



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Alba Regia Műszaki Kar
Geoinformatikai Intézet

SZAKDOLGOZAT

Adatbázisok geometriai adattípusainak használata egy üzemi információs rendszerben

OE-AMK

2023

Hallgató neve: Tasnádi Bálint Péter

Hallgató törzskönyvi száma: T000752/FI12904/A-G



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Óbudai Egyetem
Alba Regia Műszaki Kar
Geoinformatikai Intézet

SZAKDOLGOZAT FELADATLAP

Hallgató neve: Tasnádi Bálint Péter

Szaktervezési száma: [REDACTED]

Törzskönyvi száma: T000752/FI12904/A-G

Neptun kódja: [REDACTED]

Szak: geoinformatikai szakember

Specializáció:

A dolgozat címe: Adatbázisok geometriai adattípusainak használata egy üzemi információs rendszerben

A dolgozat címe angolul: Use of the geometry data types from databases in a business information system

A feladat részletezése:

Szaktervezési témavázlat

Szerkezet:

- 1) A geometriai adatok kezelésének bemutatása a térinformatikai rendszerekben.
- 2) Térbeli/geometriai adatok tárolására és kezelésére alkalmas relációsadatbázis-kezelő rendszerek - illetve azok kiterjesztéseinek - áttekintése: PostgreSQL és PostGIS, SQLite és Spatialite, Oracle és Oracle Spatial and Graph, MS SQL Server és MS SQL Spatial.
- 3) Ezen adatbázis-kezelő rendszerek összehasonlítása és annak vizsgálata, hogy melyik a legalkalmasabb egy „üzemi információs rendszerben” („PI System”) történő használatra.
- 4) Prototípuszerű átvitel a Lovion BIS (<https://lovion.de/en/lovion-main-page-en>) nevű „üzemi információs rendszerbe” („PI System”).

Intézményi konzulens neve: Pödör Andrea Dr.

A kiadott téma elévülési határideje: 2025. 07. 10.

Beadási határidő: 2023. 05. 15.

A szaktervezési: Nem titkos.

Kiadva: Székesfehérvár, 2023. 03. 27.



[Signature]

Intézetigazgató

A dolgozatot beadásra alkalmasnak találom: 2023. 05. 15.

[Signature]

belső konzulens

.....

külső konzulens



HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat/diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozat/diplomamunkában található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja, a titkosításra vonatkozó esetleges megkötések mellett.

Kelt: 2023. május 14.

.....
hallgató aláírása

Tartalomjegyzék

1. ABSZTRAKT	1
2. BEVEZETÉS	2
3. A GEOMETRIAI TÍPUSÚ ADATOK BEMUTATÁSA ÉS AZOK KEZELÉSE A TÉRINFORMATIKAI RENDSZEREKBEN	3
3.1. A térinformatikai rendszerek által használt adatok típusai.....	3
3.2. A geometriai típusú adatok bemutatása.....	4
3.3. Az adatok tárolásának módjai (általános ismertetés az adatbázisokról).....	8
4. TÉRBELI ADATOK TÁROLÁSÁRA ÉS KEZELÉSÉRE ALKALMAS RELÁCIÓSADATBÁZIS-KEZELŐ RENDSZEREK ÁTTEKINTÉSE	11
4.1. MS SQL Server (Spatial Data).....	11
4.2. PostgreSQL és PostGIS	14
4.3. SQLite és SpatiaLite	16
4.4. Oracle és Oracle Spatial and Graph.....	18
5. AZ ÜZEMI INFORMÁCIÓS RENDSZEREK BEMUTATÁSA.....	20
5.1. Az üzemi információs rendszerek általános áttekintése	20
5.2. A Lovion BIS szoftver részletes bemutatása	22
5.3. A relációsadatbázis-kezelő rendszerek Lovion BIS szoftverhez történő kapcsolásának metodikája és gyakorlata.....	27
6. A GEOMETRIAI ADATOKAT TÁMOGATÓ ADATBÁZIS-KEZELŐ RENDSZEREK FELHASZNÁLÁSÁNAK VIZSGÁLATA AZ ÜZEMI INFORMÁCIÓS RENDSZEREKBEN	30
6.1. Az egyes geometriai adatokat támogató relációsadatbázis-kezelő rendszerek összehasonlítása	30
6.2. Alkalmassági vizsgálat egy üzemi információs rendszerbe történő integrálásra vonatkozóan.....	32
6.2.1. A PostgreSQL és a PostGIS kompatibilitásának vizsgálata.....	33
6.2.2. Az SQLite és a SpatiaLite kompatibilitásának vizsgálata	36
7. EGY SPATIALITE-AL LÉTREHOZOTT ADATBÁZIS LOVION BIS RENDSZERBE TÖRTÉNŐ INTEGRÁLÁSÁNAK TESZTELÉSE.....	37
7.1. Az adatbázis létrehozása.....	37
7.2. Az adatbázis importálása a Lovion BIS rendszerbe	43
8. ÖSSZEFOGLALÁS, KONKLÚZIÓ	47
9. IRODALOMJEGYZÉK.....	49

1. ABSZTRAKT

The thesis originally aimed to present the geometrical (spatial) data-types, research (analyse) the spatial extensions of some popular object-relational databases, and explore the use of these database-systems in the “*Plant Information Management Systems*” / “*Process Intelligence Systems (PI-Systems)*”. The most significant points of the thesis can be summarized as follows:

- 1) Presentation of the handling of geometries in geoinformation systems (GIS)
- 2) Overview of geometry support of databases: *PostgreSQL* and *PostGIS*, *Oracle Spatial and Graph*, *MS SQL Server (Spatial extensions)*, *SQLite* and *SpatiaLite*
- 3) Comparison of the concepts and research of the compatibility
- 4) A prototypical implementation in the *Lovion BIS* software

This topic was offered by the German “*ITS Gruppe*” company. They have their own software, which is called *Lovion BIS*, and it is a “*Plant Information Management System*” / “*Process Intelligence System (PI-System)*”. Since 01.01.2023 I’m working with this software and I’m participating until 31.08.2023 in the training program for this system.

The thesis had more sub-results, which were the following:

- In *Part 4.*, I could compare four object-relational database systems, which support geometrical (spatial) data-types.
- In *Part 5.*, I have compiled a documentation in Hungarian from the “*Plant Information Management Systems*” / “*Process Intelligence Systems*”.
- In *Part 6.*, I have analysed the strengths and weaknesses of the various database systems tested, enabling me to select the most advantageous one.
- In *Part 7.*, I have successfully completed the creation of a custom database.

The last objective of the thesis was to implement my own build database in the *Lovion BIS* software. Unfortunately, I was unable to resolve the issue in the last step, and therefore I didn’t receive a complete result. From Spring 2024, I’m going to continue my studies at the *UniGIS* in Salzburg; and I’m going to continue and improve this same theme in my master thesis. There I’m going to employ a different approach and method.

2. BEVEZETÉS

Szakdolgozati témaválasztásom fő indoka az volt, hogy 2023. elejétől egy térinformatikai szakágazatban tevékenykedő cégnél (*GRINTEC GmbH*) kezdtem el dolgozni. A cég székhelye Grazban (Ausztria) van, de tavaly felvásárolta őket egy német cégcsoport (*ITS Gruppe*). Ennek a cégcsoportnak a fő terméke a saját maga által létrehozott és fejlesztett *Lovion BIS* üzemi információs rendszer; amely egy komplex, térinformatikai szoftver (*Smallworld GIS*) alapú program. Ennek a programnak a betanítási programján veszek részt 2023.01.01-től 2023.08.31-ig és ezt használom a napi munkavégzésem során is.

Mivel a cégcsoportnál rendelkezésre áll egy hivatalos lista, amely geoinformatikai vonatkozású szakdolgozati témákat javasol, ezért onnan választottam ki a számomra legoptimálisabbat. A kiválasztás során próbáltam arra koncentrálni, hogy a mostani napi munkafolyamataimmal is kapcsolatban legyen a téma; valamint, hogy a *geoinformatikai szakember képzés* során megszerzett tudásomat a legjobban tudjam kamatoztatni (az ott hallgatott következő tárgyak játszottak szerepet a szakdolgozati téma kiválasztása során: *Adatbázis programozás, Térinformatikai műveletek, OpenGIS műhely*).

