

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
INFORMATIKAI KAR
TÉRKÉPTUDOMÁNYI ÉS GEOINFORMATIKAI TANSZÉK

HÁROMDIMENZIÓS MODELLEK A VÁROSFEJLESZTÉSBEN

~~~

Esettanulmány a pécsi Zsolnay Kulturális Negyed példáján keresztül  
ESRI ArcGIS környezetben

Készítette

**Mesterházy Anna Dóra**

térképész mesterszakos hallgató

Témavezető

**dr. Gede Mátyás**

adjunktus



Budapest, 2014

## TARTALOMJEGYZÉK

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| FOGALOMMAGYARÁZAT.....                                                     | 3  |
| BEVEZETÉS.....                                                             | 4  |
| 1. ZSOLNAY-GYÁRBÓL ZSOLNAY KULTURÁLIS NEGYED.....                          | 6  |
| 1.1. A Zsolnay-gyár jelentősége.....                                       | 6  |
| 1.2. A Zsolnay hagyomány.....                                              | 7  |
| 2. AZ EURÓPA KULTURÁLIS FŐVÁROSA CÍM.....                                  | 10 |
| 3. AZ ARCGIS TECHNOLOGIA.....                                              | 12 |
| 3.1. 3D az ArcGIS-ben.....                                                 | 13 |
| 3.1.1. ArcGIS Pro.....                                                     | 13 |
| 3.1.2. ArcGlobe.....                                                       | 14 |
| 3.1.3. ArcScene.....                                                       | 14 |
| 4. A CITY ENGINE.....                                                      | 16 |
| 4.1. A felhasználói felület.....                                           | 18 |
| 4.1.1. Navigator.....                                                      | 18 |
| 4.1.2. Előnézet.....                                                       | 19 |
| 4.1.3. Réteg- és szabályfájlkezelő.....                                    | 20 |
| 4.1.4. Viewport.....                                                       | 21 |
| 4.1.5. Inspector.....                                                      | 22 |
| 5. MÓDSZERTAN.....                                                         | 23 |
| 5.1. Az alapanyagok előkészítése.....                                      | 23 |
| 5.1.1. Az épületalaprajzok.....                                            | 24 |
| 5.1.2. Az ortofotó.....                                                    | 25 |
| 5.1.3. Magasságadatok.....                                                 | 26 |
| 5.1.4. Textúrák.....                                                       | 28 |
| 5.2 Szabályfájlok.....                                                     | 29 |
| 5.2.1 Nevezéktan és tagolás.....                                           | 29 |
| 5.2.2 Szabályfájlok jellegzetes elemei.....                                | 31 |
| 6. ÉPÜLETEKKEL KAPCSOLATOS ELEMZÉSEK ARCSCEINE-BEN ÉS CITY ENGINE-BEN..... | 34 |
| 6.1 Napbesugárzási vizsgálat.....                                          | 35 |
| 6.2 Láthatósági vizsgálat.....                                             | 38 |
| 6.3 Beépítési szabályok ábrázolása.....                                    | 40 |
| ÖSSZEFOGLALÁS.....                                                         | 41 |
| MELLÉKLETEK.....                                                           | 46 |

## Fogalommagyarázat

|                           |                                                                                                                                                                                                                                                                                                                   |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| GIS                       | Geographic Information Systems (Földrajzi Információs Rendszerek). Olyan számítógépes rendszer, amely térbeli adatok tárolására, elemzésére, kezelésére, megjelenítésére szolgál. Integrálja a térbeli és leíró információkat.                                                                                    |
| ARCGIS                    | Az ESRI által gyártott térinformatikai megoldások általános gyűjtőneve.                                                                                                                                                                                                                                           |
| ARCGIS PRO                | 2014 decemberében a 10.3-as ArcGIS for Desktoptal egyszerre piacra került alkalmazás. Új felhasználói felületen olvasztja magába az ArcMap, az ArcScene és az ArcGlobe funkcionalitásait.                                                                                                                         |
| CITY ENGINE               | Az épített környezet modellezésére kifejlesztett procedurális alapokon működő program, az ArcGIS szoftvercsalád tagja.                                                                                                                                                                                            |
| CGA                       | (Computer Generated Architecture). Kifejezetten a City Engine-hez létrehozott programozási nyelv, háromdimenziós épületek megjelenítésére. A modellezés során a készítő a scriptben definiálja az épületek tulajdonságait, a 2D-s kiinduló alakzat egyre több részlet hozzáadásával válik valósághű 3D-s modellé. |
| FEATURE CLASS             | Térképelem-osztály. Geometriájuk típusa szerint azonos térképi elemek gyűjtőhelye a geodatbázison belül.                                                                                                                                                                                                          |
| FEATURE DATASET           | Azonos koordinátarendszerű feature classok csoportja egy geodatbázison belül.                                                                                                                                                                                                                                     |
| GUI                       | (Graphic User Interface). Szoftverek felhasználói felülete.                                                                                                                                                                                                                                                       |
| GEOADATBÁZIS              | (.gdb). Az ArcGIS elsődleges adattároló formátuma. Vektoros, raszteres és leíró adatok tárolására                                                                                                                                                                                                                 |
| MULTIPATCH                | Háromdimenziós GIS objektum, annak határait és minden tulajdonságát (pl. szín, textúra) az adatbázisban egy rekord tárolja.                                                                                                                                                                                       |
| SHAPE                     | A City Engine kiinduló kétdimenziós alakzatai, amelyekből a CGA-nyelv használatával 3D-s modell szerkeszthető.                                                                                                                                                                                                    |
| WEBES<br>TÉRKÉPALKALMAZÁS | Szélesebb közönség számára elérhetővé tett, saját URL-lel rendelkező alkalmazás. Általában valamilyen tematikus tartalmat hivatott megjeleníteni.                                                                                                                                                                 |

## **Bevezetés**

Háromdimenziós városmodellek az épített környezettel foglalkozó számos szakterületen elterjedtek: az építészet, várostervezés és környezetvédelem mellett az ingatlanértékesítésben is egyre nagyobb szerepe van ennek a megjelenítési módnak. A harmadik dimenzió szemléltetésével érzékelhetővé válnak a terek, magasságok, tömegviszonyok, jobban megérthetővé válik a földrajzi tér. Segítségével nemcsak egy pusztán látványosabb végeredményt kapunk egy kétdimenziós térképhez képest: az esztétikai megfontolásokon túl egy sokkal felhasználóbarátabb, információgazdagabb és az adatvizualizáció magasabb fokán álló végeredményt tarthat majd “kezében” a felhasználó, amely az alkalmazott szoftverek függvényében elemző funkciókkal is bírhat.

A fejlett, elemzőképessegekkel is felruházott háromdimenziós térképek használata a városfejlesztésben külföldön igen elterjedt gyakorlat, a fejlesztési döntések állandó támasza, az önkormányzati térinformatika napi szinten alkalmazott módszere, hazánkban azonban érdekes látványterveket és néhány úttörő céget leszámítva kevésbé terjedt még el ez a gyakorlatias megközelítés.

Dolgozatom témájául ezt a problématerületet választottam, célom bemutatni azt, hogyan képes egy 3D-s térkép a várostervezésben és –fejlesztésben segítségére lenni a döntéshozóknak, akár egy új épület vagy fejlesztési terület elképzelésének kapcsán vagy annak utólagos hatáselemzésében. Miben ad többet egy ilyen modell a hagyományos térképekhez képest? Milyen módszerek állnak rendelkezésre egy ilyen multifunkciós térkép elkészítéséhez? Milyen elemzések készíthetők és ezek mennyire lehetnek hatással egy terv megvalósulására, értékelésére?

Mintaterületül Pécs-et választottam, ahol 2010 előtt az Európa Kulturális Fővárosa cím kapcsán mind városfejlesztési, kulturális és politikai értelemben nagy port kavaró építkezések indultak meg. A Zsolnay Kulturális Negyed, az EKF öt kulcsprojektjének egyike Közép-Európában az elmúlt évek legnagyobb iparterületet célzó építkezése volt. A csaknem 70 ezer négyzetméteres beruházás a huszadik században virágkorát élő Zsolnay-gyár területét öleli fel, amely napjainkra kiüresedett, a gyár egykori épületei omladozva

álltak. Sürgetővé vált ennek a városon belüli zárványnak megnyitása, újragondolása. A pályázat lecsengésével, az idő szűke miatt sebtében felépített és restaurált épületek utóélete a mai napig vita tárgya. Az üzemeltetési problémák, az egyetemi hallgatók létszámának csökkenése, a várt látogatók számának elmaradása mind felveti a kérdést: Mennyire sikeres a negyed hosszú távon? Hogyan tud az új egység beépülni a városszövetbe és az emberek mindennapjaiba? Ezekre a kérdésekre adhat választ egy háromdimenziós városmodell elkészítése.

Térképészeti munkáról lévén szó, nem célja a dolgozatnak Pécs városföldrajzát, a Zsolnayak történelmi fejlődését vagy a ZsKN megépülése óta eltelt időszak eredményeit tüzetesen átvizsgálni vagy urbanisztikai kérdésekbe bocsátkozni, sokkal inkább módszertani szemléletű írásról van szó, amely a szoftveroldalon talált lehetőségeket próbálja a gyakorlatban kipróbálni, a szemléletét alkalmazni, hogy kiderüljön, milyen befolyással lehetnek ezek a háromdimenziós térképek és azok eszközei a városfejlesztésben.

Szoftverkörnyezetként az ArcGIS családot választottam, amely könnyen átjárható infrastruktúraként a térképkészítés minden egyes fázisát lefedi, a formátumok között az átjárás is biztosított. A külön háromdimenziós városok készítésére kifejlesztett programjával, a City Engine-nel, a háromdimenziós adatok feldolgozására alkalmas 3D Analyst extension-nel és a webes megjelenítést segítő City Engine webviewerrel ideális környezetet biztosított a mintaterület megjelenítéséhez és a dolgozat létrejöttéhez.

## **1. Zsolnay-gyárból Zsolnay Kulturális Negyed**

### **1.1 A Zsolnay-gyár jelentősége**

Dolgozatom tématerületének kiválasztásakor elsődleges szempont volt, hogy hazánk vagy valamely nagyvárosunk életében fontos szerepet betöltő, városszerkezeti vagy várostörténeti szempontból jelentős beruházást járjak körül, amely önmaga és tágabb környezetére is hatással van és háromdimenziós modelljének megalkotása nagyobb területen értelmezhető. Szempont volt az is, hogy már megvalósult fejlesztést modellezzek le, hiszen egyrészt egy éppen csak körvonalazódó terv sok esetben képlékeny lehet, a dolgozat elkészültekor könnyen előfordulhat, hogy mégis lesöprik az asztalról, egészen más formában kivitelezik, vagy máshol valósítják meg. Másrészt a kézzelfogható új vagy restaurált épületek, terek utóélete, beilleszkedése a város keringésébe kiemelt kérdés a városfejlesztők számára, így esett a választásom a pécsi Zsolnay-gyár rekonstrukciójára, amelynek során a hagyományok megőrzése és a modern építészeti elemek együtt élnek, kiegészítik egymást.

