

Professional Tutorial

Production workflows: multi-district cities, hero shots, night lighting, real places from OpenStreetMap, and keeping heavy scenes fast.

Who this is for

You have already built a city from a preset. This guide is about **controlling** the result — building a city to a brief, lighting it, and getting it to render without a scene that fights you.

1 Build a city to a brief, not to a preset

A preset is one designer's answer. Real work starts from **districts**: each is a centre of gravity with its own grain and skyline. A believable city is usually three or four.

- 1 **Start from the closest preset**, then open **Districts** and work on the list rather than starting from nothing.
- 2 **Give each district a job**. A downtown is a small **Radius**, small **Block Size**, high **Max Floors**. A suburb is a large Radius, large Block Size, **Max Floors 2–3**. An old quarter is a small Block Size with high **Irregularity**.
- 3 **Rotate them against each other**. Set a different **Rotation** per district (say 0°, 18°, –35°). Where two grids meet at an angle you get the wedge-shaped blocks that make a city look grown rather than generated — this single control does more for realism than any other.
- 4 **Use Strength to decide who wins** where districts overlap. The higher Strength imposes its grid.
- 5 **Set Downtown Falloff** in Buildings so height drops away from each centre instead of stopping at a hard edge.

Technique — the wedge test. If your city reads as "procedural", it is almost always because every street meets at 90°. Rotate one district 20° and rebuild. The irregular junctions it creates are the difference between a grid and a city.

Obstacles are a design tool

An **Obstacle** is a region streets bend around and stop at — a lake, a park, a rail yard. Add one and the network reorganises around it, which produces waterfronts, ring roads and dead-end lanes you would never place by hand. **Influence** controls how far away streets start to notice it.

2 Terrain and the city that sits on it

Terrain changes everything downstream: buildings sit on the ground, streets follow it, and bridges appear where the ground falls away.

CONTROL	USE
Ground Shape 0–5	0 flat · 1 hills · 2 slope · 3 ridge · 4 basin · 5 coast. The fastest way to real ground with no map at all.
Shape Size (m)	The wavelength of the ground. Large values are usually right — make it smaller than a block and the ground ripples underneath individual houses.
Terrain Height (m)	Total relief. San Francisco character comes from height with a large Shape Size, not from a small bumpy one.
Building Sit %	Where a building meets sloped ground: 0 lowest corner, 50 average, 100 highest. 50 is right almost always.
Terrain Detail	Grid resolution. Raise it only if roads look faceted — it is the most expensive control in the section.

Buildings on a slope are given a **plinth** automatically: the walls reach down to the lowest ground under the footprint so nothing hovers, while the floors and window rows stay level. You do not need to do anything for this.

3 Night: the highest-value hour of work

Night is where this plugin pulls away from a grey massing model. The lighting is per-window, not a texture, so it holds up close and at distance.

- 1 **Windows on**, then **Night** on.
- 2 **Lit Windows %** — start at **18**. A real city is mostly dark; 50% reads as a stadium.
- 3 **Light Colour Spread %** and **Light Dim Spread %** — these stop every lit pane being identical. Keep them high (100 / 65 are good) so the facade has warm and cool, bright and dim.
- 4 **Commercial Lit Boost %** — offices burn brighter and later than flats. This is what separates a business district from housing at night.
- 5 **Shopfronts** — set ground floors to always lit for a busy street.
- 6 **Light Zones** — divide the city into zones and black one out with **Dark Zone** for a blackout or a district under construction.
- 7 **Rooftops** → **Aviation Red Light** — the red obstruction lights on tall roofs. Small detail, large effect: it is what makes a skyline read as a real one.
- 8 **Street Lamps Lit** in Street Furniture, then rebuild the material.

9 Create Street Lights — and this is the step that actually lights the road. **Street Lamps Lit** and **Lamp Brightness %** make the lantern *glow*; an emissive surface shines without illuminating anything, so the tarmac under it does not change however far you push the slider. This button puts a real Max light under every lantern. It tells you how many it is about to create and asks first — if the number is large, set **Prop Radius** and press **Generate** before accepting. They are named **VGC_StreetLight** so you can select them together and set intensity and colour; **Delete Street Lights** removes them.

Order matters. Set night up *first*, then press **Build City Material**. The material is built from the state the city is in, so building it before switching night on leaves the glowing surfaces unconfigured.