Véleményem szerint a téma kidolgozása mellett szóló két legfontosabb érv a következő:

- Jelenleg még nem áll rendelkezésre magyar nyelvű összefoglaló szakirodalom az üzemi információs rendszerekre vonatkozóan.
- A térbeli adatok kezelésére alkalmas relációsadatbázis-kezelő rendszerek képezik az alapját az összes geoinformatikai vonatkozású (alapú) programnak és így az üzemi információs rendszereknek is. Ezért ezen adatbázis-kezelő rendszerek alapos vizsgálata és körültekintő kiválasztása nagy mértékben javíthatja az ilyen szoftverek használatát, alkalmazását és fejlesztését.

A szakdolgozat során szeretném átvenni a vektoros – geometriai – adattípusok legfontosabb tulajdonságait/jellemzőit, majd megvizsgálni, hogy ezek melyik adatbázis-kezelő rendszerben tárolhatóak és kezelhetők a legmegfelelőbbben. Ezt követően szeretném szakszerűen bemutatni az üzemi információs rendszereket és megvizsgálni, hogy melyik (vizsgált) adatbázis-kezelő rendszerrel használhatóak a leghatékonyabban. Végezetül szeretnék egy saját – geometriai adatokat tartalmazó – adatbázist építeni, amelyet integrálnék a *Lovion BIS* üzemi információs rendszerbe.

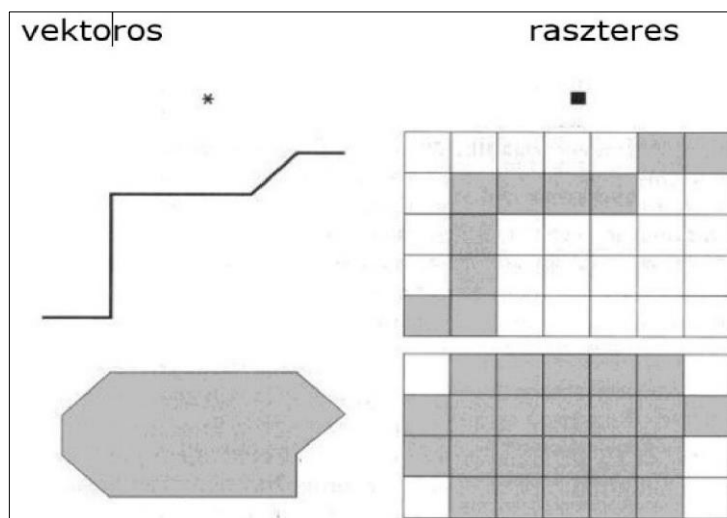
3. A GEOMETRIAI TÍPUSÚ ADATOK BEMUTATÁSA ÉS AZOK KEZELÉSE A TÉRINFORMATIKAI RENDSZEREKBEN

Ebben a részben részletesen bemutatásra kerül, hogy a térinformatikai rendszerek milyen alapvető adattípusok felhasználásával működnek és ezek a térbeli (geometriai) adatok milyen tulajdonságokkal rendelkeznek, milyen szabványok szerint működnek.

3.1. A térinformatikai rendszerek által használt adatok típusai

Alapvetően két csoportba sorolhatjuk be a térinformatikai adatokat:

- **vektoros adatok:** koordináta alapú **geometriai** formák
 - Alapformák (/alakzatok): **pont, vonal, poligon**
 - Az alapformákon felül léteznek összetett alakzatok, illetve azok változatai is (például: multipoligon, Spline, stb.).
 - Egységes modelljüket az **OGC Simple Feature Access (SFA)** szabvány definiálja (amely a következő részben kerül részletesen bemutatásra).
 - A független alakzatok mellett megadhatóak (komplex) **topológiai formák** is.
- **raszteres adatok:** képpont (pixel) alapú rácsok
 - Általában távérzékelési adatok (pl. űr- vagy légifelvételek) tartoznak ide.
 - Nincs rájuk vonatkozó egységes szabvány vagy formátum.



3.1. ábra – A vektoros (pont, vonal, poligon) és raszteres ábrázolás közötti különbség (Márkus, 2020)

A továbbiakban elsősorban a **vektoros adatokkal** fogok foglalkozni, mivel ez a csoport képezi a szakdolgozat fő vizsgálati témáját, ide tartoznak a **geometriai típusú adatok**.

3.2. A geometriai típusú adatok bemutatása

1994-ben jött létre a nemzetközi önkéntes konszenzuson alapuló szabványügyi szervezet az **OGC** (kezdetben *Open GIS*, később ***Open Geospatial Consortium***). Az OGC-ben több mint 500 kereskedelmi, kormányzati, nonprofit és kutató szervezet működik együtt konszenzusos folyamatban, amely ösztönzi a térinformatikai tartalmakra, szolgáltatásokra, adatfeldolgozásra és adatmegosztásra vonatkozó nyílt szabványok kidolgozását. (Márkus, 2020)

Az OGC által kiadott ***Simple Feature Access*** (SFA) szabvány a legáltalánosabban elfogadott modell a **vektoros geometriák** kezelésére. (Giachetta, Cserép, 2018) Sok esetben a *Simple Features for SQL Specification* elnevezéssel is találkozhatunk. Egyebek mellett a következő legfontosabb tulajdonságai jegyezhetők meg:

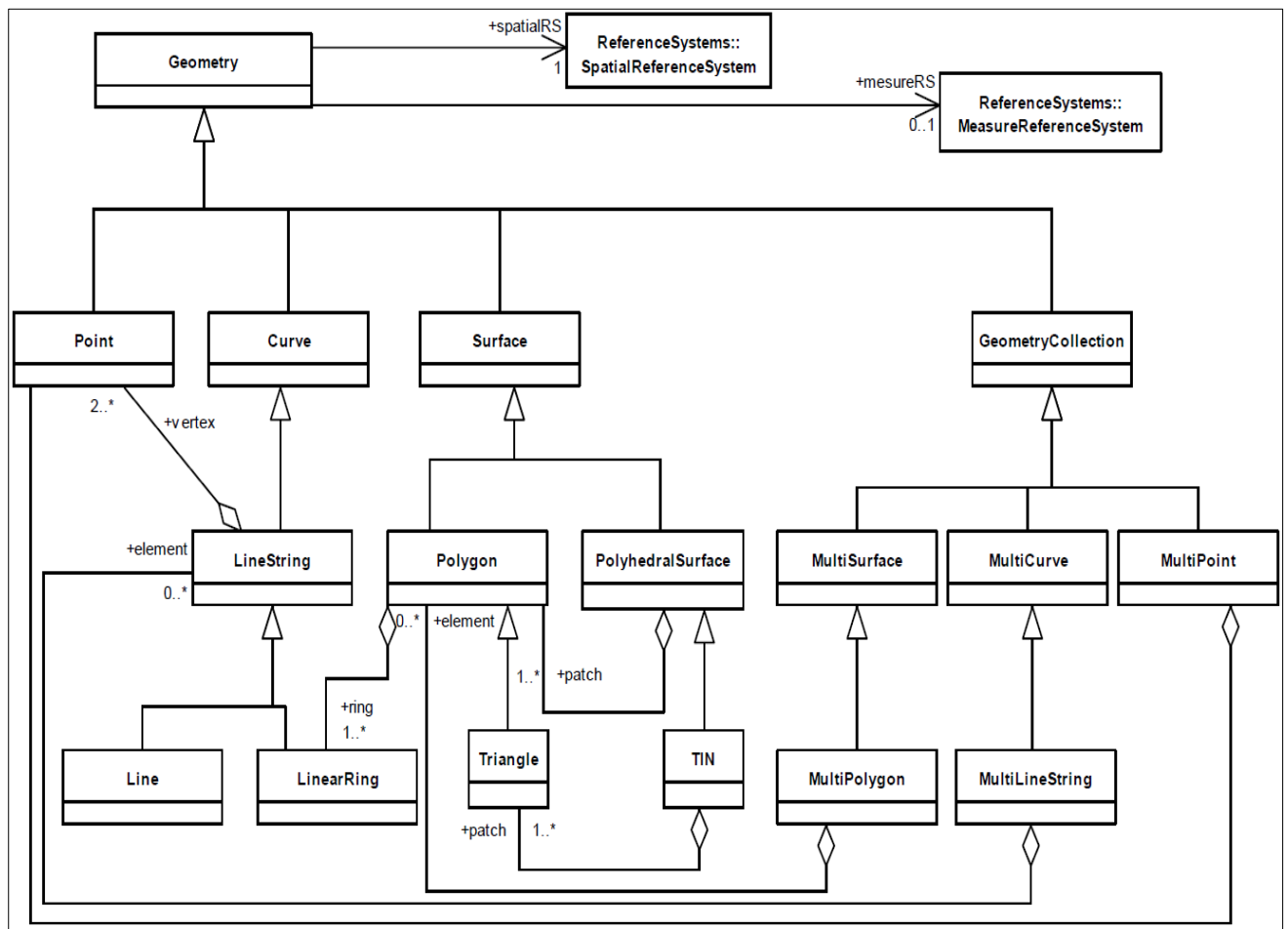
- Egyrészt definiálja a geometriák típusait, azok leírási formátumait (WKT, WKB) és a rajtuk végezhető műveleteket; másrészt pedig meghatározza a geometriák tárolási és elérési szabályait.
- Objektumorientált modell, azaz alapvetően öröklődésre épül.
- Leginkább kétdimenziós alakzatokat definiál. Háromdimenziós alakzatok esetén a műveletek kétdimenzióban kerülnek kiértékelésre.