Az Európa Kulturális Fővárosa címet 2010-ben Isztambullal és Essennel osztozva Pécs kapta meg, amelynek keretében a kulturális programokon kívül több fejlesztési területet jelöltek ki a városban. Hangversenyterem és tudásközpont felépítésén kívül a legösszetettebb feladatot a Zsolnay-gyár jelentette. A több száz éves múlttal bíró, a maga idejében az innovációt, tudást és a magyar ipar fellegvárát megtestesítő gyáron és környezetén meghagyta nyomát az 1929-es világgazdasági válság és a szocializmus. A virágzó, több száz munkást foglalkoztató gyár helyén az ezredfordulón üres és elhanyagolt épületek álltak, a gyártás a terület keleti részére húzódott vissza. A többtíz hektáros gyártelek számos épületegyüttest foglal magába, amelyek között sok műemléki védeltséget is élvez.

A Zsolnay-gyár területe afféle belső zárványként, egy letűnt kor szimbólumaként él már évtizedek óta a pécsiek tudatában, ennélfogva revitalizálása nemcsak a terület

nagysága miatt volt komplex építészeti és urbanisztikai feladat, hanem annak a kulturális és ipari hagyománynak is, amelyet a Zsolnay család kerámiái ma is képviselnek.

## 1.2 A Zsolnay hagyomány

A kezdetben kőedény- és téglagyártással foglalkozó kis manufaktúrát Zsolnay Ignác vezette, azonban a fejlettebb osztrák ipar miatt nem volt nyereséges, hamar válságba került. Ignác feladta a gyáregység vezetését, amelyet fivére, Vilmos vett át, aki szorgalmával és tehetségével sikerre vitte a vállalkozást, 1868-ban Pécsi Cement, Chamott és Tűzálló anyagáruk néven jegyeztette be. Az addigi termékpalettához újakat illesztett, folyamatos kísérletezgetéseinek hála pedig egyre javult az előállított kerámiák minősége. (A ZSOLNAY-HAGYOMÁNY).

Az iparosodásban bízva Vilmos kezdetben nagy hangsúlyt fektetett az edénygyártás mellett a cementgyártásra is, az azonban nem öltött akkora méreteket, hogy a gyártás gazdaságos lett volna, ezért a cement és téglakészítés hamarosan leállt. Az edénygyártás terén viszont gyors előrelépést sikerült elérnie, a legjobb alapanyagok felkutatása és a mázakkal való kísérletezések 1873-ban értek be, amikor a Bécsi Világkiállításon nemzetközi elismerésben részesült a mester. Az elismerés beindította az üzemet, egyre több megrendelés érkezett, amely tovább serkentette az egész család alkotókedvét, hiszen Vilmos mellett felesége és három gyermeke, Teréz, Júlia és Miklós is nagyban kivették részüket a terméktervezésből, üzletvezetésből (ZSOLNAY 1975).

A Párizsi Világkiállítás fődíját 1878-ban nyerték el addig sosem látott anyagokból készült díszes agyagáru termékeikkel, amely újabb löketet adott az üzletnek. Az 1880-as évek egyedülálló találmánya a pirogránit, amely fagy-, sav- és tűzállóságánál fogva a kor kedvelt építőanyaga lett, színes mázzal borított változatait csodálhatjuk mai is az Iparművészeti Múzeum, a Mátyás-templom vagy a Magyar Földtani Intézet tetőszerkezetén (1. ábra). A pirogránit mellett a másik Zsolnay-vívmány az 1885-ben kifejlesztett híres eozin-technika. Az égetés során fémes kiválás képződik az edényen, ettől válik a máz különlegesen fémes fényűvé (2. ábra). A díszes kerámiák gyártása mellett Vilmos kályha- és villamoshálózatokhoz használt porcelánszigetelő-gyártásával is bővítette az üzemet (A ZSOLNAY-HAGYOMÁNY).



1. ábra Pirogránit elemek a Mátyás-templom tetején



2. ábra Eozin-mázás bikafej, az Európa Kulturális Fővárosa program pécsi jelképe

Zsolnay Vilmos halála után fia, Miklós vitte tovább az üzletet és kamatoztatta sikerrel apja találmányait, az 1929-es világválság azonban megtörte a korábban töretlen lendületet. A keresletet követve a gyár az olcsó tömegtermékek felé fordult, a termelés csökkent, majd a második világháborúban teljesen leállt és az üzemet államosították. Ezt követően a diktatúra éveiben a családot megfosztották otthonuktól, a gyárban pedig megszűnt a termelés. A hullámvölgyet 1953-ban látszott először kiheverni a gyár, ekkor indult újra a kályha- és edénygyártás, önállóságát azonban 1963-ban ismét elveszítette és az Észak Magyarországi Finomkerámia-ipari Országos Vállalat Pécsi Porcelángyára néven működött tovább. A rendszerváltás éveiben végül privatizálták, a Magyar Befektetési és Fejlesztési Bank kezébe került, 1999-ben három részre, a Zsolnay Porcelángyár Rt<sup>1</sup>-re, a Zsolnay Porcelánmanufaktúra Rt<sup>2</sup>-re és Zsolnay Örökségkezelő Kht<sup>3</sup>-re vált szét (GYÁRTÖRTÉNET).

A kétezres években tulajdonosa Pécs Önkormányzata lett, a kezelési feladatokhoz azonban nem állt rendelkezésre elég erőforrás, ezért 2013-ban külföldi befektető bevonása mellett döntött a városvezetés: Bachar Najari szír származású svájci-magyar üzletember 74,5%-os tulajdonrészt szerzett a gyárból, 180 millió forinton kívül 500 millió forint tőkejuttatást is nyújt az üzemnek (HVG 2013).

<sup>1</sup> épületek bérbeadásával, alapanyagokkal és energiaellátással foglalkozik

<sup>2</sup> a termékek gyártásáért felel

<sup>3</sup> a gyár területén lévő műemléki ingatlanok kezelését végzi



A Zsolnay-gyár bonyolult tulajdonviszonyai vezetnek el a 2010-es évek eseményekig. A tulajdonosváltások során identitását egyre jobban elveszítő műhely terei egyre inkább átalakultak. Az épületamortizáció mellett számos hozzátoldást, kiegészítő épületet húztak még fel, amelyek egyre inkább sűrűbbé és átláthatatlanná tették a terület szerkezetét. A ZsKN létrehozása kétségtelenül vonzóbbá tette a városrész megjelenését, a fölösleges toldások lebontásával, letisztított új formákkal és az eredeti homlokzatok megtartásával visszaállította a helyszínen a hely szellemét és új funkciókat adott a tereknek. A 2010-es EKF kacsaringós történetéből kiragadva akár építészeti bravúrként, a komplex városfejlesztés példaterületének is tekinthető, ám érdemes egy pillanatra a háttértörténet mögé nézni, hogyan indult el a kulturális fővárosság felé törekvés Pécsen.

## 2. Az Európa Kulturális Fővárosa cím

Az Európa Kulturális Fővárosa címet 1985 óta ítéli oda az Európai Unió, amellyel a város lehetőséget kap kulturális életének fellendítésére és arra, hogy felhívja magára a figyelmet Európa-szerte. A többmilliárdos pénzbeli forrásokat is megnyitó cím nemcsak a programszervezés terén kínál lehetőséget a városoknak, hanem új kulturális intézmények felépítését és beindítását is szorgalmazza, amelyek a város problémásabb területein elhelyezve afféle fejlődési gócpontként működjenek, hogy a városok olyan részein is megindulhasson a fellendülés, ahol arra korábban nem került sor.

A címet a nemzetközi zsűri 2006-ban ítélte oda Pécsnek, amit a dél-dunántúli település „A határtalan város” programmal pályázott meg. A program nagy ívű fejlesztéseket célt meg, a rendezvényszervezésen túl nagy építészeti vállalkozást is tartalmaztak a tervek: öt kulcsprojekt, egy kiállítótér, a Tudásközpont, egy könyvtár, a Kodály Központ és a Zsolnay Kulturális Negyed létrehozását is megcélozták. A pályázat a kulturális városok alapreceptjét tartalmazta, annak minden idealizmusával együtt. A kreatív város fogalma Nyugat-Európában ismert és élő fogalom, amelynek gyakorlatba ültetésével már számos város vett új fejlődési irányt. Az ilyen városok vezetősége felismeri a kultúra városszervező erejét, és aktívan bevonja a szponzoráló cégeket, civil szervezeteket és bárkit, aki építő javaslataival, szakértelmével hozzájárul a fejlődéshez. A helyi társadalom együtt, a közös igényeknek megfelelően alakítja városát, a központilag elinduló építkezések pedig is egymást erősítik, egy nagy mozaik elemeiként, egymással összhangban valósulnak meg (SOMLYÓDY 2010).

A kreatív városok alapismérve azonban, amely a magyar valóságtól elütni látszik, hogy külhonban e városoknak letisztult, logikus és főként hosszú távra előre meghatározott városrendezési terve van, amelyekben ezek az egyedi, pontszerű építkezések is megtalálják a helyüket, és illeszkednek egy olyan korábban definiált elképzelésbe, amelyre minden mást felfűznek (SÁRI 2010).

Ideális esetben, ha egy település ösztönző erőnek tekinti a kultúrát, akkor a település önkormányzatának partnerként kell tekintenie a kulturális életben szerepet játszó intézményekre, civil szervezetekre és egyéb szereplőkre is, valamint összefogni mindazon

ágazatokat (mint például a városfejlesztés, turizmus, városmarketing), amelyek ennek alakításában szerepet játszanak. Ahogy a pályázat fogalmaz: „*A kultúra tervezése a város tervezését is jelenti és viszont.*” (A határtalan város, p. 6.). Ezt a Közép-Európában még kevésbé elterjedt gyakorlatot és hozzáállást ültette be koncepciójába a még igencsak bürokratikus és konzervatív alapokon működő pécsi önkormányzat is, ahol a kreatív város gyakorlati megvalósításának feltételei még kevésbé voltak adottak. A társadalmi részvétel és szakmai-kreatív erők bevonásának nincsenek még jól bevált hagyományai: emiatt igencsak nagy hátrányból indult „A határtalan város” társadalmi erőkre építő, túlzottan nagy területet körülölelő víziójának megvalósítása. Más EKF-városokban (sikertörténetként Glasgow emelhető ki) ez a cím egy hosszabb fejlődési periódusnak adta meg az utolsó anyagi és szellemi lökést, Pécsen azonban az a pályázat jelentette egy új városkép kidolgozásának kezdetét. Néhány értelmiségi állt csupán a pályázat mögött, annak tulajdonképpen céljait, hosszú távú elképzeléseit hamar előrölték a bürokrácia malmai (SOMLYÓDY 2010).

### 3. Az ArcGIS technológia

A GIS, vagyis Geographic Information Systems, gyűjtőfogalom mindazon eszközöket és szoftvereket magába foglalja, amelyek hely alapú adatok vizualizációjára szolgálnak. GIS segítségével megjeleníthetünk, elemzéseket végezhetünk, az eredményekből pedig kapcsolatok, térbeli minták rajzolódnak ki, láthatóvá válnak a földrajzi térhez köthető összefüggések. A GIS alkalmas hely-alapú döntések meghozására: hol létesüljön egy új lakópark? Milyen nyomvonatra érdemes a tervezett utat építeni? Demográfiai adatok alapján hová épüljön az új általános iskola a városban? E tulajdonsága miatt legfőképp a földrajztudományhoz áll közel, de mint módszer, egyre több szektorban és tudományágban veti meg a lábát, hiszen a hely-alapú gondolkozás számos területen jelen van, legyen az ingatlan-közvetítés, mezőgazdaság, természetvédelem, kormányzati döntések vagy éppen a városfejlesztés (ARCGIS – WHAT IS GIS?).

Az ESRI (Environmental Systems Research Institute) által gyártott ArcGIS szoftvercsalád a térinformatika piacvezető vállalkozása, az 1986-os alapítás óta eltelt csaknem 30 évben komplett platformmá vált, a térképkészítés minden lépését felöleli az adatkezeléstől kezdve a két- és háromdimenziós megjelenítési módszereken át az online megosztásig. Infrastruktúraként is tekinthetünk rá, ahogy azt a 3. ábra is mutatja, a rendszer minden eleme összeköttetésben van egymással. A terepen dolgozók mobil eszközökkel történő felméréseit a desktop-felhasználók valós időben látják és elemezhetik, abból felhő-alapú térképet készíthetnek, végül az eredményt közzé tehetik a világhálón – , ahonnan böngészővel, vagy más okoseszközökkel bárki számára hozzáférhetőek lesznek (ARCGIS – WHAT IS ARCGIS?).

Ahogy Jack Dangermond, az ESRI igazgatója összefoglalja, napjaink GIS-e egyre fejlettebb, a felhő-alapú szolgáltatásuk bevezetésével egyre szélesebb körben válik elérhetővé; a társadalomnak és gazdaságnak pedig ezzel párhuzamosan egyre nagyobb igénye van kifinomult statisztikai adatelemző és vizualizációs eszközökre a megalapozottabb döntések érdekében (ARCGIS AS A PLATFORM). Ez az öngerjesztő folyamat az, amely az ArcGIS-t, mint terméket, és mint technológiát is folyamatos mozgásban tartja és növekedésre serkenti.



3. ábra: az ArcGIS platform részei

*Forrás: ArcGIS Resources What is ArcGIS?*

### 3.1 3D az ArcGIS-ben

#### 3.1.1 ArcGIS Pro

Az újabb és újabb verziók kiadása folyamatos, a 10.3-as Desktop alkalmazást idén, 2014 decemberében dobták piacra az újgenerációs, 64 bites desktop-alkalmazással, az ArcGIS Proval együtt. Az ArcGIS Pro forradalmasítja az ArcGIS eddigi megszokott struktúráját: az eddig külön ablakokban megjelenő funkciókat, pl. térképet az ArcMap-ben, adatkezelést az ArcCatalogban, 3D-s megjelenítést az ArcSceneben és ArcGlobe-ban, megosztást az ArcGIS Online-on keresztül most mind egy felületbe csomagolták, egy sokkal fejlettebb megjelenítési megoldással ötvözve. Az eddigi szabadon elhelyezhető *toolbar*-okat felváltotta a CAD-alapú programokhoz hasonló és az Office-ban megszokott szalagos menüelrendezés, amely folyamatosan változik az éppen aktuális feladathoz igazodva (például címkézés esetén csak a címkézéssel kapcsolatos lehetőségek elérhetőek). A két- és háromdimenziós megjelenítés immár egy felületen, egymással szimultán, osztott ablakelrendezésben történik. Nincs szükség külső szoftverek elindítására, az ArcMap, ArcGlobe és ArcScene funkciói is elérhetőek, ez minden eddiginél gyorsabb

adatfeldolgozást és pontosabb vizualizációt tesz lehetővé (VIDEO.ESRI.COM –UNVEILING THE NEW ARCGIS PROFESSIONAL APPLICATION). Fontos megemlíteni, ez a szoftver nem hivatott felváltani az ArcGIS for Desktopot, inkább azzal azonos szinten álló, más felhasználói élményt nyújtó és szemléletű szoftverről van szó.