4 Real places: OpenStreetMap (1-Year plan)

OSM import builds the city from the real street network of a real place.

- 1 Copy Overpass Query** in the Layout section puts a ready-made query on your clipboard.
- 2** Go to **overpass-turbo.eu**, paste it, move the map to the area you want, and **Run**.
- 3 Export** → **download as raw OSM data**. You get a **.osm** file.
- 4** Back in Max, press **Import OSM File...** and choose it.
- 5 OSM Mode** — **1** brings the streets and lets the plugin subdivide and build (recommended: street data is excellent worldwide, building data is patchy). **0** uses the real building footprints where they exist.

Once imported, the city **stays** imported: street lights, poles, cables and trees populate along the real roads, and moving a slider edits that city rather than replacing it. Picking a preset or **Reset to Defaults** is how you deliberately go back to a generated city.

Practical limits

Keep the extracted area modest — a few square kilometres of dense city is a very large mesh. Use **OSM Radius** to clip a big file down rather than importing everything and hiding it.

5 Bridges

Bridges are **derived, never scattered**. The generator cuts streets where they run into water, and a bridge is the mend — so bridges appear where a crossing genuinely belongs.

CONTROL	USE
Min Span (m)	The "fewer, bigger bridges" control. Raise it to keep only the major crossings.
Min Region Radius (m)	Refuses to bridge a small region. Raise it if a small park or plaza is being treated like a river.

CONTROL	USE
Type -1 / 0 / 1 / 2	-1 lets the span decide. 0 beam, 1 arch, 2 cable-stayed.
Arch / Cable Bias %	Shifts the automatic choice. Lower Cable Bias if you want piers rather than towers — a cable-stayed bridge has no mid-span columns by design.
Deck Lamps	Street lights along both deck edges, facing inward over the road.

Because the deck is a *height the road has* rather than a separate object, road markings, kerbs and lamps continue across a bridge with no extra setup.

6 Performance and budgets

Every heavy feature has a triangle budget, and each one is spent **evenly across the whole city** — so a bound budget thins detail everywhere instead of building one perfect corner and leaving the rest bare.

BUDGET	WHEN TO RAISE IT
Window Budget	The status line says the share it reached. Raise for hero shots; the default covers most cities completely.
Roof Budget / Max Roof Items	Raise for close rooftop work.
Marking Budget	Raise if lane lines disappear on a very large city.
Max Props / Prop Model Budget	Raise for dense street furniture; this is the one most likely to bind on a big city.
Max Buildings	A ceiling spent by striding, so lowering it thins the city evenly.
Cable Budget	The wiring spends <i>fidelity</i> before coverage — fewer conductors, then flat ribbons, then a coarser sag — so a bound budget never leaves a gap between two standing poles. Raise it for close work on the wires.
Interior Budget	Floors and furniture behind the glass. This one keeps every floor and pays out of the furniture, because the floors are what stop a window reading as painted on. Use Interior Radius rather than raising it on a big city.
Traffic Moves	Not a budget, but the same decision. With it on the city mesh becomes time-varying and the vehicles are re-patched every frame; the static city is cached and is not rebuilt. Leave it off while you are laying the city out, and turn it on to preview or render the animation.

Radius controls for working fast

Window Radius, **Roof Radius** and **Prop Radius** limit expensive detail to an area. With **Radius Follows View** the prop radius tracks your viewport camera — detail where you are looking, nothing where you are not. Turn these off before a final render.

Working method. Block the city out with windows, rooftops and markings *off*. Get the layout, the districts and the terrain right first — they are cheap to iterate. Turn the detail on last.

7 Materials and your own models

Build City Material creates a Multi/Sub-Object with a correct surface in every slot — road, pavement, kerb, wall, roof, glass, lit glass, bridge deck, pier, rail, sign face, sign frame, roof mast, dish, beacon, terrain. Every slot is a normal Max material: replace any of them with your own.

For scattered objects, pick your own model per kind, set its **Size**, then **Scatter Instances**. They are created as true Max instances, so one edit to the source updates every copy in the city.

8 When something is wrong, ask the plugin

Save Debug Report... writes a text file containing every setting, what the generator produced from them, which budgets bound, and a list of health checks that each either pass or name their complaint. It carries no geometry and no file paths. Read it yourself first — it usually states the cause plainly — and send it to support if not.