A ***Simple Feature Access*** (SFA) szabvány két részre tagolódik:

- **OGC 06-103r4** (ISO 19125-1) – *Simple feature access - Part 1: Common architecture* (SFA-CA) → Ez a rész (-szabvány) közös architektúrát hoz létre és definiálja a feltételeket más OGC szabványokhoz. Ennek keretében meghatározza a geometriai objektumok típusait és ezen objektumok módszereit, továbbá szöveges (WKT) és bináris (WKB) formátumokat definiál az objektumok geometriájának leírására.
- **OGC 06-104r4** (ISO 19125-2) – *Simple feature access - Part 2: SQL option* (SFA-SQL) → Ez a rész egy szabványos *Structured Query Language* (SQL) sémát határoz meg, amely támogatja az objektumgyűjtemények (feature

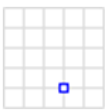
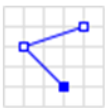
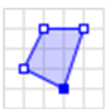
collections) tárolását, visszakeresését, lekérdezését és frissítését az *SQL Call-Level Interface* (SQL/CLI) segítségével.

Az OGC 06-103r4 (SFA-CA) különböző osztályokat (geometriai típusokat) határoz meg, amelyek mindegyike a „Geometry” absztrakt osztály alá tagozódik be. Az egyes osztályokat (illetve azok hierarchiáját) az alábbi ábra szemlélteti:

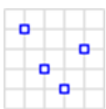
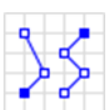
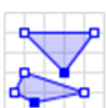
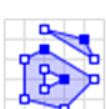


3.2. ábra – Az OGC 06-103r4 (SFA-CA) szabvány által meghatározott geometriai típusok hierarchiája (Herring, 2010)

Szintén az OGC 06-103r4 (SFA-CA) szabványrészben kerül meghatározásra az úgynevezett **Well-Known-Text** (WKT) szöveges jelölőnyelv, amely az egyes objektumok (vektoros) geometriájának konkrét leírására szolgál. Az egységes formátumnak megfelelően a szöveg az objektumtípus nagybetűs nevével kezdődik, majd egy vesszővel elválasztott, többszintű pontlistával folytatódik zárójelben. (Nagy, 2010) Az alapvető felosztás szerint megkülönböztethetünk **primitív** és az **összetett** alakzatokat (kétdimenziós ábrázolás esetén).

Típus	Alakzat	WKT leírás
Pont		POINT (30 10)
Vonallánc		LINESTRING (30 10, 10 30, 40 40)
Sokszög		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

3.1. táblázat – WKT szerinti – kétdimenziós – primitív alakzatok (forrás: <http://www.geo.hunter.cuny.edu/~ssun/R-Spatial/intro.html>)

Típus	Alakzat	WKT leírás
Ponthalmaz		MULTIPOINT ((10 40), (40 30), (20 20), (30 10))
		MULTIPOINT (10 40, 40 30, 20 20, 30 10)
Vonalláncok halmaza		MULTILINESTRING ((10 10, 20 20, 10 40), (40 40, 30 30, 40 20, 30 10))
Multipolygon		MULTIPOLYGON (((30 20, 45 40, 10 40, 30 20)), ((15 5, 40 10, 10 20, 5 10, 15 5)))
Alakzat gyűjtemény		MULTIPOLYGON (((40 40, 20 45, 45 30, 40 40)), ((20 35, 10 30, 10 10, 30 5, 45 20, 20 35), (30 20, 20 15, 20 25, 30 20)))

3.2. táblázat – WKT szerinti – kétdimenziós – összetett alakzatok (forrás: <http://www.geo.hunter.cuny.edu/~ssun/R-Spatial/intro.html>)

A *Well-Known-Text* (WKT) bináris változata a **Well-Known-Binary** (WKB) formátum, amely szintén az OGC 06-103r4 (SFA-CA) szabványrészben van definiálva. Ennek logikai felépítése megegyezik a WKT-el, de a fizikai szinten bináris adatábrázolást alkalmaz. Ennek köszönhetően tömörebb adatrepresentációra és tárolási módra ad lehetőséget; gyorsabb a feldolgozhatósága, de az ember számára nem olvasható.

Az objektumok geometriáját pontok írják le. Ezeket a pontokat két (XY, kétdimenziós), három (XYZ, háromdimenziós vagy XYM, kétdimenziós mért) vagy négy (XYZM, háromdimenziós mért) koordináta határozza meg. Az extra koordináta (M) a mért tartományt reprezentálja. (Nagy, 2010)

Bizonyos geometriai **műveletek** (függvények) is meghatározásra kerülnek a *Simple Feature Access* szabvány alapján. A szabvány(ok) alkalmazásának itt az az előnye, hogy a különböző (térbeli adatok tárolására és kezelésére alkalmas, valamint az *SFA* specifikációt implementált) relációsadatbázis-kezelő rendszerek használatakor mindig ugyanazokkal a műveletekkel találkozhatunk. (Kottyán, 2010)

A *Simple Feature Access* (SFA) szabvány alapján meghatározott függvények/műveletek általában „ST” előtaggal rendelkeznek, amely minden esetben a „standard” szóra utal.

Egy lehetséges felosztási módszer alapján a műveletek a következő csoportokat alkothatják (Kottyán, 2010):

- **Geometriai objektumok létrehozására szolgáló függvények.** → Ezekkel a függvényekkel geometriai objektumokat hozhatunk létre *Well-Known-Text* (WKT) vagy *Well-Known-Binary* (WKB) formátumú adatok alapján.
- **Geometriai objektumok kezelését biztosító függvények.** → Az ide tartozó függvényekkel lehetőség van a meglévő geometriák módosítására, illetve új geometriák előállítására is. Ilyen műveletek lehetnek például:
 - az övezetgenerálás (*buffer*), konvex burok generálása (*convexHull*),
 - kettő geometriai objektum metszése (*intersection*), uniója (*union*), különbsége (*difference*), szimmetrikus különbsége (*symDifference*),
 - egy geometriai objektum tetszőleges pontjának lekérdezése, stb.
- **Geometriai objektumok közötti kapcsolatok meghatározására alkalmas függvények.** → Az egyes objektumok közötti kapcsolatok lekérdezésére vonatkozó műveletek tartoznak ide. Bizonyos esetekben logikai válaszokat (igaz/hamis) kaphatunk a kiszolgálótól, más esetekben pedig konkrét számértékeket. Többek között a következő lekérdezési függvények állnak rendelkezésre az egyes objektumok kapcsolataira vonatkozóan: egyenlő (*equals*), különálló (*disjoint*), keresztezi (*crosses*), érinti (*touches*), belül (*within*), átfedi (*overlaps*), metszi (*intersects*): belül inverze, tartalmaz (*contains*): különálló inverze, távolság (*distance*), stb.