### 3.1.2 ArcGlobe

Az ArcGIS for Desktopnak két harmadik dimenzió megjelenítésre alkalmas szoftver is a része, ezek egymástól eltérő jellegű adatok térképezésére használatosak és eltérő funkcionalitással bírnak. Az ArcGlobe a *3D Analyst extension*-ön keresztül érhető el, alapértelmezett felülete a Földgömb, amelyen a navigációs eszközökkel könnyen változtatható a zoomszint, a dőlésszög. Tengereket és óceánokat érintő tematikus adatok – hajózási útvonalak, törésvonalak, hurrikánok nyomvonalának – megjelenítésére is használható (ARCGLOBE OVERVIEW).

Nagy területen lévő nagy adatmennyiség leképezésre képes, emiatt jól használható egész országok 3D-s térképezésére, viszont érdemes figyelembe venni, hogy a 3D Analyst toolbar opciói nem támogatottak, elemzések ezért nem készíthetők vele (ARCGIS RESOURCES –WORKING WITH ARCGLOBE AND ARCSCENE).

### 3.1.3 ArcScene

Az ArcScene-t ezzel szemben inkább kisebb területek perspektivikus ábrázolására fejlesztették ki, segítségével animációk is készíthetők (pl. egy völgy berepülése céljából). A 3D Analyst minden funkciója elérhető benne, ezért az ArcScene legnagyobb előnye az elemzések készítésében rejlik: *multipatchek* importálásával pl. épületek láthatósági, árnyék- és napbesugárzási vizsgálata is megoldható.

Eltér az ArcGlobe-tól abban is, hogy felszín alatti jelenségeket is le lehet vele képezni, így pl. talajszinteket, erőforrásokat, felszín alatti vizeket, kutakat is be lehet vonni a vizsgálatok körébe.

Az ArcGIS technológián belül a negyedik háromdimenziót kezelni képes szoftver a City Engine. Nem része a Desktop alkalmazásoknak, azoktól különálló és céljait tekintve is elüt azoktól, de mivel ESRI termékről van szó, az átjárás a formátumok között biztosítva van. Be lehet illeszteni a munkafolyamatba is, hiszen egy City Engine-t magába foglaló projekthez is szükség van az adatok előkészítésére (ArcMap és ArcCatalog), a City Engine-ben elkészített épületeken pedig az ArcScene-be importálva elemzések készíthetőek.

Dolgozatomban legnagyobb részt ezzel a programmal dolgoztam, ezért a továbbiakban szeretnék a City Engine-ről, használatáról és dolgozatomban elfoglalt funkciójáról és a munkafolyamatról beszélni.

## 4. A City Engine

A City Engine az ArcGIS szoftvercsaládba tartozó, kifejezetten várostervezési és építészeti szemlélettel kifejlesztett, programnyelven alapuló modellező program. A City Engine legnagyobb előnye más tervezőprogramokkal szemben, hogy nagymennyiségű adatot képes dinamikusan és hatékonyan megjeleníteni. A hagyományos modellezéssel szemben, amelynek során a felhasználó közvetlenül dolgozik az objektumokkal és egyesével hozza létre azokat, a City Engine-ben a CGA programnyelv írja le az épületek tulajdonságait szabályfájlok segítségével, amelyeket az épületalaprajzokhoz társítva épül fel a modell. Több száz épület hozható létre alaprajzokból néhány másodperc alatt, egyetlen szabályfájl hozzárendelésével (ARCUSER 2012).



4. ábra: Részletek kidolgozásának menete CGA segítségével

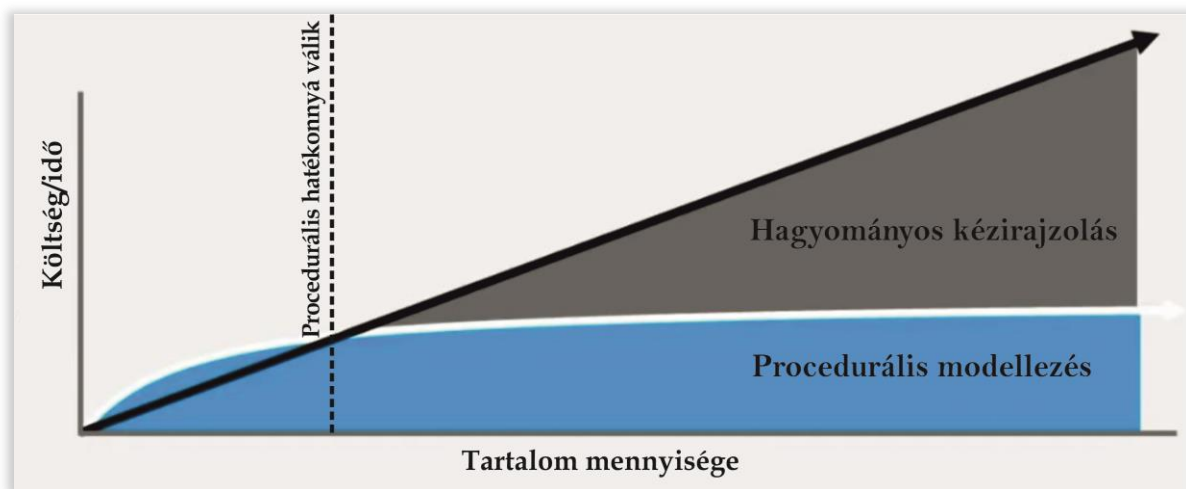
*Forrás: ArcGIS Resources City Engine Overview*

A CGA, vagyis Computer Generated Architecture egy helyben létrehozott vagy importált kétdimenziós alakzathoz (*shape*) kiindulva alkotja meg az adott objektum virtuális modelljét, ahogy az a 4. ábrán is látható. Az egyes falak x és y irányban történő továbbosztásával, textúrák, színek és finomkidolgozású építészeti elemek hozzáadásával válik a fenti épület is egyre élethűbbé. A CGA segítségével kezdőshape-ek importálásán kívül programon belül készült tömegmodelleket is lehet előállítani a City Wizard és az automatikus utcahálózat-szerkesztés (*Grow Streets* menüpont) segítségével. Különböző paraméterek (utcahosszak, településtípus, generált utcák pontos száma) megadásával lehet azoknak megfelelő elképzelt utcahálózatokat lemodellezni (CITY ENGINE HELP – WORKFLOW)

Az elkészült .cga fájlt az épülethez kell rendelni (*Assign Rule File...*), amelynek alapja a Kezdőszabály (*Start Rule*). A Kezdőszabály kulcsfontosságú, anélkül a modell nem fog felépülni, a .cga fájl legelején és az Inspector ablakban, az adott shape tulajdonságainál



is szerepelnie kell – az ablakok funkciójáról bővebben később (CITY ENGINE HELP – GRAMMAR SHAPE MODELLING).



5. ábra: A procedurális és kézi modellezésre fordított idő összehasonlítása

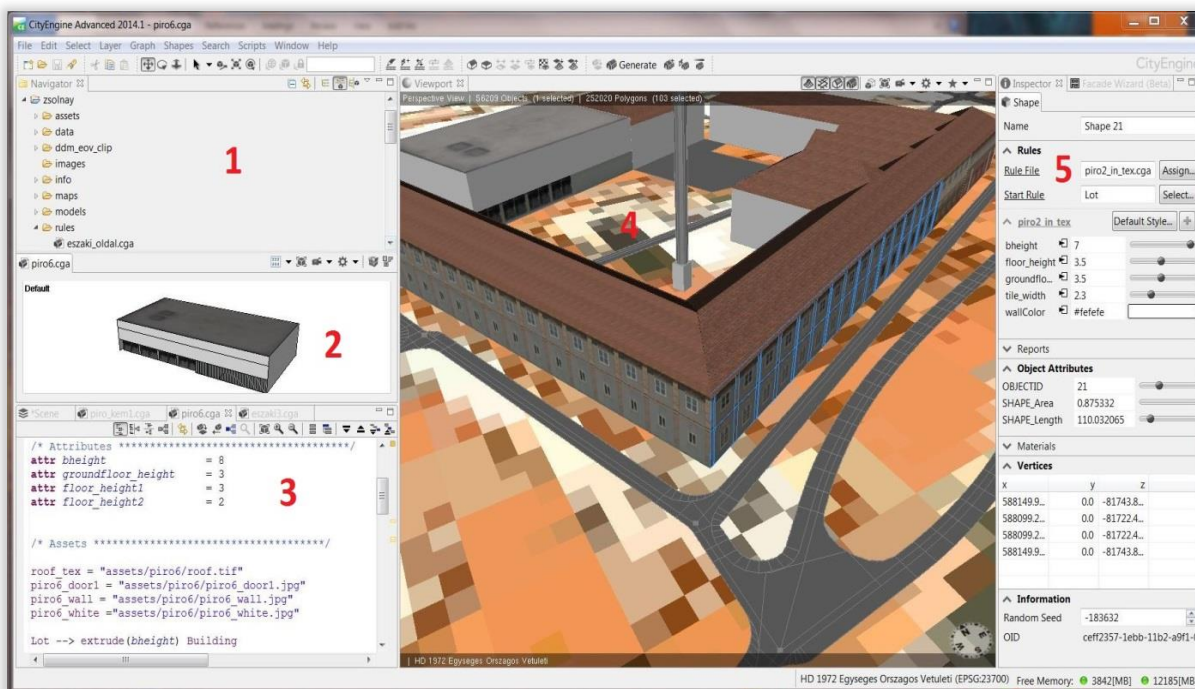
Forrás: *Creating a smart 3D city model from start to finish, City Engine TV Webinar alapján*  
[https://www.youtube.com/watch?v=MxT39BK1\\_iU](https://www.youtube.com/watch?v=MxT39BK1_iU)

Ahogy azt az 5. ábra is szemlélteti, a procedurális modellezés a kezdeti szakaszban időigényesebb a szabályfájlok megalkotása miatt, azonban magának a modellnek a megalkotása a kézi modellezés idejének töredéke. Nagy mennyiségű modell generálásakor, egész városok virtuális felépítéséhez eredményesebb és időgazdaságosabb ezt a módszert választani. Az utólagos módosítások, például a beruházási tervekben bekövetkezett változások egyszerűen adaptálhatóak a már kész szabályfájlban vagy a szoftver kezelőfelületén, az egyik ablakban lévő csúszkák segítségével (CITY ENGINE HELP-GRAMMAR SHAPE MODELING).

Oktatóanyagok kizárólag az ESRI hivatalos honlapján, az ArcGIS Resources oldalon érhetőek el angol nyelven, valamint számos fórum és videómegosztó portálon közzétett tutorial foglalkozik a témával, és az ESRI is több, webinarokon készült videóját teszi közzé hivatalos-félhivatalos csatornáin keresztül. A City Engine-en belül, a Help menüpontban is elérhető egy igen fejlett, fejezetekre bontott segédlet, ebben részletes magyarázatok és kódrészletek is találhatóak.

### 4.1 A felhasználói felület (GUI)

A City Engine komplex szoftver, számos ablakból és ikonok sorából épül fel. Az ablakok külön-külön méretezhetőek, sorrendjük is megcserélhető, ugyanakkor az egyes ablakokban változtathatók a nézetek és egyéb funkciók is elérhetőek. Az átláthatóság és a könnyebb érthetőség kedvéért érdemes kitérni az egyes ablakok tartalmára és a program kezelőfelületének általános felépítésére.



6. ábra: A City Engine egy tipikus ablakelrendezése munka közben  
1. Navigator, 2. Preview, 3. Réteg- és szabályfájlkezelő, 4. Viewport, 5. Inspector

*Forrás: saját szerkesztés*

#### 4.1.1 Navigator

Ebben az ablakban találhatóak annak a workspace-nek a mappái, amelyben az aktuális modellünket szerkesztjük. A projekt alapértelmezett mappákból áll a háromdimenziós térképhez szükséges különböző fájl típusoknak, ki- és bemeneti adatoknak megfelelően.

Ezek:

- *assets*: minden, a szabályfájlok során felhasznált textúrafájl itt található
- *data*: a beimportált gdb-k, obj-k vagy egyéb bemeneti adatforrások tárolására szolgál
- *images*: a Viewportról készült snapshotok, és légifotók elhelyezésére alkalmas
- *maps*: a modell alapjául szolgáló alaptérképeket, felszínmodelleket, domborzatmodelleket stb. teszik tipikusan ebbe a mappába
- *models*: az exportált 3D-s objektumokat szokás ide tenni (pl. .dae, .fbx, .obj)
- *rules*: szabályfájlok gyűjtőhelyeként szolgál.
- *scenes*: City Engine scene-ek tárolására használatos mappa, .cej formátumú fájlok találhatóak benne. Az ArcGIS Desktop világában az .mxd-vel megfeleltethető fájl típus a City Engineben, amely a megjelenítésért felelős.

A Navigator ablakban érhetőek el azok az alapfunkciók, amelyekre a fájlok megnyitásakor szükség van. Az egyes fájlok neveire kattintva a megnyitáson kívül az ablakokban további információkat nyerünk, például szabályfájlok esetén egy kattintással az adott épület legenerált modellje jelenik meg kis méretben az előnézet ablakban, két kattintással szerkeszthetővé is válik az adott fájl a szabályfájl-szerkesztőben.

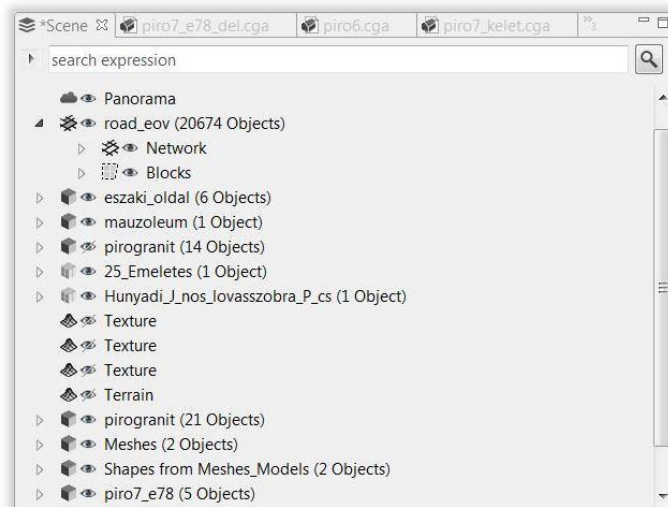
#### 4.1.2 Előnézet (Preview)

Ebben az ablakban kaphatunk további információkat a Navigatorban kiválasztott fájlról. Fájl típusonként más adatok jelennek meg: térképeknél kétdimenziós, szabályfájlok esetén a modellek háromdimenziós előnézete, amelyeknél a kameraállás (perspektivikus, felülről, oldalról, szemből) és a megjelenés (textúrázott, árnyékolt, vázkeretes) is változtatható. Egyéb formátumoknál az elérési út és az utolsó módosítás dátuma jelenik meg. Látszólag kevésbé fontos ablaknak tűnhet, de jelentősen képes meggyorsítani a szerkesztési munkát: ha a CGA fájlban hibát észlel, a modellt nem generálja le előnézetben, az ablak üres marad. Ezzel sok idő takarítható meg, hiszen nem kell újra és újra legenerálni

az egész modellt a Viewportban, elég a szabályfájlt elmenteni és a Navigátor ablakban balkattintással ráállni, rögtön látszani fog, hogy helyesen írtuk-e meg a scriptet.


#### 4.1.3 Réteg- és szabályfájlkezelő (Scene editor, CGA Rule Editor):

Az ablakban alapértelmezett esetben füléken találhatóak a réteg- és a szabályfájl kezelőablakok, de igény szerint ezeket külön ablakba is szét lehet választani, ha azt a képernyő mérete megengedi.



7. ábra: A City Engine rétegkezelő felülete

➤ Rétegkezelő: Mint általában minden térképkészítésre fejlesztett vagy arra is alkalmas szoftverben, a City Engine-ben létrehozott térképek is rétegekből épülnek fel. A scene, amelyben a térképen dolgozunk, ötféle rétegtípusból állhat össze (az ikonjaik alapján könnyen beazonosíthatóak), ezeket az Inspector ablakban lehet szerkeszteni, az attribútumaikat megváltoztatni. Az *Environmental Layer*-ek szabályozzák a scene megvilágítását (☀ Scene Light réteg) és a háttérben a horizont megjelenését (☁ Panorama). A *Map Layer*-ek (🗺) tartalmazzák az összes DDM, alaptérkép, orthofotó stb. jellegű fájlokat, ezeknek is szintén könnyedén állíthatóak a tulajdonságai (pl. Elevation offset, láthatóság, befoglaló téglalap koordinátái). A *Graph Layer*-ekben (🔗) tárolja a program az utcahálózatot (Network) és azok által körbehatárolt telkeket is (Blocks), *Shape Layer*-ek (

 ) a kétdimenziós épületalaprakozokat tartalmazzák. Más programokból beimportált alaprajzokat is shape-ként tárol el a City Engine, amelyeket a CGA fájlokban felhasználunk a 3 dimenziós épületek kiindulópontjaként (CITY ENGINE HELP – CITY ENGINE USER INTERFACE).

➤ CGA Rule Editor: A Navigator ablakban duplán a .cga fájlra kattintva megjelenik az épületet leíró script, vagyis rule file. A fájl kétféle hibatípus különít el, a piros X-szel jelölt syntax errort és a sárga háromszögben látható felkiáltójelet, amely figyelmeztetere utal. A figyelmeztetések csupán lehetséges problémákra hívják fel a figyelmet (pl. nem definiált attribútumok), magát a modellgenerálást nem befolyásolják ellentétben az Error jelzéssel, ezeket minden esetben ki kell javítani, különben a modellt nem tudja felépíteni a szoftver. Nagyobb lélegzetvételi modellek esetén hibakereséshez jó szolgálatot tehet a Problems View ablak, ahol tételesen felsorolva található a scriptben talált összes hiba leírása (CITY ENGINE HELP – RULE ERRORS AND WARNINGS).

#### 4.1.4 Viewport

A Viewport ablakban történik a tulajdonképpeni szerkesztés, ebből adódóan ez foglalja a legnagyobb teret, itt jelölhetőek ki a szerkesztésre szánt objektumok. A jobb felső ikonsorban a Viewportra vonatkozó funkciókat találunk, ezeken keresztül különböző kameraállásokat, fényviszonyokat, könyvjelzőket és egyéb megjelenéssel kapcsolatos tulajdonságokat lehet beállítani. A két talán legfontosabb beállítási lehetőség a *Frame Selection* (gyorsbillentyű F) és az *Isolate Selection*. Előbbivel akár a Viewportban, akár a rétegkezelőben egy adott shape legkisebb befoglaló téglalapjára zoomol, és a továbbiakban annak középpontja körül forgatható a kameraállítás, míg az utóbbira kattintva a kiválasztott objektum marad egyedül látható, a többi ideiglenesen eltűnik a Viewportból, ami rendkívül hasznos funkció, ha olyan oldalait szeretnénk szerkeszteni egy épületnek amelyhez más épület vagy építmény csatlakozik.

#### **4.1.5 Inspector**

Az Inspector ablakban megjelenő információk az éppen kiválasztott objektum tulajdonságait jelenítik meg, ezért az ablak felépítése fájl típusonként és attribútumonként eltérő lehet. A szabályfájlban definiált attribútumok ebben az ablakban csúszkák segítségével módosíthatóak, egy kattintással megnövelhető például az emeletek száma, a földszint vagy az egész épület magassága is. Az Inspector ablakban kell a Start Rule-t definiálni, amelynek összhangban kell lennie a kapcsolódó szabályfájl első meghívott szabályával – enélkül a kapcsolat nélkül nem fog létrejönni a háromdimenziós objektum (CITY ENGINE HELP – USER INTERFACE).

