Weta Digital (*Avatar*,

*Lord Of The Rings*) has systems in place that simulate biomes, where every plant grows to a unique form, based on, for example, its access to light. A similar technical approach was chosen by simulating wind that distributes plant seeds for the 'Kite Demo' for Unreal Engine. The sky is the limit – so it's your choice. Let's have a look at two example approaches:

### Example for Type a): Fully procedural

Going fully procedural is everybody's dream, but of course, the hardest to implement, as it creates the biggest amount of variation. It's typical that first, e.g. in CityEngine or Houdini, a street network is drawn (or generated), which is then converted to streets and blocks. Each of the different types of resulting polygons is then processed into a unique 3D geometry [see Figures 2 & 3].

Yet even if deciding to take this route, you should ask yourself critically: does my



Above: [Figure 4] GIS data – 2D building footprints  
Below: [Figure 5] Instanced 3D building assets on 2D GIS data



project really benefit from the fact that my entire city is fully procedural down to the last building at the horizon, where a building is about three pixels high, and you may never even get close to it? Think about the data volume this creates per city iteration (can easily be gigabytes in VFX scenarios), which you need to load and store, push through your network, versionize, and backup.

#### Example for Type e): Instancing on GIS footprints

A completely different approach, but for certain applications a very efficient method, is leveraging GIS data and simply instancing building assets on top. It's a low-hanging fruit which helps out in a lot of scenarios, and without being too complicated [see Figures 4 & 5]. You will likely find that it's not easy to choose the right granularity that is ideal for your project and budget, but when is 'easy' ever an option in life?

## “A VERY EFFICIENT METHOD IS LEVERAGING GIS DATA AND SIMPLY INSTANCING BUILDING ASSETS ON TOP”

### 3. STRATEGIES: 12 BEST PRACTICES

Here are my 12 tips for success with procedural systems:

1. If possible, budget the development and implementation over multiple projects.
2. Research on all fronts: isolate (artistic/technical) bottlenecks and pitfalls early on.
3. Implement the pipeline first, using proxy content.
4. Try to aim for the absolute minimal procedural solution first, only then add the artistic content.
5. Make sure early on the pipeline scales (especially geometry and material transfer).
6. Build up trust in the process within your team. You will only see the first good results once you have 90 per cent of the system implemented.
7. Define clear points for design processes, data hand-off (file I/O) within the team.
8. Define a robust naming convention early on. Everything depends on it.
9. Document all technical steps well, so that every artist and manager knows the nitty-gritty.
10. Be sure to allocate enough time for any asset and texture fixes. Sometimes errors just happen!

- 11. Implement enough quality control steps and mechanisms.
- 12. Provide enough snacks to the team!

Some of those points may sound a bit generic, but I cannot stress enough how important they are in the context of procedural modelling! Trust me – I have learned them all the hard way!

#### 4. PRACTICAL EXERCISE: STATISTICAL ANALYSIS

Topic: Training the eye and statistical analysis of reference images.

Let's assume our task is to create a procedural environment, inspired by today's Tokyo, based on the urban layout, density, building types, and colour palette. Now, as we don't need to replicate the exact city, we need to extract the city's nature – the broad strokes that mimic the typical style. In procedural modelling, we always want to go for those broad strokes and 'general rules'.

**Step 1:** Collect multiple high-res reference images of Tokyo: aerial views, photos from skyscrapers and towers, but also from the pedestrian level.

**Step 2:** Start a table with element categories: classes of streets, parks, open spaces, water bodies. Then small and large residential buildings, midrises, and skyscrapers. Further to this, also note down which administrative, religious, industrial, and infrastructure (e.g. parking) related buildings you find. For each type, count how many there are and how often they appear.

**Step 3:** Which elements are unique? If they are, these are landmarks. Would you say they are essential for the look and feel of 'Tokyo'? If yes, note them down.

**Step 4:** Start a collage of images of the common building and open space types and sort them based on colours and structure. As you can see in Figure 6 below I have sorted the buildings by white, beige, brown, and other colours.