- **Geometriai objektumok tulajdonságainak lekérdezését lehetővé tevő függvények.** → A *Simple Feature Access* (SFA) szabvány által meghatározott geometriai típusoknak megfelelően, az objektumok számos jellemzőjének lekérdezésére van lehetőség. Például:
 - objektum geometriai típusa (*geometryType*),
 - objektumok pontjainak jellemzői (*nPoints*), pontok koordinátái (*m*),
 - objektum dimenziójának (*dimension*), területének (*area*), hosszának (*length*) meghatározása,
 - objektum befoglaló területének meghatározása (*envelope*),
 - objektum határvonalainak meghatározása (*boundary*),
 - zárttság, üresség (*isEmpty*), egyszerűség (*isSimple*) lekérdezése,
 - alkalmazott vetületi rendszer lekérdezése (*SRID*), stb.

3.3. Az adatok tárolásának módjai (általános ismertetés az adatbázisokról)

A geometriai típusú (vektoros) adatok adatbázisokban történő tárolásának alapelvei, alapvető tulajdonságai a következőképpen foglalhatóak össze (Nagy, 2019):

- Egy adott objektum geometriájának leírására szolgáló adato(ka)t, mindig az adatbázis egy (vagy több) meghatározott mezőjében tároljuk.
- Ezeket az adatokat különböző – az 3.2-es részben már bemutatott – függvényekkel (műveletekkel) tudjuk kezelni az adatbázison belül (is).
- A geometriai adatok ilyen jellegű tárolásánál térbeli indexelésre is van lehetőség.
- Továbbá lehetőség nyílik a topológiai információk tárolására is, valamint adott esetben raszteres adatok is kezelhetők.

A *Simple Feature Access*, az OGC 06-104r4 (SFA-SQL) résszabvány alapján meghatározza az adatbázis-kezelő rendszerekhez történő kapcsolódásokat (SQL/MM) is. Sok esetben az adott **relációsadatbázis-kezelő rendszerek** egyes **kiterjesztései** (vagy kiegészítő moduljai, változatai, stb.) alkalmasak a térbeli adatok tárolására, kezelésére;

de léteznek olyan esetek is, amikor ezek a funkciók a „**törzsrendszerekbe**” (azaz magukba az alap adatbázis-kezelő rendszerekbe) vannak integrálva:

- MonetDB → **MonetDB – GIS extension** (kiterjesztés)
- PostgreSQL → **PostGIS** (kiterjesztés)
- SQLite → **Spatialite** (kiterjesztés/változat)
- Oracle → **Oracle Spatial and Graph** (kiegészítő modul)
- IBM Db2 → **IBM Db2 Spatial Extender** (kiterjesztés) és **IBM Informix Spatial DataBlade** (kiegészítő modul)
- MySQL → Itt egy „integrált kiterjesztéssel” – **MySQL Spatial Extension** – nyílik lehetőség a térbeli adatok tárolására és kezelésére.
- **Microsoft SQL Server** → 2008 óta (SQL Server 2008 kiadásától kezdve) a Microsoft SQL Server önállóan képes a térbeli adatok tárolására és kezelésére.
- **SAP HANA** → Ezen adatbázis-kezelő rendszer esetében egy integrált szolgáltatáscsomag – **SAP HANA spatial services** – segítségével lehetséges tárolni és kezelni a térbeli adatokat.

A továbbiakban ezek közül csak négyet – **PostGIS, Spatialite, Oracle Spatial and Graph, Microsoft SQL Server** – fogok részletesebben megvizsgálni, mivel a szakdolgozati témakiírás alapján ezeket szükséges összehasonlítani.



3.3. ábra – A PostGIS, Spatialite, Oracle Spatial and Graph, Microsoft SQL Server relációsadatbázis-kezelő rendszerek logói (forrás: cégek weboldalai)

A következő három formátum bemutatása feltétlenül szükséges a térbeli adatok adatbázisokban való tárolásának megértéséhez. Többek között ezen formátumok segítségével valósítható meg az adatok cseréje, továbbítása és biztosított azok feldolgozhatósága is. Ezek a következők:

- **GeoJSON** formátum:

- JSON (*JavaScript Object Notation*) alapú formátum, amely földrajzi jellemzők és azok nem térbeli attribútumainak megjelenítésére alkalmas. Végződése: **.geojson**
- A *Simple Feature Access* (SFA) szabványhoz illeszkedik és alapvetően a térinformatikai adatcsere céljából került kifejlesztésre.
- Széles körben használható, többek között a következő szoftverek, alkalmazások által: *OpenLayers*, *Leaflet*, *MapServer*, *GeoServer*, *GeoDjango*, *PointPlot*, *GDAL*, *QGIS*, *CartoDB*, *ArcGIS*, stb.
- API-n (alkalmazásprogramozási interfészek) keresztül használható a *Bing Maps*, a *Yahoo!* és a *Google Maps* alkalmazásokban is.

- **GML** (*Geography Markup Language*) formátum:

- Az OGC (*Open Geospatial Consortium*) által kifejlesztett, XML (*Extensible Markup Language*) alapú jelölőnyelv. Végződése: **.gml**
- Az XML jelölőnyelvhez hasonlóan, eltérő rendszerek közötti adatcserére szolgál; egészen pontosan a különböző földrajzi információs rendszerek közötti adatcserét teszi lehetővé.

- **KML** (*Keyhole Markup Language*) formátum:

- Kifejezetten a *Google* alkalmazásai (*Google Earth*, *Google Maps*) számára kifejlesztett, a térben ábrázolt alakzatok megjelenítésére alkalmas jelölőnyelv. (Korlátozottan más alkalmazásokhoz is felhasználható.) Végződései: **.kml**, **.kmz**
- Szintén az OGC (*Open Geospatial Consortium*) által szabályozott és szintén XML (*Extensible Markup Language*) alapú megoldás.

A GML és a KML formátumok közötti legfőbb különbség: Míg a **GML** formátum a **földrajzi vonatkozású** – térbeli – **tartalom modellezésére** és/vagy **kódolására** szolgáló jelölőnyelv, addig a **KML** formátum ugyanezen tartalom **vizuális megjelenítésére** alkalmas.

4. TÉRBELI ADATOK TÁROLÁSÁRA ÉS KEZELÉSÉRE ALKALMAS RELÁCIÓSADATBÁZIS-KEZELŐ RENDSZEREK ÁTTEKINTÉSE

A népszerűbb relációsadatbázis-kezelő rendszerek szinte mindegyike rendelkezik olyan kiterjesztéssel vagy kiegészítő csomaggal (plusz modullal), amelyek valamilyen formában támogatják a térbeli – illetve ennél fogva a geometriai – adatok tárolását és kezelését is. Ezen relációsadatbázis-kezelő rendszerek közül a szakdolgozati téma szempontjából legfontosabb – a 3.3. pontban felsorolt – négyet (*PostGIS*, *SpatiaLite*, *Oracle Spatial and Graph*, *Microsoft SQL Server*) szeretném részletesen bemutatni; valamint megvizsgálni azt, hogy melyik a legalkalmasabb a *Lovion BIS* szoftverben történő felhasználásra, integrálásra.