## 5. Módszertan

A City Engine-nel készült térképek egyik tulajdonképpeni célja, hogy egy még meg nem valósult beruházást tárjon a döntéshozók elé oly módon, hogy az a továbbiakban szabadon alakítható legyen a lehető legkevesebb idő és költség alatt, és hogy biztosítsa a fejlesztés során megváltozó földrajzi tér könnyebb megértését. Az elképzelés, hogy egy még nem létező beruházást vegyek mintaterületül, pusztán tervrajzok és látványtervek és nem valós körüljárható épületek alapján készítsem el ezt a modellt, nem volt kivitelezhető, hiszen diákként, civilként, és nem egy fejlesztőiroda munkatársaként ilyen adatok nem álltak rendelkezésemre (ha mégis kezembe kerültek volna ilyen iratok egy ilyen feltételezett helyzetben, az adatvédelem kérdése felmerült volna). Ezenkívül számos egyéb felhasznált forrás nem létezett volna, ha egy még kezdeti stádiumban, csak a tervrajzokon szereplő épületet, épületegyüttest szerettem volna modellezni. Emiatt döntöttem a Zsolnay-negyed mellett, és egy irodalomban gazdagabb, elkészült, publikussá tett tervrajzokkal-tanulmányokkal dokumentált, körüljárható területet modelleztem le.

Ezzel sérül ugyan a modell tulajdonképpeni célja, hogy a dinamikusság révén könnyen alakíthatóak a tervek, a térkép funkcionalitása azonban nem csorbult és lehetőségem nyílt olyan források felhasználására, mint például a Google Earth Pro és a Google Street View, amelyek információtartalma nagyban elősegítette a valósághű ábrázolást.

### 5.1 Az alapanyagok előkészítése

Az alapanyagok előkészítése során megfelelő formátumba és kivágatba kellett helyeznem a rendelkezésre álló forrásokból származó épületalaprajzokat, utcahálózatot és az ortofotót, meg kellett határozni a mintaterületen belüli épületek magasságát és egyéb metrikus tulajdonságait, valamint össze kellett gyűjtenem az épületekhez tartozó textúrákat (pl. ablak, tető, ajtó), amelyeket később a szabályfájlok szerkesztése során hozzárendeltem a házak és kerítések egyes oldalaihoz.

### 5.1.1 *Az épületalaprajzok*

Az elsődleges forrást az épületalaprajzok esetében az OpenStreetMap-ről ingyenesen exportálható poligonok jelentették, azonban ezek pontossága igencsak elnagyolt volt, sok esetben elcsúsztak vagy kevesebb részletet tartalmaztak a valódi objektumhoz képest, ezért tervrajzok és az ortofotó segítségével tettem részletesebbé ezeket.

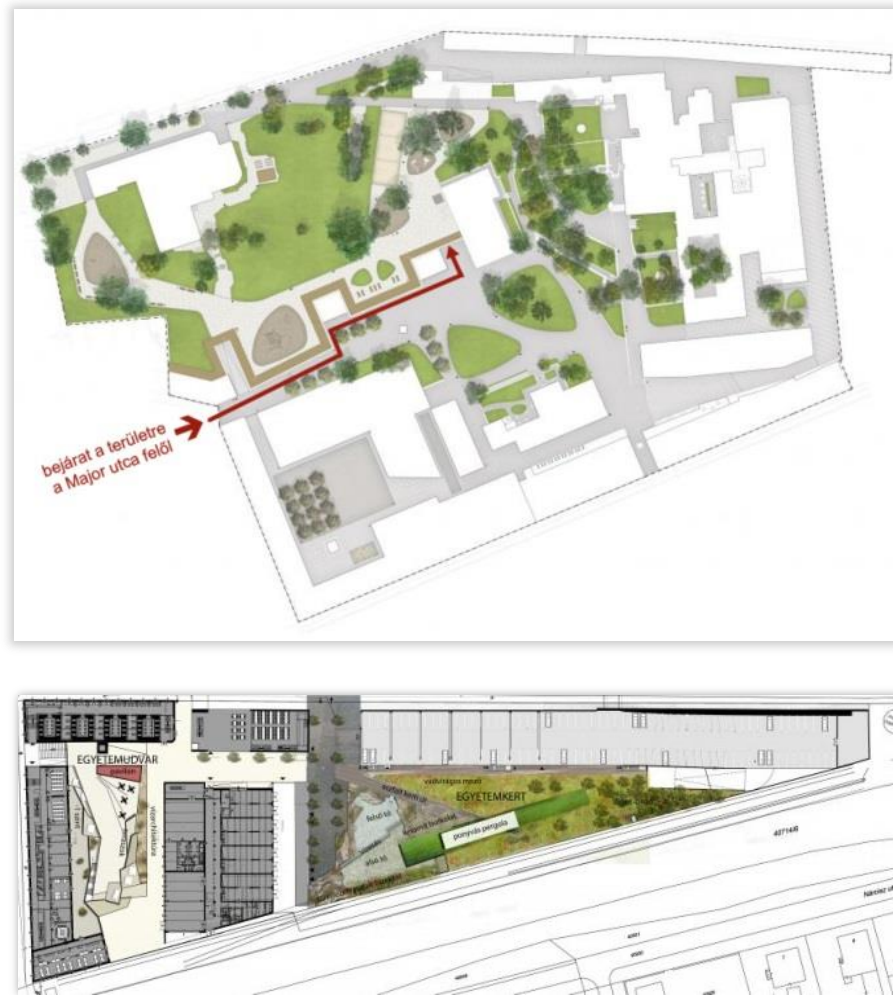
Az OpenStreetMap-ről .shp formátumban töltöttem le az egész Magyarországot tartalmazó épületalaprajzokat és az úthálózatot, majd a kívánt kivágatot egy poligonnal vágtam a tématerületnek megfelelő méretűre a Clip tool segítségével. Az adatokat file geodatabase-be rendeztem a könnyű kezelhetőség kedvéért. File geodatabase használata shapefile vagy personal geodatabase helyett kézenfekvő megoldás volt, mivel végtelen feature-class elhelyezhető benne, egy feature class mérete alapértelmezetten 1 terabyte lehet (de testreszabva ez 265 terabyte-ig növelhető), az adattárolásnak pedig nincsenek terjedelmi korlátai, ezen kívül kevesebb memóriát és kisebb tárhelyet emészt fel és átláthatóan lehet kezelni benne a fájlokat. A shapefile-oknak és personal geodatabase-eknek ezzel szemben 2 gigabyte-os tárolási kapacitásuk van, ezen kívül a shapefile-okkal végzett műveletek lassabban futnak le, és nagyobb memóriát emésztenek fel. A gdb használata mellett szólt az is, hogy az adatok City Engine-be történő importálása így egy ablakban, rövidebb idő alatt elvégezhető, nincs szükség egyedi fájlok külön-külön való kijelölésére. Az is fontos tényező volt, hogy a kész modellek exportálása multipatch-be történt, ez szintén csak geodatabase-ben valósítható meg (COLIN CHILDS 2009).

A kész OSM-adatbázishoz ezek után hozzágeoreferáltam a 8. ábrán található tervrajzokat, amelyek az építészfórum.hu<sup>4</sup>-n közzé tett dokumentumokból és a KÖZ—TÉR—KÖZ Pécs 2010 Kulturális Főváros pályázati munkáit tartalmazó gyűjteményből származnak.

---

<sup>4</sup> Építészeti kérdésekkel foglalkozó internetes szakfolyóirat, gyakran terveket, pályázatokat, dokumentumokat is elérhetővé tesznek.





8. ábra: Az épületek alakjának pontosításához használt tervrajzok a Zsolnay negyed északi és déli területéről

Forrás: [www.epiteszforum.hu](http://www.epiteszforum.hu)

### 5.1.2 Az ortofotó

Alaptérképként egy tif formátumú ortofotó rasztert<sup>5</sup> használtam fel, amelyet a Földmérési és Távérzékelési Intézet bocsátott rendelkezésemre (engedélyszám: FF/1516/1/2014). 2013-ban történt a repülés, ezért a 2011 decemberében átadott Zsolnay Negyed minden része szerepel rajta. A 0,4 m/pixel a legnagyobb elérhető terepi felbontás, megfelelő részletességű épületek sarokpontjainak elkülönítésére és alaptérképnek is. A rasztert a szelvényhatárok tagolása miatt négy részletben kaptam meg, ezért első lépésben

<sup>5</sup> 14-134, 14-143, 14-312 és 14-321 10:000 szelvények tématerületet érintő kivágatai.

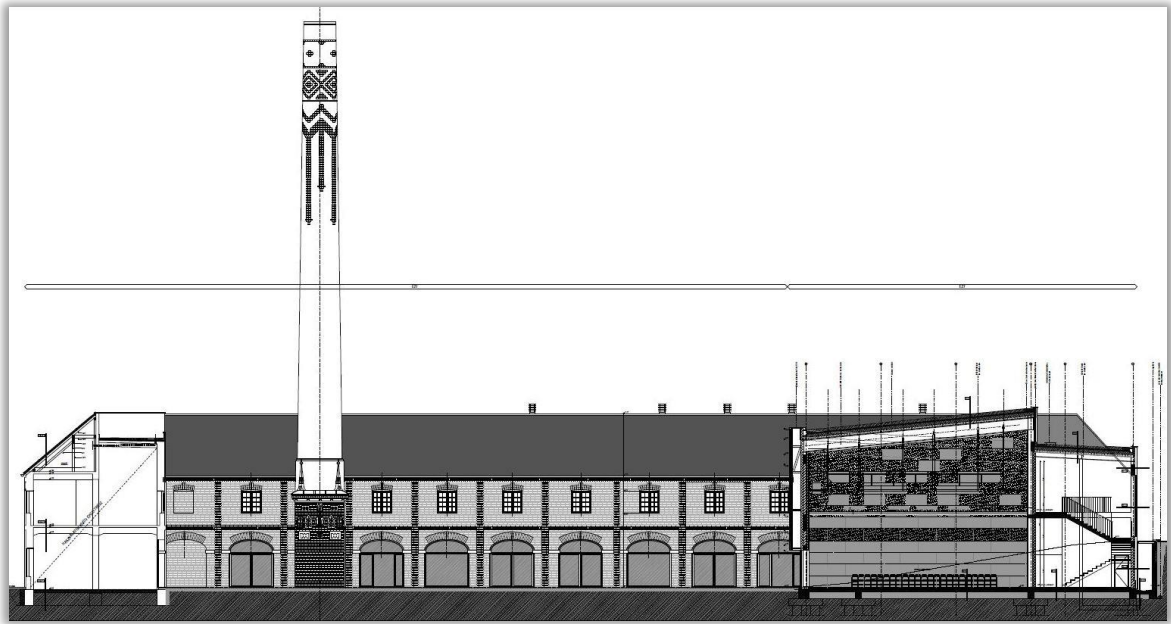
az ArcMap-ben összeforrasztottam őket a Data Management Tool-ok között található Mosaic to New Raster tool segítségével, majd importáltam a City Engine-be a Layer→New Map Layer menüpontra keresztül, Texture-fájlként.

Az ortofotó igen fontos része az elkészült modellnek. Keretet biztosít az épületeknek, igaz, csak két dimenzióban, de megjeleníti a felszín egyéb tulajdonságait, mint például a növényzeti fedettséget és a modellezésből kimaradt környező épületeket és utcákat. Kiterjedésükben és színükben is érzékelhetővé teszi azokat, valamint elhelyezi a fejlesztési területet Pécs egészéhez viszonyítva. A modell szempontjából megfelelőbb egy ilyen ortofotót a térkép alapjául elhelyezni, mint valamilyen várostérképet vagy EOTR-szelvényt, mivel ilyen nagy méretarányban és a felszínhez közel vizsgálódva azok generalizáltsága és felbontása már inkább zavaró lett volna és nagyban csökkentette volna a realisztikus ábrázolás sikerességét.

### 5.1.3 *Magasságadatok*

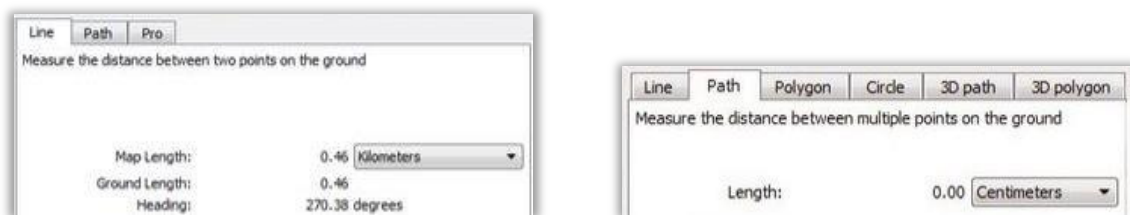
A magasságadatok megléte egy háromdimenziós modell alapvető feltétele, azok nélkül nem valósulhat meg az objektumok kiemelése, nincs azonban könnyű helyzetben az, aki városok épületeinek pontos magasságára kíváncsi, mivel nincsenek olyan nyilvános adatbázisok sem országos, sem városi szinten, ahol ezek összegyűjtve elérhetőek lennének, még a városrendezési tervben sem szerepelnek. Bizonyos üzleti célú térinformatikai adatbázisok tartalmazzák ilyen adatokat, azonban azok is inkább Budapestet részesítik előnyben, a vidéki nagyvárosok adatmérete elhanyagolható és hiányos ilyen szempontból.

Dolgozatom elkészítése során tehát alternatív megoldások felé kellett fordulnom, összességében két adatforrásból sikerült ezeket az adatokat kigyűjtenem: az egyik ilyen forrás a Zsolnay Kulturális Negyed témájában fellelhető tervrajzok, cikkek mellékletei (epiteszforum.hu) és a KÖZ-TÉR-KÖZ EKF-fel kapcsolatos pályázatokat bemutató könyve volt. Ezek a 9. ábrához hasonló részletes rajzokat tartalmazzák a szóban forgó épületekről és perspektivikus torzulás nélkül láthatóak rajtuk az egyes részletek közötti arányok is.



9. ábra: Példa a felhasznált tervrajzokra. A Pirogránit-udvar keletről nyugat felé nézve  
 Forrás: [www.epiteszforum.hu](http://www.epiteszforum.hu)

A másik forrás, amelyet a magasságok meghatározására választottam, a Google Earth Pro. Ez a szoftver a Google Earth üzleti szemlélettel továbbfejlesztett változata, amely elődjéhez képest bővebb funkcionalitással bír. A műholdképek (a Google Earth képernyőfelbontásával ellentétben) nagyfelbontásban letölthetőek és felszíni objektumok mérése is lehetővé vált a sokszög vagy kör területének mérésén és az épületmagasságok meghatározásán keresztül. A Google Earth Pro továbbá támogatja többek között címek tömeges importálását, videók készítését és GIS-adatok importálását is, mindezt természetesen fizetős formában, de elérhető a 30-napos próbaverzió<sup>6</sup>, így ezen keresztül sikerült lemérnem a szükséges épületek adatait (10. ábra).



10. ábra. A Google Earth és a Google Earth Pro mérőeszközei

<sup>6</sup> Letöltés elérhető: <https://www.google.com/work/mapsearch/products/earthpro.html>

#### 5.1.4 Textúrák

Textúrák nélkül egy háromdimenziós városmodell csupán egyszínű alakzatok halmaza lenne, az egyediséget pusztán a magasság, a tető formája és egyéb építészeti elemek biztosítanák, de nem adna realisztikus képet. A textúrák révén lesz egy háromdimenziós modell valóban a valóság leképezését célzó munka és válik térképszerű ábrázolássá. A munkának ehhez a fázisához, az adatgyűjtéshez és –feldolgozáshoz, több szoftvert és a Google Street View-t is felhasználtam. Egyik legfontosabb forrásom a Street View-ról elérhető képek voltak, amelyeket egyesével kivágva (Snipping Tool program segítségével) az *assets* mappában, azon belül az adott épület mappájában tároltam el. A képek elforgatásához és az egyes fényképek közötti színbeli eltérések javításához Microsoft Picture Manager-t használtam.

A Street View bár nagyban hozzájárult a textúrák létrejöttéhez, azok elkészítését mégsem fedi le teljesen, kisebb szűk utcákban, épületek közti átjárókban vagy nem volt elérhető kép, vagy olyan mértékű volt a perspektivikus torzulás, hogy a kép használhatatlan lett volna. A munka kezdeti szakaszában eltervezett terepei bejárást ezért a textúrák hiánya is sürgette, amelyre végül 2014. december 26-án került sor. Az elkészült képekkel sok esetben pótoltam a Street View rossz minőségű felvételeit, és lefedtem vele olyan falakat, amelyeknél nem volt elérhető kép. Probléma volt azonban az, hogy emeleti ablakokat és kiegészítőket (pl. téglaszegély) a földről szintén csak perspektivikus torzulással tudtam lefényképezni, ezért a képek nagy hányadánál felhasználtam a Corel Draw szoftver *3D forgatás*, *Beszívás/Felfújás* és *Perspektíva* eszközeit<sup>7</sup>. A 11. ábrán érzékelhető, hogyan nézett ki egy terepi fotó utómunka előtt valamint a Picture Manager és Corel Draw használata után.

---

<sup>7</sup> Bitképek/3D effektusok menüpont



*11. ábra: Terepen készült fénykép utómunka előtt és után  
Forrás: saját szerkesztés*

Mivel a tetők mintázatáról nem tudtam terepen képet készíteni (ez alól csak egy kivétel van), ezért azokat egyrészt az ortofotó, másrészt a City Engine alapértelmezett textúrárából rekonstruáltam, ahol különféle előre kivágott és egymással átmenet nélkül kapcsolódó textúra található. Ezt a megoldást különösen a gerinces, cseréptetejű épületeknél használtam fel, ahol sokkal nagyobb részletességet lehet így elérni, mint ortofotóval (12. ábra)



12. ábra: Tetőtextúra minősége City Engine textúra és ortofotó használata esetén

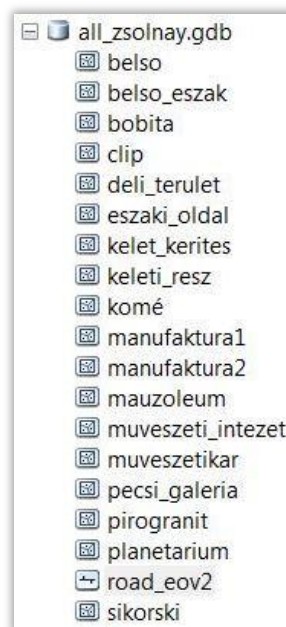
## **5.2 A szabályfájlok**

### *5.2.1 Nevezéktan és tagolás*

A szabályfájlok nevezéktanának kialakításához a feature classok neveit vettem alapul. A gdb-n belül a feature classok a már korábban említett módon, épületcsoportonként szerveződnek (13. ábra), ennek az az oka, hogy a szabályfájlok

szerkesztése során többször volt szükség az épületek utólagos tagolására ArcMapben (*Editor toolbar/Cut polygons tool*).

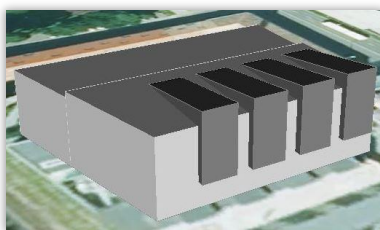
Mivel a City Engine nem érzékeli a geoadatbázisban történő változásokat, ezért újra importálni kellett ezek után a műveletek után a *shape*-eket, ezt pedig jóval egyszerűbb volt mindig csak az adott épületcsoportra elvégezni, csak azt a néhány alakzatot újra elnevezni és szabályfájl társítani hozzá, mint az egész már meglévő modellhalmazt újragenerálni. Az épületekhez tartozó textúramappák és textúrák elnevezésekor rövidítéseket használtam, amelyek a szabályfájlírásnál lerövidítették az azokra való hivatkozások karakterigényét.



13. ábra: A feature classokra tagolt adatbázis

Az épületek ábrázolásánál három féle típus különböztethető meg:

- ❖ Tisztán szabályfájlokkal leírt, két dimenziós kiindulóalakzathoz definiált épületek
- ❖ Mesh típusú objektumok: olyan házrészletek (pl. speciális tetőszerkezetek) amelyek modellezését kézi eszközökkel, a Polygonal Shape Creation és Rectangular Shape Creation eszközökkel végeztem el. Erre csak néhány összetett geometriájú épület miatt volt szükség. Értelemszerűen ezeket a szabályfájlban nem kellett kiemelni (*extrude*), hanem *Envelope* kezdőszabály megadása után az egyes oldalait kellett csak textúrázni.



14. ábra: Egy részben kézzel modellezett épület

- ❖ .dae kiterjesztésű objektumokat töltöttem be a modellbe a Trimble SketchUp 3D Warehouse oldalán szabadon letölthető objektumai közül a kémények ábrázolásához. Ezek alapértelmezetten .kmz vagy .kml formátumban érhetőek el, amelyet egy meglévő scene-be nem tudtam elhelyezni a geokódolás miatt (hibaüzenet: *Data is located too far from current scene's content*), így újraexportáltam a SketchUp-ban és megváltoztattam az eredeti textúráját is, majd egyszerűen behúztam a *scene*-be és elhelyeztem őket a megfelelő helyeken.

### 5.2.2 Szabályfájlok jellegzetes elemei

A modellezéskor felhasznált scriptekre az 1. melléklet tartalmaz egy kidolgozott példát. A CGA nyelvben vannak bizonyos sűrűn használt jelzések, amelyek textúrázásra és az épületek részekre osztására hivatottak. Például ahogy az alábbi