**Step 5:** Research which building types have common façade patterns, and note down the statistics.

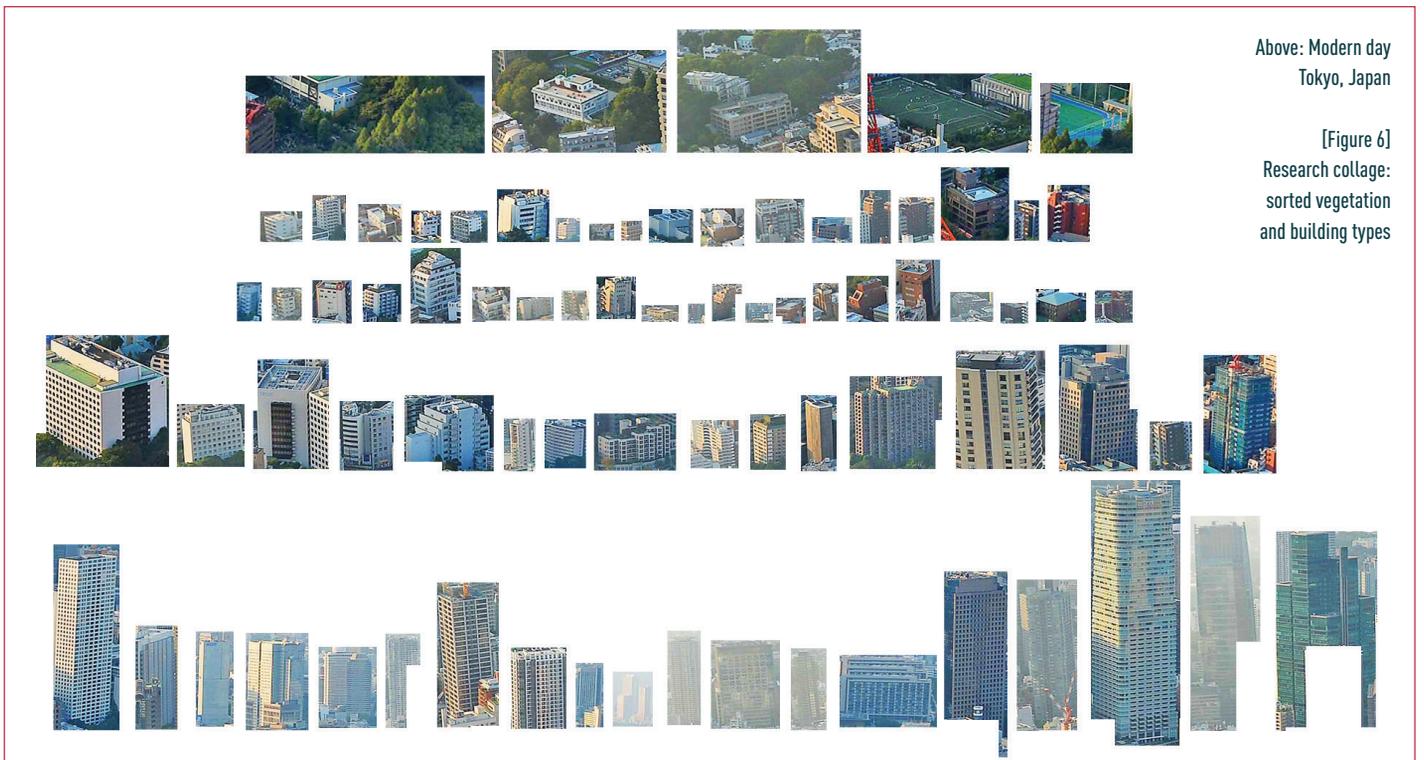
**Step 6:** Research and note down the roof forms, construction types, roof furniture types (AC units, antennae, etc). This process should, in a real project scenario, take at least 20 hours. The further you dive into the images and statistics, the clearer the 'system' becomes how

## “RESEARCH WHICH BUILDING TYPES HAVE COMMON FAÇADE PATTERNS”



Above: Modern day Tokyo, Japan

[Figure 6] Research collage: sorted vegetation and building types





Going procedural – or not?  
Buildings raytraced in Unreal Engine

the structure of a city works. This exact process is one of the keys to the success of a convincing procedural project. Personally, this is one of the most fulfilling project parts, as I can fully immerse myself in the art of analysing patterns and trying to find the best possible approach to tackle the entire puzzle.

## 5. CONCLUSION

As with everything in life, planning and strategy helps you to reach your goals faster. And with procedural cities, there are lots and lots of decisions to be made. This article touched on a few of the important aspects of the broad approach to tackling a 3D city based on a few examples. In your project you will, of course, be facing other challenges. So it will help to take the general recipe of the strategies described and instrumentalise the same problem-solving thought process. With time, you will refine your senses for it, and it remains a very satisfying learning process – even for me, who has been in this field for over ten years!

## 6. FINAL THOUGHTS

As we have seen during these three articles, creating a convincing procedural environment needs more than just mastering procedural systems' technicalities. We always want to be guided by the story to be told and by the art direction – yet there is a strong need to keep track of production and management efforts. The evaluation of which step shall be implemented must be discussed in detail with all stakeholders, as some decisions can lead to hundreds of hours of work.

When investing in procedural systems, we can create a ton of content, yet the levers, knobs, and dials that influence the system are hard and risky to manage: with great power comes great responsibility.

I hope you have gained a basic understanding of the fascinating field of procedural modelling of 3D cities through this three-part series.

**FYI** For any questions, please feel free to contact the author via [info@vrbn.io](mailto:info@vrbn.io)

## APPROACH AT VRBN SOLUTIONS

Internally, we don't have one specific approach to how we tackle procedural cities, as we don't have to just build one specific for a project or game. Though as we're consultants in this field, we know most of the pitfalls and decisions that are costly.

Developing a procedural system as a sentence sounds magically easy. But the more realistic and photoreal your goals are, the exponentially more complex things become. The library, the pipeline integration, the consistency and realism of the buildings, and the urban layout can quickly sum up to costs of 200k to 500k – not a budget every studio has, and why our second business unit, vrbn studios, focuses on high-quality off-the-shelf building assets.