4.1. MS SQL Server (Spatial Data)

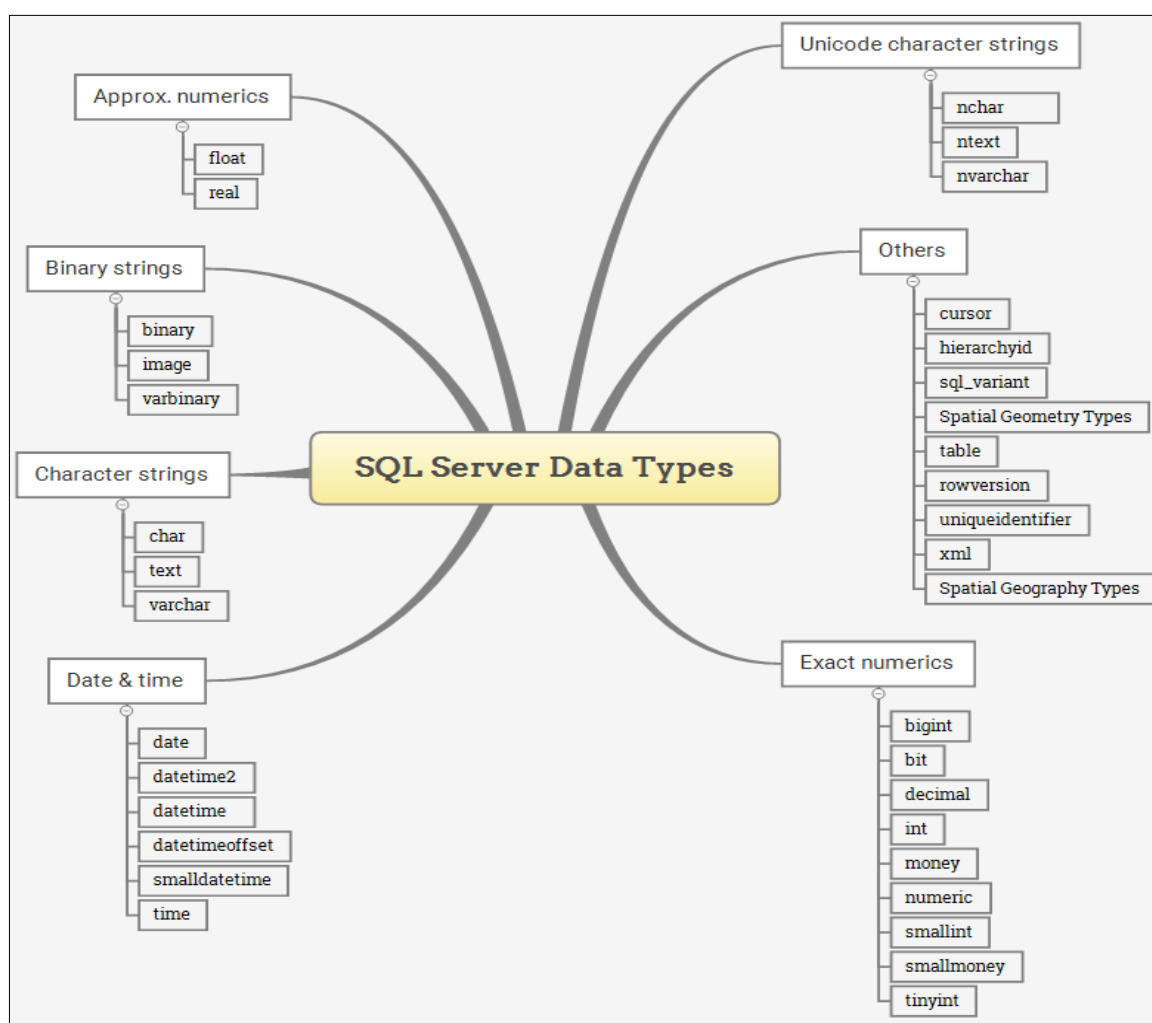
A *Lovion BIS* szoftver felhasználói köre szinte teljes egészében közüzemi szolgáltatókra terjed ki. (Erről később fogok részletesebben írni az 5.2-es részben.) A felhasználók nagyjából 85-90 %-a jelenleg az *MS SQL Server* relációsadatbázis-kezelő rendszer valamelyik kiadását (*MS SQL Server 2017*, *MS SQL Server 2019*, stb.) használja a szoftverben kezelt adataik tárolására. Ez azt jelenti, hogy ez a relációsadatbázis-kezelő rendszer van hozzákapcsolva a személyre szabott üzemi információs szoftverhez. Ennek megfelelően első körben ezt az *MS SQL Server*t szeretném bemutatni, megvizsgálni.

Az *MS SQL Server* a Microsoft által fejlesztett **kereskedelmi** célú relációsadatbázis-kezelő rendszer, amelyet kifejezetten a Windows operációs rendszerekhez terveztek. Első verziója – *SQL Server 1.0* néven – 1989-ben került kiadásra. Ezt követően a 2008 augusztusában megjelent *SQL Server 2008* kiadásig tizennégy további verzió jelent meg. Az ***SQL Server 2008*** az első kiadás volt, amely megoldást kínált a **térbeli adatok tárolására és kezelésére**. Két évvel később ezt a kiadást továbbfejlesztették és megjelent az *SQL Server 2008 R2* (az *SQL Azure* után), amely számos további újítást tartalmazott a Microsoft alkalmazások közötti integrációkra, illetve az egyes a kiszolgáltítótípusok adminisztrálására vonatkozóan. (Kottyán, 2010)

Az *SQL Server 2008 R2* óta a következő további verziók jelentek meg, amelyek elnevezése minden esetben jelzi a kiadás évszámát is: *SQL Server 2012*, *SQL Server 2014*, *SQL Server 2016*, *SQL Server 2017*, *SQL Server 2019*, *SQL Server 2022*

Ezen további verziók mindegyike képes már a térbeli adatok tárolására és kezelésére. A térbeli adatokon végezhető műveletek köre folyamatosan fejlesztésre került az újabb verziók kiadása során. Az utolsó számottevő fejlesztés ezen a téren az *SQL Server 2019*-ben történt; ahol a fejlesztőkörnyezet funkcióin végeztek javításokat, a grafikus és térbeli adattípusokhoz kapcsolódóan.

Az a legújabb verzió, azaz az *SQL Server 2022* jelenleg a következő adattípusok kezelésére alkalmas:



4.1. ábra – Az *SQL Server 2022* által kezelt adattípusok csoportjai (forrás: www.sqlservertutorial.net/sql-server-basics/sql-server-data-types/)

A fenti ábrán szereplő adattípusok közül a következő kettő a legfontosabb a jelenlegi szakdolgozat (vizsgálati téma) szempontjából:

- ***Spatial Geometry Types***: Ez a formátum használható az adatok **sík** (euklideszi) **koordináta-rendszerben** történő ábrázolásához, megjelenítéséhez kapcsolódóan.
- ***Spatial Geography Types***: Ez a formátum használható az **ellipszoidi** (geodéziai) **koordináta-rendszer** adatainak tárolására. Ilyenek lehetnek például a GPS/GNSS (vagy egyéb műholdas) rendszerek által meghatározott szélességi és hosszúsági koordináták (adatok).

A geometriai (mind a földrajzi) adattípusok itt is az *OGC 06-103r4 (SFA-CA)* szabványhoz illeszkednek, azaz az ott meghatározott típusok tárolására és kezelésére van lehetőség.

Az *SQL Server* verziói több különböző kiadásban készültek, ezek közül az úgynevezett „*Express*” kiadások azok, amelyek ingyenesen hozzáférhetőek és használhatóak.

Az *SQL Server* 2022-ben használható legfontosabb térbeli – geometriai (*Spatial Geometry Types*) és földrajzi (*Spatial Geography Types*) típusú adatokon végezhető – műveletek listái (illetve azok ismertetései) a következő weboldalakon érhetőek el (forrás: *learn.microsoft.com*):

- OGC (*Open Geospatial Consortium*) által meghatározott **standard műveletek**: <https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/ogc-methods-on-geometry-instances?view=sql-server-ver16>
- OGC által meghatározott **standard műveletek kiterjesztései**: <https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/extended-methods-on-geometry-instances?view=sql-server-ver16>
- OGC által meghatározott **statikus műveletek**: <https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/ogc-static-geometry-methods?view=sql-server-ver16>
- OGC által meghatározott **statikus műveletek kiterjesztései**: <https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/extended-static-geometry-methods?view=sql-server-ver16>

- Statikus „aggregátum” műveletek: <https://learn.microsoft.com/en-us/sql/t-sql/spatial-geometry/static-aggregate-geometry-methods?view=sql-server-ver16>

Itt a geometriai adatokhoz (*Sp. Geometry T.*) kapcsolódó listákat soroltam fel, de ugyanezek a műveletek végezhetőek a földrajzi típusú (*Sp. Geography T.*) adatokon is.