```
Groundfloor→split(x) { ~1.8: Grey_tex}*
```

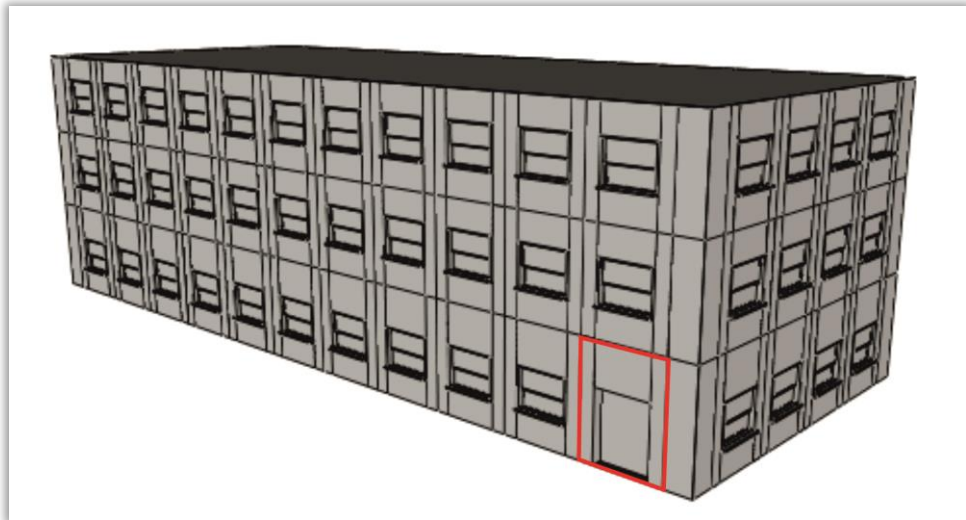
felosztásban látható, a kapcsos zárójelek között szereplő tartalom definiálja egy adott fal földszintjének vízszintes kiosztását (*split(x)*), vagyis egy körülbelül (~) 1,8 méter széles területre illessze be a *Grey\_tex* textúrát, majd ezt ismételgesse (\*) addig, amíg az egész falat ki nem tölti vele. A *Grey\_tex* megfeleltethető egy bizonyos képfájllal, de tovább is osztható *y* irányban is, vagyis tartalmazhat több további textúrát is. A ~ biztosítja azt, hogy bármilyen hosszú is az adott fal, ne maradjon textúra nélkül egyik oldalon sem.

Szabályfájlok írásánál első teendő az attribútumok meghatározása, később ezek script-be ültetésével jelennek meg az Inspector ablakban a csúszkák, amelyekkel a script-ben definiált értékeket utólag tetszés szerint módosítani lehet. Az attribútumok elnevezésénél érdemes figyelni a közérthetőségre, hogy később is egyértelmű legyen, melyik csúszka melyik emeletet, ablakot vagy éppen épületmagasságot módosítja (pl. *floor\_height1*, *tile\_width*). Célszerű minél több attribútumot használni és azokkal definiálni méreteket, hiszen így lesz a kimenet a lehető legtestreszabhatóbb. A fájlban ezután minden később felhasznált textúrát hozzá kell rendelni annak elérési útjához, ez egy részletes kidolgozású háznál 15-20 sor is lehet.

A textúrák után sor kerül a tulajdonképpeni szerkesztésre, amelynek három fő része a *Shape*-ek kiemelése ( *Lot* → *extrude(bheight)Building* ), az ebből következő modell

oldalakra osztása ( $Buildings \rightarrow comp(f) \{...\}$ ) és az egyes oldalak részletes leírása. A  $comp$  művelettel a CGA-ban előre definiált oldalakra vághatjuk fel a házakat. A  $comp$  után zárójelben deklaráljuk, hogy éleket ( $e$ ,  $edges$ ) vagy éppen oldalakat ( $f$ ,  $faces$ ) szeretnénk elválasztani. A szerkesztés során leggyakrabban a Frontfacade, Leftfacade, Rightfacade, Backfacade és Roof kifejezéseket használtam.

Az épületek oldalainak elválasztása után az  $x$  és  $y$  irányú felosztás következik, ebben az ajtók, ablakok és egyéb elemek egymástól való távolságát és arányait kell megírni a  $split$  művelet segítségével. Ahogy a 15. ábrán is látszik, az egyes emeletek  $x$  irányú továbbbontása után tile-ok az ablakok és ajtók elhelyezkedésének függvényében további részekre oszthatóak  $y$  irányban, például a bekeretezett részben látható Entrancetile Door és Wall elemre bontható szét.



15. ábra: Tile-ok elrendezése és felosztása egy egyszerű épület esetében

A plasztikusság érdekében  $.obj$  kiterjesztésű fájlok adhatóak hozzá az épülethez, amellyel az épület síkját megtörő kiszögelléseket és mélységeket lehet érzékeltetni:

```
s('1','1',0.4)
t(0,0,-0.25)
texture(belso2_win)
i(window_asset)
```

Az  $s$  értékkel az objektum befoglalótéglatestjét, vagyis scope-ját adjuk meg az  $x$ ,  $y$  és  $z$  koordinátákkal, a  $t$  ( $translation$ ) értékkel pedig ezt a téglatest mozgatható a homlokzaton,



jelen esetben a -0,25-ös érték az ablakok beugróját hozza létre a homlokzaton belül. A texture művelettel a más korábban leírt elérési úton keresztül kiveszi a rendszer a vonatkozó képfájlt, majd mindezt behelyezi az *i (insert)* művelettel az ablak .obj-vel és scope-pal meghatározott helyére. (A .obj fájlokra elérési útját a textúrák után szokás a script elején leírni).

## 6. Épületekkel kapcsolatos elemzések ArcScene-ben és City Engine-ben

Világunk háromdimenziós, és bizonyos kérdésekre csak háromdimenziós térben tudunk választ adni. Ezt az igényt már több éve felismerték a szoftverfejlesztők, a technika haladásával egyre pontosabb leképezések, virtuális városok elkészítése vált lehetővé, ez a tendencia ma is felfelé ívelőben van: háromdimenziós városmodellekkel foglalkozó szakirodalom, diplomamunkák, cikkek napról napra jelennek meg, gyakran már a napi sajtó is foglalkozik egy-egy érdekesebb esettel. A témában érdekeltek mindinkább törekednek arra, hogy minél pontosabban, minél kevesebb idő alatt, minél költséghatékonyabban legyenek képesek virtuális városokat megjeleníteni egyre fejlettebb elemzőképességekkel egybekötve (SINGH–JAIN–MANDLA 2013).

A zsúfolt városi térben különösen nagy szerepe van ennek a megjelenítési módnak, legyen szó felszín alatti vagy feletti objektumokról, mivel az üzleti érdekek területigényének növekedésével párhuzamosan szükség van olyan elemzésekre, amelyek segítenek a döntéshozók elé tárni például egy új színház, irodaépület vagy éppen egy rekultiválandó terület városi térre kifejtett hatását. Milyen magasra építhető bizonyos területen egy épület anélkül, hogy kitüntetett helyekről nézve megváltoztatná a horizontot? A tervezet irodaépület udvarába mennyi fény jut be? Mekkora árnyékot fog vetni az új épület a környező házakra? Az utcáról nézve meddig látnak be az emberek a lakásomba? Az új autópálya egyes pontjairól meddig látom be a tájat? Ilyen és ehhez hasonló kérdések mind-mind segítenek megérteni és elképzelni azt, milyen hatása van a fejlesztési beavatkozásoknak a földrajzi térre (SOLVE 3D PROBLEMS). Egy új kutatási irány, a háromdimenziós ingatlan-nyilvántartások modellezése például segít a felszín feletti és az alatti objektumok, mint például vezetékek, pincék tulajdonviszonyainak tisztázásában. Egy ilyen komplex rendszer kidolgozása jogi viták rendezésének referenciájaként is használható lenne, sajnos Magyarországon még alig néhány kutatója van.

### 6.1 Az elkészített elemzések

Az ArcGIS platform több lehetőséget is kínál elemzések készítésére, ezekből szeretnék most kiragadni néhány elkészített példát. Az ArcGIS for Desktopon belül a *3d*

*Analyst* foglalkozik ilyen kérdésekkel, de ezeken az eszközökön túl is vannak további lehetőségek. Az <http://desktop.arcgis.com/en/3d/> oldalon az ESRI által kifejlesztett előre definiált 3D city template-ek találhatóak, amelyek már felépített modelleket<sup>8</sup> és dokumentációt tartalmaznak, amely alapján saját adatok is egyszerűen lefutttathatóak az elemzések.

Dolgozatomban a 3D city template-ek közül a napbesugárzási vizsgálatot (*Solar Analysis*), a 3D *Analyst* eszközei közül a Create Sight Lines és Intervisibility-t végeztem el.

### 6.1 Napbesugárzási vizsgálat

A vizsgálathoz szükség volt a letölthető template-re, az exportált épületmultipatch-ekre, az ArcGIS for Desktopon belül a Data Interoperability extension-re és ArcScene-re. Első lépésként a falakat és tetőket kellett elválasztani egymástól, ehhez ki kellett jelölni az összes modellt, majd a Shapes → Separate faces gombbal azokat különálló objektumokká tenni, ezt követően hozzájuk rendeltem a template-ben található GetRoofs.cga-t, amely a tetők leválogatását elvégzi. A leválogatott tetőelemeket ezután egy új geoadatbázisba kell exportálni. Ezt a folyamatot a falak leválogatásánál is meg kellett ismételni azzal a különbséggel, hogy ebben az esetben az Inspector ablakban a GetRoofs attribútumot “false”-ra állítottam.

Az adatelőkészítés után következhetett az elemzés lefutattása, ehhez a template-ben található *SolarRadiationBurnBuildings* modellre (workflow) volt szükség, amely a tetők dőlésszöge alapján ad értéket azoknak. Következő lépésként a *SolarRadiationRoofZones* modellt is le kellett futtatni, amely sávosan osztályozza a kapott napbesugárzás átlagos értékeit (kWh/m<sup>2</sup>/nap)és hozzárendeli azokat az egyes tetőelemekhez és falakhoz. Ennél a tool-nál kimenetként az adatelőkészítés során definiált Roofs és Walls feature classokat kell megadni, amelyekbe a tool lefuttatása után új mezők kerülnek:

---

<sup>8</sup> ebben az értelemben munkafolyamatot jelent. Egymáshoz kapcsolt tool-okból áll, az előző outputját használja fel a következő tool az inputjaként. Modellek létrehozásával nagymennyiségű adat dolgozható fel egyszerűen, bármikor újra felhasználható vázat adnak gyakran használt munkafolyamatokhoz.

- ❖ *SolarRadiation*: átlagos évi besugárzás mértékét (kWh/m<sup>2</sup>/nap) adja meg
- ❖ *SolarSuitability*: kétfelé, alkalmas és nem alkalmas osztályra bontja a tetőket besugárzási értékük szerint, 1.72 kWh/m<sup>2</sup>/nap alatt *NotSuitable* értéket kap, afelett *Suitable*-t.

A kész, attribútumokkal bővített feature class-okat ezután importáltam City Engine-be, és egy korábban megszerkesztett solar.cga fájlt rendeltem hozzá a shape-ekhez. Ebben van definiálva, hogy milyen színeket vegyen fel az adott tetőelem bizonyos értékek esetén:

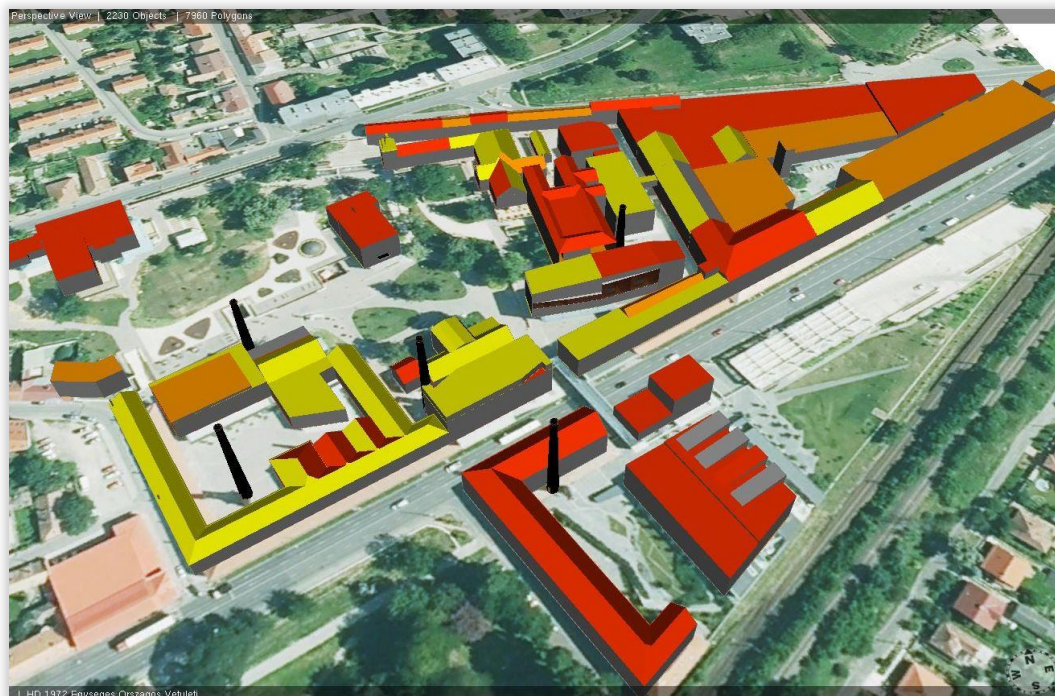
```
SolarRad -->
  case SolarRadiation == 0 :
    color(wall_color)
  case SolarRadiation > 0.1 && SolarRadiation < 2:
    color(rad1_color)
  case SolarRadiation > 2 && SolarRadiation < 3 :
    color(rad2_color)
  case SolarRadiation > 3:
    color(rad3_color)
  else:
    NIL.
```