Az *SQL Server* adminisztrációjához, konfigurációjához és egyéb menedzsmenti feladataihoz jelenleg használható szoftveralkalmazás az **SQL Server Management Studio BI** (SSMS). Szintén a Microsoft fejlesztése és az *SQL Server 2005*-tel együttesen került először kiadásra. (Ezt megelőzően a *Enterprise Manager* látta el ezeket a feladatokat.) Az *SQL Server* mellett, az *Azure SQL* relációsadatbázis-kezelő rendszer kezelésére is alkalmas (legfrissebb verziója: 19.0.2.). (forrás: *learn.microsoft.com*)

4.2. PostgreSQL és PostGIS

A *PostgreSQL* egy olyan **nyílt forráskódú** relációsadatbázis-kezelő rendszer, amelynek a fejlesztését nagyrészt önkéntesek végzik. SQL implementációja szinte teljes egészében illeszkedik az SQL-szabványokhoz, teljes mértékben **ACID** (atomicity – atomicitás, consistency – konzisztencia, isolation – izoláció, durability – tartósság) **kompatibilis** és a népszerűbb operációs rendszerek (*Linux*, *UNIX*, *Windows*) mindegyikével használható. (Kottyán, 2010)

1989-ben jelent meg első verziója *Postgres 1.0* néven és egészen 1996-ig a kaliforniai *Berkley Egyetemen* munkatársai és hallgatói fejlesztették. Az első négy verzió az egyetemi projekt keretében került kifejlesztésre, majd a *Global Development Group* vette szárnyai alá az adatbázis-kezelő rendszer további verzióinak kidolgozását. A *PostgreSQL* név a hatodik (6.0) verziótól kezdődően létezik. A legutolsó stabil verziója a 15.2-es, amely 2023. február 9-én került kiadásra. (Kottyán, 2010)

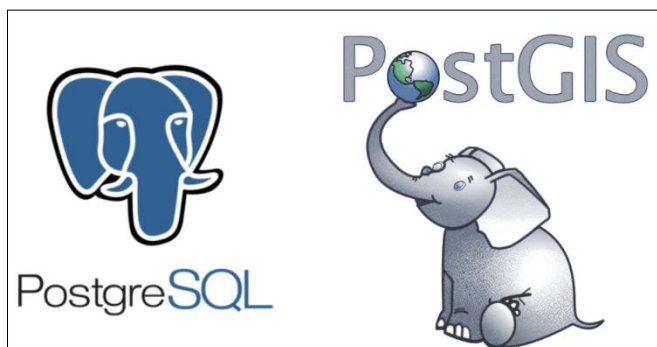
A *PostgreSQL* kiegészítő csomagja a **PostGIS**, amely a *Simple Features for SQL (SFS)* modell alapján működő bővítmény és a *Refractions Research* fejleszti. Ez a kiterjesztés alkalmas a térbeli adatok tárolására és kezelésére.

A *PostGIS* főbb tulajdonságai a következők (Nagy, 2010):

- Teljes mértékben illeszkedik az OGC két legfontosabb szabványához: *OGC 06-103r4*, *OGC 06-104r4*

- Jelenleg a 3.3.2-es verzió a legfrissebb kiadása.
- A *PostGIS* további nyílt forráskódú szoftvereket használ a hatékonyabb működés érdekében:
 - **GEOS** (Geometry Engine, Open Source) a térinformatikai műveletekhez
 - **PROJ.4** a reprojekcióhoz
- A *PostGIS* rendelkezik az *OGC 06-103r4* és az *OGC 06-104r4* szabványokban meghatározott műveletekkel, de ezen felül további függvények elvégzésére is lehetőséget kínál.
- A *PostGIS*-ben alkalmazható műveletek listája, illetve azok írásai a 3.3-as verziókhoz tartozó használati utasításban találhatóak meg. A következő weboldalon érhető el: <https://postgis.net/docs/>
- A függvények (műveletek) és az aggregált függvények mindig „ST_” előtaggal kezdődnek (ST_SRID). Ez az előtag néhány esetben elhagyható, de van, amikor a megkülönböztetés miatt mindenképpen szükséges használni (pl.: ST_UNION).
- A *PostGIS*-ben is elkülönülnek a földrajzi (*geography type*) és a geometriai (*geometry type*) adattípusok, ám ezek egymással nagyon hasonlóak. A „*geography type*” objektumai egy gömbön vagy egy (forgási) ellipszoidon vannak meghatározva. Ez azonban különbözik a gömbi vonatkoztatási rendszerű geometria típustól, mivel az sík topológiai rendszert használ, „*geography type*” viszont szférikus.

A *PostgreSQL* és a *PostGIS* rendszerek kezeléséhez és adminisztrációjához a legjobban használható kliens alkalmazás a **pgAdmin 4**, amelynek legfrissebb változata a 2023. május 4-én kiadott 7.1-es verzió.



4.2. ábra – A *PostgreSQL* és a *PostGIS* logói (forrás: termékek weboldalai)

4.3. SQLite és SpatiaLite

Az **SQLite** – a *ProstgreSQL*-hez hasonlóan – szintén egy **nyílt forráskódú** (opensource) relációsadatbázis-kezelő rendszer. Fő jellemzője az **egyszerűség**, mivel kis méretű és állomány alapú adatbázis-kezelési lehetőséget nyújt. Ez a gyakorlatban azt jelent, hogy egy adatbázis egy állománynak (fájlnak) felel meg. Használatához nincs szükség hálózati elérésre, nem rendelkezik felhasználói adatbázissal és jogosultságkezelése sincsen. A legtöbb relációsadatbázis-kezelő rendszerhez hasonlóan **ACID kompatibilis**.

Az *SQLite* a legtöbb programozási nyelvvel kompatibilis (*BASIC, C, C++, Common Lisp, Java, C#, Visual Basic, .NET, Delphi, Curl, Lua, Tcl, R, PHP, Perl, Ruby, Objective-C, Python, newLisp, Haskell, OCaml, Smalltalk, Scheme, Javascript, VBScript*). C forrású programkönyvtárként került megvalósításra és legjobban az SQL-92 szabványra hasonlít a nyelve. (Blahota, 2013)

Az *SQLite* egy beágyazható adatbázis-kezelő rendszer, fontos tulajdonsága, hogy a gazdaprogram részeként működik (a hozzálinkelt programkönyvtár révén) és nem különálló folyamatként. Erre lehet példa többek között a *QGIS* szoftver. (Blahota, 2013)

Az adatbázis-kezelő rendszer első verziója 2000-ben jelent meg és D. Richard Hipp tervezte, illetve programozta. Napjainkban egy nemzetközi csapat foglalkozik a fejlesztéssel és a szakmai támogatással. Legújabb verziója a **3.41.2-es**, amely 2023. március 22-én került kiadásra.

A **SpatiaLite** az *SQLite* rendszernek egy olyan kiterjesztése, amely alkalmas a **vektoros téradatok** (geometriai adatok) **tárolására**, továbbá segítségével lehetőség van **geometriai adatokkal történő műveletek** elvégzésére is. Gyakorlatban a *SpatiaLite* az *SQLite* *OGC* szabványok szerinti ajánlásokkal bővített kiegészítője.



4.3. ábra – Az SQLite és a SpatiaLite logói (forrás: termékek weboldalai)

A **SpatiaLite** legfontosabb tulajdonságai a következőkben foglalhatóak össze:

- 2008 óta használható az *SQLite* kiegészítőjeként. Jelenleg az 5.0.1-es verzió a legfrissebb, ami 2021. február 7-én jelent meg.
- Az *OGC 06-103r4* és az *OGC 06-104r4* szabványokhoz illeszkedik, WKT és WKB formátumokat is használ.
- A *PostGIS*-hez hasonlóan további nyílt forráskódú szoftvereket használ a hatékonyabb működés érdekében:
 - Bizonyos térbeli műveletek (mint pl.: Overlaps, Touches, stb.) a **GEOS** (*Geometry Engine, Open Source*) programkönyvtár alapján működnek.
 - A koordináta leképezések (vetületi számítások) a **PROJ.4** programkönyvtárra alapulnak.
 - Az eltérő kódolású karaktersorozatok (string-ek) átkonvertálásához **iconv** programot használ.
- Képes *Shape* fájlok (.shp) importálására és exportálására.
- A *VirtualShape* és a *VirtualText* bővítmények lehetőséget adnak külső *Shape* fájlok (.shp) és CSV fájlok (.csv) használatára anélkül, hogy azokat importálni kelljen a programba.
- A legtöbb térinformatikai szoftverrel kompatibilis a *SpatiaLite*, úgymint például a következők: *ArcGIS*, *QGIS*, *AutocadMap*, *Global Mapper*, *GeoDjango*, stb.

A **SpatiaLite** is használ „ST_” prefixumot egyes térbeli műveletek megnevezésénél, ám ez nem minden függvény esetében igaz. Eltérő megnevezések vannak, az egyes csoportoknak megfelelően. A *SpatiaLite* 5.0.1-es verzió műveletekeinek teljekörű listája a következő weboldalon található meg: <https://www.gaia-gis.it/gaia-sins/spatialite-sql-5.0.1.html>

A **SpatiaLite GUI** egy grafikus felhasználói interfész (*Graphic User Interface*), amely az *SQLite* és a *SpatiaLite* kezelésére is egyaránt alkalmas. A szoftver legfrissebb verziója a 2.1.0-beta1, amely 2020. augusztus 24-én jelent meg.

4.4. Oracle és Oracle Spatial and Graph

Az **Oracle DB** a világ egyik legnépszerűbb és legrégebbi relációsadatbázis-kezelő rendszere, amely az előző kettővel ellentétben nem nyílt forráskódú. Az *Oracle Corporation* fejleszti az 1979-es első megjelenése óta és jelenleg a – 2020. decemberében megjelent – *21c* a legfrissebb stabil verziója. Az előző verzió a *19c*, amely egyelőre még több kiadásban (*SE2*, *EE*, stb.) érhető el, mint a *21c*. (Nagy, 2010)

Az *Oracle* adatbázis-kezelő rendszerekben relációs adatok és objektumorientált adatok egyaránt tarolhatóak és kezelhetőek. Az összes népszerűbb operációs rendszer (Windows, Linux, UNIX, stb.) támogatja a használatát. Létezik egy ingyenes kiadása is, az *XE* (*Express-Edition*), amelynek a funkció erősen korlátozottak. Emellett oktatási és tanulási célokra biztosítanak ingyenes/kedvezményes licenzeket is.

Az SQL nyelv mellé az *Oracle Corporation* kifejlesztette a **PL/SQL** (*Procedural Language*) nevű saját procedurális programozási nyelvét. A kiterjesztés alapja az *Ada* programnyelv és ennek segítségével képes a felhasználó tárolt eljárásokat (pl. triggereket) írni. A PL/SQL természetesen a normál SQL nyelvet is magába foglalja. (forrás: <https://docs.oracle.com/>)

Az *Oracle*-ben lehetőség van XML típusú adatstruktúrák (XMLDB, XDK) tárolására is. Továbbá nagyon sok nem-relációs adatformátum (pl. képek, videók, zenék, stb.) indexelése támogatott az, ezeknek tárolását BLOB-ok (*binary large objects*) és CLOB-ok (*character large objects*) támogatják.

Az **Oracle Spatial and Graph** (2012-ig *Oracle Spatial*) egy **kiegészítő komponens**, amellyel lehetőség nyílik a térbeli adatok tárolására és kezelésére. 2019-ig egy teljesen külön licenszhez kapcsolódó összetevő volt, ám azóta az *SE2* (*Standard Edition 2*) és az *EE* (*Enterprise Edition*) kiadások alapkomponense.

A földrajzi és geometriai koordináták tárolása mellett, az *Oracle Spatial and Graph* támogatja – többek között – **gráfok** létrehozását is például közlekedési- vagy közösségi-hálózatok esetében. Útvonaltervezési szoftverek fejlesztéséhez is kiválóan alkalmazható, továbbá széles körben felhasználható például létesítménygazdálkodási vagy különböző helyzetmeghatározási célokra is. Ennek megfelelően a termék célágazatai/célcsoportjai – nagy vonalakban – a következők: közüzemi szolgáltatók, hulladékkezeléssel foglalkozó

szolgáltatók, szállítmányozási és logisztikai szolgáltatók, telekommunikációs szolgáltatók, területhasznosítással és -fejlesztéssel foglalkozó szolgáltatók, épületüzemeltetési szolgáltatók, direkt-marketinggel és értékesítéssel foglalkozó szolgáltatók, stb. (forrás: <https://docs.oracle.com/>)



4.4. ábra – Az Oracle DB és az Oracle Spatial and Graph logói (forrás: termékek weboldalai)

Az *Oracle Spatial and Graph* struktúrája a következő összetevőkből áll (forrás: <https://docs.oracle.com/>):

- MDSYS séma, amely meghatározza a támogatott geometriai adattípusok tárolását és szemantikájára vonatkozó szabályokat,
- Mechanizmus a térbeli indexelésre vonatkozóan,
- Függvények, operátorok az egyes műveletek végrehajtásához,
- Topológiai szabályok,
- *GeoRaster* funkció, amely lehetővé teszi raszteres adatok és a hozzájuk kapcsolódó metaadatok tárolását, indexelését, lekérdezését és elemzését.

Az *Oracle Spatial and Graph* két legfontosabb gráf típusa a következő:

- **Network Data Model** → Olyan gráf modell, amely logikai és fizikai hálózatok elemzésére is alkalmas, így kiválóan hasznosítható például a szállítás, a logisztika vagy a közüzemi szolgáltatások területén.
- **RDF semantic** → Ez a gráf modell támogatja a World Wide Web (W3C) RDF által létrehozott szabványokat, így nagyon sok esetben (pl. szemantikus adatintegrációt, közösségi hálózatelemzés, stb.) specifikusan használható.

5. AZ ÜZEMI INFORMÁCIÓS RENDSZEREK BEMUTATÁSA

Mivel magyar nyelven nem – idegen nyelven (angolul, németül) pedig csak korlátozottan – áll rendelkezésre teljeskörű leírás, vagy egyéb ismertető anyag az **üzemi információs rendszerekre** vonatkozóan, ezért ebben a pontban már leginkább **saját munkát**, saját magam által gyűjtött és összeírt információkat fogok közölni.

Mint azt a bevezetésben (2. pont) említettem, 2023. január 1. óta dolgozok a *GRINTEC GmbH* cégnél, amelynek a többségi tulajdonosa az *ITS Gruppe* német cégcsoport. Az *ITS Gruppe* saját terméke a *Lovion BIS* üzemi információs rendszer, amellyel ez év eleje óta dolgozok, illetve amelynek a betanítási programján veszek részt 2023. augusztus 31-ig.

Ennek megfelelően a saját munka itt a gyakorlatban azt jelenti, hogy az eddig megszerzett ismereteim, valamint tapasztalataim alapján próbálom meg összeállítani a következő három pontot (5.1., 5.2., 5.3.). Céлом, hogy egy olyan összefoglaló jellegű leírást (dokumentumot) készítsek, amely elegendő információt, adatot tartalmaz az üzemi információs rendszerekre vonatkozóan.

5.1. Az üzemi információs rendszerek általános áttekintése

Az **üzemi információs rendszereket** németül „*Betriebinformationssystem* (BIS)” névvel illetik. Többféle ágazathoz kapcsolódhatnak, azaz többféle szektor igényeinek megfelelően, eltérő folyamatokhoz lehetnek optimalizálva. Ilyenek lehetnek (többek között) például: a kereskedelmi szolgáltatások, a szállítási és logisztikai szolgáltatások, a mező- és erdőgazdálkodás, a gyártás, a **közüemi szolgáltatások**, stb.