```
SolarSuit -->
  case SolarSuitability == "Not Suitable" :
    color(not_suitable_color)

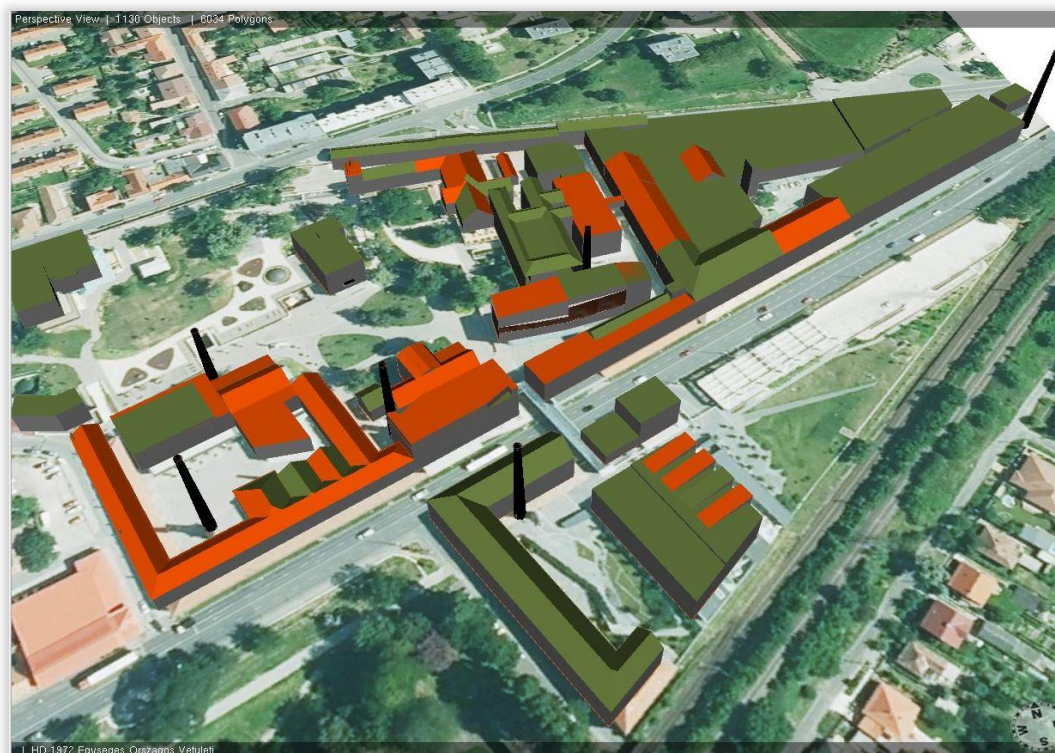
  case SolarSuitability == "Suitable" :
    color(suitable_color)

  else:
    color(wall_color)
```

A kimenet tehát két feature class, amelyeket City Engine-ben egy szabályfájlon keresztül tettem tematikussá.



16. ábra: Besugárzás mértéke City Engine-ben tematizálva: citrom: kis mennyiségű, narancs: közepes, piros: erős



17. ábra: Besugárzás mértéke napenergiatermelésre való alkalmassága szerint  
Zöld: Alkalmas, narancs: nem alkalmas

## 6.2. Láthatósági vizsgálat

A láthatósági vizsgálatához két *3D Analyst*-on belüli tool-ra volt szükség. Az adatelőkészítés során megfigyelőpontokat és nézőpontokat kellett meghatározni, vagyis mely pontokból néz a megfigyelő mely pontokba. Ehhez ArcMap-ben létrehoztam egy geodatbázist, azon belül két pont feature classt *target* és *observer* néven. Editor módban ezután mindegyik sorhoz hozzáadtam a Z koordinátát. A megfigyelendő pontokat többnyire épületek sarkaira helyeztem el, így a magasságadatok adottak voltak.

A feature classok pontjait ezután ArcScene-ben jelenítettem meg három dimenzióban térbeli szimbólumokkal, amelyekhez a szükséges magasságot a program a feature classok Z mezőjéből olvasta ki. A pontok megjelenítése után a 3D Analyston belül található *Create Sight Lines* toolt futtattam le, amelyhez inputként a pontrétegeket adtam meg. A tool kimenete egy vonal feature class, amely vonalakat generált a megfigyelő és a megfigyelendő pontok között. Ezek a vonalak még önmagukban nem hordoznak információt arról, hogy az adott pontból látható-e a kérdéses épület, az épületeket nem veszi figyelembe akadályként, hiszen a tool inputjaként sem kellett azokat megadni, ezért a kapott vonalakon az *Intervisibility* elemzőeszközt is le kellett futtatni, amely a *Create Sight Lines* eredményfájl-ját és az épületek modelljét felhasználva egy új mezőt hoz létre az attribútumtáblában: 0 vagy 1-es értéket kap a vonal aszerint, hogy látható-e a megfigyelendő pont vagy sem.

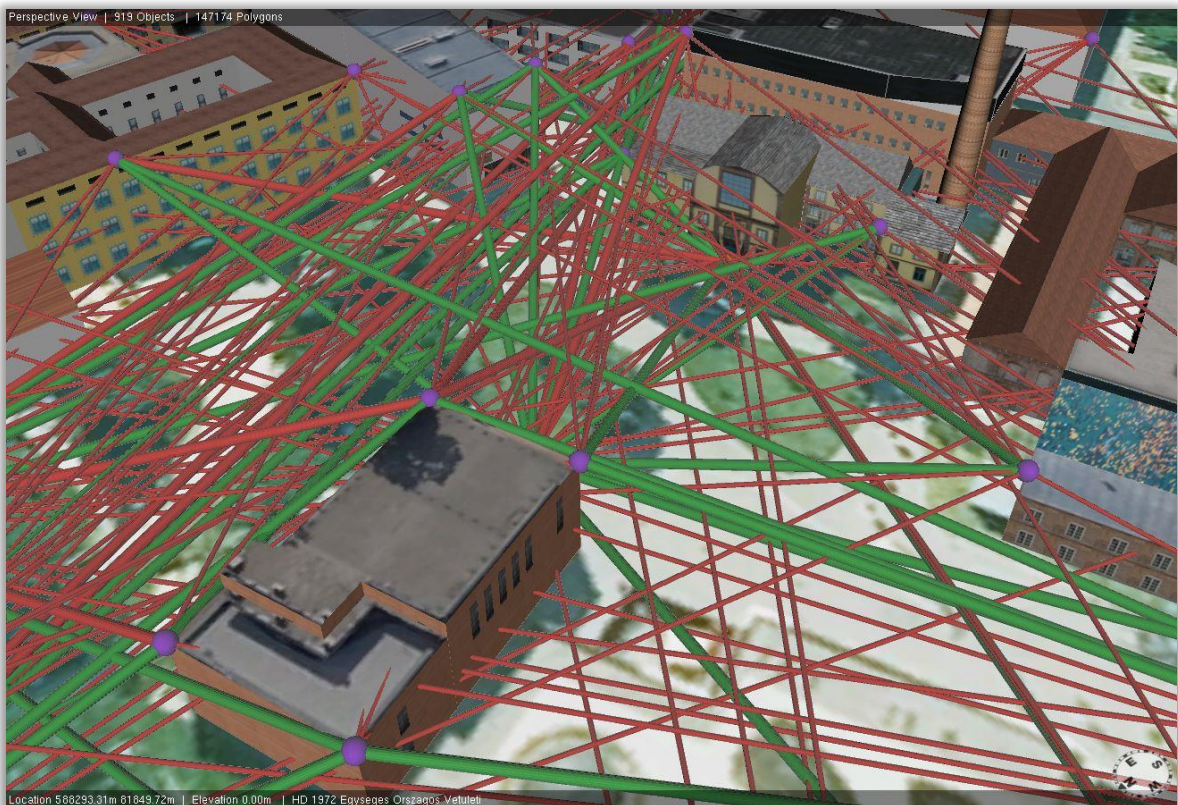
Az *Intervisibility* tool-al bővített vonalaknak ezután térbeli szimbólumot, vagyis hengert (*tube*) kellett megadni, mivel a City Engine nem tud kezelni pusztán pont és vonalobjektumokat. Az így átalakított geometriát multipatch-be exportáltam, és importáltam City Engine-be. A színeket ebben az esetben is egy általam megírt szabályfájllal határoztam meg:

```
@StartRule
Target -->
  color(target_color)
  set(material.opacity,0.5)

@StartRule
Observer -->
  color(observer_color)
  set(material.opacity,0.8)
```

```
@StartRule  
SightLine -->  
  case VISIBLE == "false" :  
    color(visible_piros)  
    set(material.opacity, 0.5)  
  else:  
    color(visible_zold)  
    set(material.opacity, 0.5)
```

A megfigyelőpontok narancssárgán, a megfigyelandő pontok lilán jelennek meg, a vonalak pedig a láthatóság függvényében kaptak piros vagy zöld színt, amelyet a “visible” mezőből állapított meg a program (18. ábra).

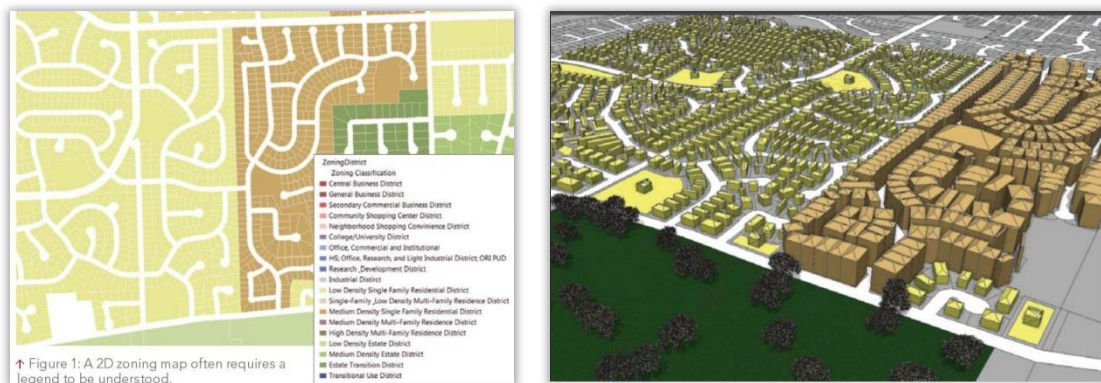


18. ábra: Láthatósági vizsgálat eredménye. zöld: látható, piros: nem látható

### 6.3 Beépítési szabályok ábrázolása

Bár az adatigénye miatt a dolgozatba nem került be ez az ábrázolási módszer, fontosnak tartom megemlíteni a városfejlesztésben játszott szerepe miatt. Akár külön diplomamunkát is megérne egy ilyen térkép elkészítése, de mivel az önkormányzat rendezési terve többnyire (ahogy Pécs esetében is) csak raszteres formában érhető el, ezért az adatelőkészítés időigényessége miatt ezt az elemzést nem végeztem el.

A már korábban említett 3D city template-ek között elérhető az úgynevezett *Zoning Regulations* is, amellyel a rendezési tervekben számszerűsített és tipizált zónák ábrázolhatóak (Isd. 19. ábra). A városrendezési tervek és építési szabályzatok kusza grafikájú térképeinke, jogi kifejezésekkel teli paragrafusainak vizuális megjelenítésére és azok felhasználására hivatott ez a módszer. Építési magasságok, beépítési területek típusainak (pl. lakóövezet, kertés övezet, iparterület) elkülönítésével és cga-val definiált szimbólumaival válik egy ilyen munka a városfejlesztésben használt interaktív térképpé, amely rendszerbe a tervezett épületeket elhelyezve, könnyen felismerhetőek a szabályoknak nem megfelelő éplettulajdonságok (ARCUSER 2012).



19. ábra: Beépítési zónák két- és háromdimenzióban City Engine-en keresztül

A térképet .3ws formátumban exportáltam az ArcGIS Online-ra City Engine Web Viewerként, ahol ki-bekapcsolható rétegekként megtekinthető az összes modell és az elemzések eredménye is. A térkép a világhálón nincs megosztva, csak az ArcGIS Online-fiókba való bejelentkezés után tekinthető meg.



## ÖSSZEGZÉS

A Zsolnay Kulturális Negyed területe ideális tématerület volt egy háromdimenziós modell megvalósításához. Kompaktsága, történelmi hagyománya, a terület közelmúltban történt megújulása, a textúrák és a házak egyedisége révén kerek egységet alkot és ideális háttérrel biztosított egy ilyen jellegű munka elkészüléséhez. A modell önmagában is térképészeti értékkel és gazdag információtartalommal bír a tématerületről, ezért úgy gondolom önállóan is megállja a helyét, az elemzésekkel kiegészítve azonban tematikus adattartalom is kapcsolódik hozzá, amely nagyban megnöveli térinformatikai és információértékét is.

Úgy gondolom, a feltett kérdésekre csak részben lehet válaszolni egy ilyen modellel, egy városfejlesztési program sikerességét nem lehet csupán térinformatikai eszközökkel mérni. A láthatósági, napbesugárzási vizsgálat ugyan hozzájárul a fizikai környezet megváltozásának vizsgálatához vagy annak tulajdonságainak méréséhez, azonban pusztán ilyen és ezekhez hasonló metrikus eszközökkel nem igazolható sem egy elkészült fejlesztés sikeressége, sem egy tervezett épület létjogosultsága. A térképalkalmazásom egységet alkot, modellezési és térinformatikai szempontból pedig rendkívül tanulságos volt elkészítése, de legyen bármilyen fejlett vagy elemzőeszközökben gazdag is egy térképészeti-geoinformatikai munka, nem hanyagolhatóak el olyan a kvantitatív eszközök sem az adott fejlesztési terület vizsgálatakor, mint amit egy városi, megyei vagy országos szintű hatásvizsgálat tud nyújtani statisztikai, társadalmi és gazdasági adatokra alapozva.

## FELHASZNÁLT IRODALOM

### *KÖNYV, CIKK*

SÁRI ISTVÁN (2010): A kreatív város. Építészfórum  
<http://epiteszforum.hu/a-kreativ-varos>

SURENDRA PAL SINGH–KAMAL JAIN–RAVIBABU MANDLA (2013): Virtual 3D city modeling: techniques an International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.

<http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-2-W2/73/2013/isprsarchives-XL-2-W2-73-2013.pdf>

SOMLYÓDY NÓRA (2010): A Balkán kapuja? Pesti Kalligram Kft.

SOMOGYI KRISZTINA–CZAKÓ ZSOLT (SZERK.): KÖZ–TÉR–KÖZ – PÉCS2010 EURÓPA KULTURÁLIS FŐVÁROSA ÉPÍTÉSZETI PÁLYÁZATOK. HUNGAROFEST KHT, 2010.

ZSOLNAY TERÉZ (1975): Zsolnay – A gyár és család története 1863-1948, a gyár története 1948-1973. Corvina Kiadó, Budapest.

### *INTERNETES FORRÁSOK*

COLIN CHILDS (2009): Top nine reasons to use a file geodatabase  
<http://www.esri.com/news/arcuser/0309/files/9reasons.pdf>

A ZSOLNAY-HAGYOMÁNY: EGY VILÁGHÍRŰ GYÁR MÁSFÉL ÉVSZÁZADA  
<http://www.ertekmentok.hu/>

ARCGIS RESOURCES WORKING WITH ARCGLOBE AND ARCSCENE  
<http://resources.arcgis.com/en/help/main/10.1/index.html#//00q8000000sv000000>

ARCGIS RESOURCES WHAT IS ARCGIS?  
<http://resources.arcgis.com/en/help/getting-started/articles/026n00000014000000.htm>

ARCGIS AS A PLATFORM  
<https://www.govloop.com/resources/arcgis-as-a-platform-an-interview-with-esri-president-jack-dangermond/>

ARCGLOBE OVERVIEW

<https://www.youtube.com/watch?v=LCGxS5k5FIo>

ARCUSER 2012 WINTER

<http://www.esri.com/news/arcuser/0112/files/cityengine.pdf>

CITY ENGINE HELP – CITY ENGINE OVERVIEW

<http://resources.arcgis.com/en/help/cityengine/10.2/index.html>

CITY ENGINE HELP – CITY ENGINE WORKFLOW

[http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/CityEngine\\_Workflow/02w100000007000000/](http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/CityEngine_Workflow/02w100000007000000/)

CITY ENGINE HELP – RULE-BASED MODELING

[http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/Rule\\_based\\_Modeling/02w100000004000000/](http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/Rule_based_Modeling/02w100000004000000/)

CITY ENGINE HELP – USER INTERFACE

[http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/User\\_Interface/02w100000002000000/](http://resources.arcgis.com/en/help/cityengine/10.2/index.html#/User_Interface/02w100000002000000/)

CREATING A SMART 3D CITY MODEL FROM START TO FINISH, CITY ENGINE TV WEBINAR

[https://www.youtube.com/watch?v=MxT39BK1\\_iU](https://www.youtube.com/watch?v=MxT39BK1_iU)

GYÁRTÖRTÉNET

<http://www.zsolnay.hu/hu/gyartortenet>

PÁVA ZSOLT BÍZIK AZ ÚJ ZSOLNAY-TULAJDONOSBAN: HVG 2013. február 16.

[http://hvg.hu/kkv/20130216\\_Pava\\_Zsolt\\_bizik\\_az\\_uj\\_Zsolnaytulajdonos](http://hvg.hu/kkv/20130216_Pava_Zsolt_bizik_az_uj_Zsolnaytulajdonos)

SOLVE 3D PROBLEMS

<http://video.arcgis.com/watch/2198/solve-3d-problems>

UNVEILING THE NEW ARCGIS PROFESSIONAL APPLICATION

<http://video.esri.com/watch/2533/unveiling-the-new-arcgis-professional-application-with-jim-mckinney-and-jack-dangermond>

ÁBRÁK, KÉPEK JEGYZÉKE

|                                                                                                                        |    |
|------------------------------------------------------------------------------------------------------------------------|----|
| 1. ábra: Pirogránit elemek a Mátyás-templom tetején.....                                                               | 8  |
| 2. ábra: Eozin-mázás bikafej, az Európa Kulturális Fővárosa program pécsi jelképe....                                  | 8  |
| 3. ábra: Az ArcGIS platform részei.....                                                                                | 13 |
| 4. ábra: Részletek kidolgozásának mentee CGA segítségével.....                                                         | 16 |
| 5. ábra: A procedurális és kézi modellezésre fordított idő összehasonlítása.....                                       | 17 |
| 6. ábra: A City Engine egy tipikus ablakelrendezése munka közben.....                                                  | 18 |
| 7. ábra: A City Engine rétegkezelő felülete.....                                                                       | 20 |
| 8. ábra: Az épületek alakjának pontosításához használt tervrajzok a Zsolnay negyed északi és déli területéről.....     | 25 |
| 9. ábra: Példa a felhasznált tervrajzokra. A Pirogránit-udvar keletről nyugat felé nézve.....                          | 27 |
| 10. ábra: A Google Earth és a Google Earth Pro mérőeszközei.....                                                       | 27 |
| 11. ábra: Terepen készült fénykép utómunka előtt és után.....                                                          | 29 |
| 12. ábra: Tetőtextúra minősége City Engine textúra és ortofotó használata esetén.....                                  | 29 |
| 13. ábra: A feature classokra tagolt adatbázis.....                                                                    | 30 |
| 14. ábra: Egy részben kézzel modellezett épület.....                                                                   | 30 |
| 15. ábra: Tile-ok elrendezése és felosztása egy egyszerű épület esetében.....                                          | 32 |
| 16. Besugárzás mértéke City Engine-ben tematizálva: citrom: kis mennyiségű, narancs: közepes, piros: erős.....         | 37 |
| 17. ábra: Besugárzás mértéke napenergiatermelésre való alkalmassága szerint Zöld: Alkalmas, narancs: nem alkalmas..... | 37 |
| 18. Láthatósági vizsgálat eredménye. zöld: látható, piros: nem látható.....                                            | 39 |
| 19. ábra: Beépítési zónák két- és háromdimenzióban City Engine-en keresztül.....                                       | 40 |

MELLÉKLETEK

## 1. melléklet

```

/**
 * File:      piro6.cga
 * Created:   14 Dec 2014 18:36:44 GMT
 * Author:    mesterha
 */

version "2014.1"

/* Attributes *****/
attr bheight          = 8
attr groundfloor_height = 3
attr floor_height1    = 3
attr floor_height2    = 2
attr ledge_height     = 1.4
attr left_gr          = 6.7
attr left_fl          = 1.3

/* Assets *****/

roof_tex              = "assets/piro6/roof.tif"
piro6_door            = "assets/piro6/piro6_door.jpg"
piro6_wall            = "assets/piro6/piro6_wall.jpg"
piro6_white           = "assets/piro6/piro6_white.jpg"
piro6_grey_wall       = "assets/piro6/piro6_grey_wall.jpg"
piro6_num1            = "assets/piro6/piro6_num1.jpg"

piro6_back_win        = "assets/piro6/piro6_back_win.jpg"
piro6_back_win2       = "assets/piro6/piro6_back_win2.jpg"
piro6_back_door       = "assets/piro6/piro6_back_door.jpg"
piro6_back_elevator   = "assets/piro6/piro6_back_elevator.jpg"

Lot --> extrude(bheight) Building

Building -->
  comp(f) { front : Frontfacade | back : Backfacade | left: Leftfacade |
    right: Rightfacade | top: Roof}

#FRONTFACADE-----

Frontfacade -->
  split(y) { groundfloor_height : Groundfloor | floor_height1 : Floor1
    | floor_height2: Floor2 }

Groundfloor -->
  split(x) { 5: Tile | 5: Tile | 5: Tile | 5: Tile | ~8: Gr_white }

Tile -->
  split(x) {~0.3: Brownwall |~3: Entrance }

```

```

Brownwall -->
    s('1.1, '1, 0.1)
    t(0, 0, -0.2)
    texture(piro6_wall)
    i("builtin:cube")

Entrance -->
    s('1, '1, 0.1)
    t(0, 0, -0.3)
    texture(piro6_door)
    i("builtin:cube")

Gr_white -->
    split(x) { {~0.3: Gr_white_tex}*}

Gr_white_tex -->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_white)

Floor1 -->
    split(x) { 0.3: Gr_white_tex}*

Floor2 -->
    split(x) { 0.3: Gr_white_tex}*

#LEFTFACADE-----

Leftfacade -->
    split(y) {left_gr : Left_ground | left_fl : Left_floor}

Left_ground -->
    split(x) { ~1.8: Grey_tex}*

Grey_tex -->
    setupProjection(0, scope.xy, 1.2 ,1, 1)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_grey_wall)

Left_floor -->
    split(x) { 1.8: Grey_num | ~1.8: Grey_tex}

Grey_num -->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_num1)

##RIGHTFACADE-----piro9 felóli-----
Rightfacade-->
    split(x) {{~2.5: Grey_tex}*}

##BACKFACADE-----
Backfacade-->
    split(x) { ~15: Window_tile1 | 2.5: Entrance_tile | ~5: Window_tile2}

```

```

Window_tile1-->
    split(y) { groundfloor_height : Groundfloor_back
              | ~ledge_height : Grey_tex
              | floor_height1 : Floor_back}

Groundfloor_back-->
    split(x) { ~1: Grey_tex
              | 1: Back_win_tile1
              | 0.4: Grey_tex
              | 1: Back_win_tile1
              | ~2: Grey_tex
              | 1: Back_win_tile1
              | ~2.5: Grey_tex
              | 1: Back_win_tile1
              | ~2: Grey_tex
              | 1: Back_win_tile1
              | ~2: Grey_tex}

Back_win_tile1-->
    split(y) { 2: Back_win | ~1: Grey_tex}

Back_win-->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_back_win)

Floor_back-->
    split(x) { ~2: Grey_tex | {~1: Back_win_tile2 | ~3: Grey_tex}*}

Back_win_tile2-->
    split(y) { ~1: Grey_tex | 1: Back_win2 | ~4: Grey_tex}

Back_win2-->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_back_win2)

Entrance_tile-->
    split(y) { 3.5: Entrance_back | ~5: Grey_tex}

Entrance_back-->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_back_door)

Window_tile2-->
    split(y) { groundfloor_height : Groundfloor_back2
              | ~ledge_height : Grey_tex
              | floor_height1 : Floor_back2}

Groundfloor_back2-->
    split(x) {~2: Grey_tex
              | 1.3: Elevator_tile
              | ~2: Grey_tex
              | 0.9: Back_win_tile1
              | ~1: Grey_tex}

```

```

Elevator_tile-->
    split(y) {2: Elevator
              | ~1: Grey_tex}
Elevator-->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    setupProjection(2, scope.xy, scope.sx, scope.sy)
    projectUV(0)
    texture(piro6_back_elevator)

Floor_back2-->
    split(x) { ~1.5: Grey_tex
              | {1: Back_win_tile2
                | ~2: Grey_tex}*
              | 1.6: Grey_num_tile}
Grey_num_tile-->
    split(y) { ~2: Grey_tex | 1.3: Grey_num}

#ROOF -----
Roof -->
    setupProjection(0, scope.xy, scope.sx, scope.sy)
    texture(roof_tex)
    projectUV(0)

```





2. melléklet: A FÖMI-től kapott teljes ortofotó