Az angol elnevezés nem ennyire egyértelmű, mivel ott eltérő rövidítésekkel és eltérő definíciókkal találkozhatunk, amelyek egymással átfedésben állnak és ezen típusú üzemi információs rendszerek egy részét vagy egészét fedik le:

- BIS → *Bussiness Information System*: üzleti információs rendszer
- EIS → *Enterprise Information System*: vállalati információs rendszer
- PIMS → *Plant Information Management System*: üzemi információs irányítási rendszer
- PI-System → *Process Intelligence System*: intelligens folyamatirányítási rendszer

Az **üzemi információs rendszerek** komplex „szervezet- és szolgáltatás-irányításra” alkalmas rendszerek, amelyek több folyamatot fednek le, mint például a következők: munkafolyamat irányítás, készletgazdálkodás, humán-erőforrás gazdálkodás, infrastruktúra-üzemeltetés és -karbantartás, beruházás-tervezés és -kivitelezés, stb.

Az üzemi információs rendszerek egyik alapvető tulajdonsága, hogy több másik program/rendszer integrálásán alapulnak. Ez a gyakorlatban azt jelenti, hogy a törzsmódul mellé integrálva vannak vállaltirányítási rendszerek, **térinformatikai rendszerek**, dokumentumkezelő rendszerek, stb. Ezen felül minden üzemi információs rendszer egy vagy több adatbázis-kezelő rendszerrel képes kapcsolatot létesíteni. Vannak persze olyan üzemi információs rendszerek is, ahol az adatok egy saját belső adatbázisban vannak tárolva, de a legtöbb esetben egy külső adatbázis hozzákapsolódóval valósul meg az adatok tárolása, rendezése és kezelése.

Ilyen üzemi információs rendszerek – amelyek leginkább vagy kifejezetten közüzemi szolgáltatók számára kerültek kifejlesztésre – például a következők:

- Termék/szoftver neve: **AM Suite** – Forgalmazó neve: *Mettenmeier GmbH* (Németország) – Termék/szoftver weboldala: <https://www.mettenmeier.de/produkte-loesungen/am-suite/>
- Termék/szoftver neve: **AVEVA PI-System** – Forgalmazó neve: *AVEVA Group plc* (Egyesült Királyság) – Termék/szoftver weboldala: <https://www.aveva.com/de-de/industries/power-utilities/>
- Termék/szoftver neve: **Lovion BIS** – Forgalmazó neve: *ITS Gruppe cégcsoport – Lovion GmbH* (Németország) – Termék/szoftver weboldala: <https://lovion.de/en/lovion-main-page-en>

A gyakorlati megközelítés érdekében az üzemi információs rendszerek további jellemzőit, tulajdonságait a **Lovion BIS programon** keresztül fogom bemutatni. Ezt a terméket ismerem a legközelebbiről és mivel ez egy nagyon komplex rendszer, ezért úgy gondolom, hogy teljes mértékben alkalmas az üzemi információs rendszerek minden fontos részletének feltárására és bemutatására.

5.2. A Lovion BIS szoftver részletes bemutatása

Mint az fentebb már említésre került a *Lovion BIS* szoftver az *ITS Gruppe* nevű német cégcsoport terméke. A cégcsoport első vállalkozása 1989-ben alakult és az elmúlt több, mint 30 év alatt egy multinacionális nagyvállalattá nőtte ki magát. Jelenleg öt országban (Németország, Ausztria, Lengyelország, Hollandia, Luxemburg), 19 telephelyet működtet és összesen több, mint 400 alkalmazottat foglalkoztat.



5.1. ábra – Az ITS Gruppe cégcsoport és a Lovion BIS szoftver logói
(forrás: <https://its-service.de>)

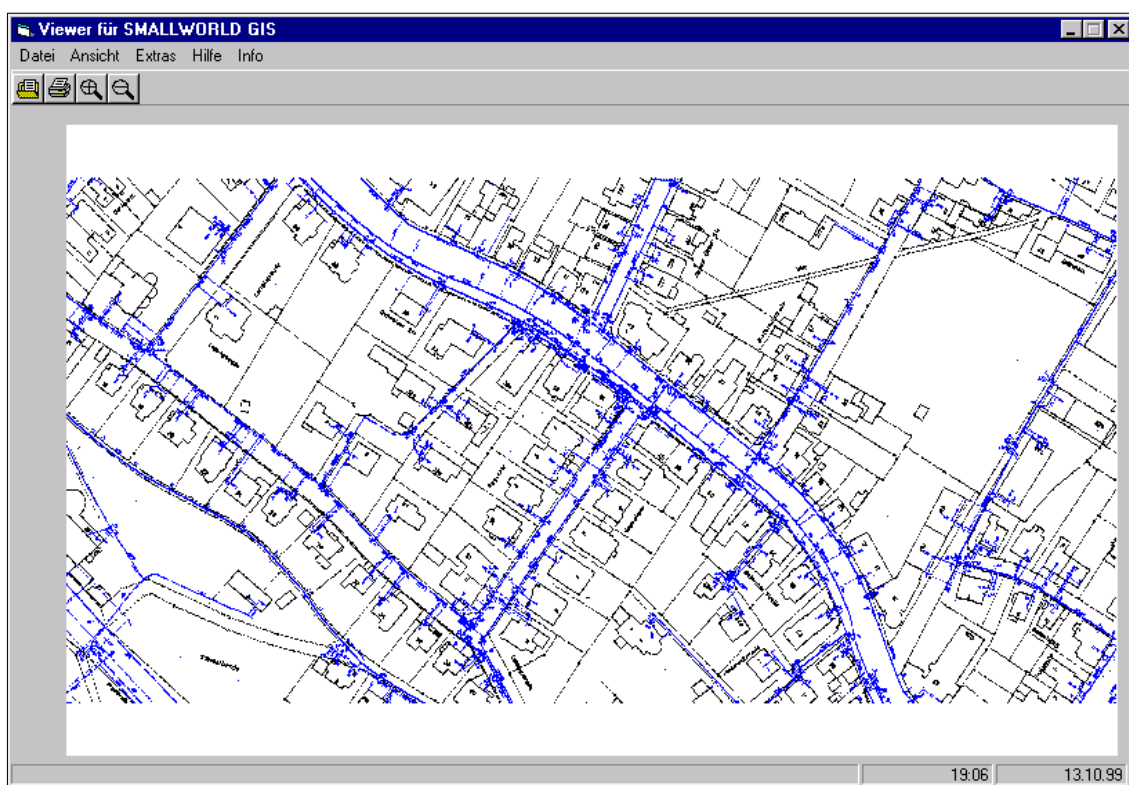
A teljes cégcsoportnak több mint 200 ügyfele van Európa szerte, amelyeknek nagyjából 95 %-át közüzemi infrastruktúra fenntartásáért felelős szolgáltatók (német nyelvterületen: „*Netzbetreiber*”) és települési infrastruktúra üzemeltetéséért felelős vállalatok (német nyelvterületen: „*Stadtwerke*”) teszik ki. Ebből egyértelműen látható, hogy a ***Lovion BIS* szoftver a közüzemi szolgáltatási szektor szereplői számára került kifejlesztésre**, minden ide tartozó szektort számításba véve: áram, gáz, vezetékes ivóvíz, csatorna, távfűtés, telekommunikáció, épületüzemeltetés, stb. (A *Lovion BIS* programon felül a cégcsoportnak vannak még más, kisebb volumenű termékei is.)

Az előző bekezdésben leírtak teljeskörű megértése céljából szükséges tisztázni, hogy Németországban és Ausztriában a közüzemi szolgáltatások ellátásáért felelős szolgáltatók köre két csoportra válik szét (Klützke, 2021):

- **Netzbetreiber:** Ezt magyarul „**hálózatüzemeltetőnek**” lehet fordítani. Az ezen csoportba tartozó szervezetek felelősek az adott (energia)**hálózat** (illetve a hozzá kapcsolódó infrastruktúra) **üzemeltetésért**, karbantartásáért és jelentkező hibák / üzemzavarok kiküszöböléséért is. Ezen felül az ő feladatuk még a fogyasztási adatok nyomon követése is. Ezek a szolgáltatók nem választhatóak szabadon, mindenkinek az adott lakóhelye határozza meg, hogy melyikhez tartozik. Ebbe a csoportba tartoznak települési infrastruktúra üzemeltetéséért felelős vállalatok, azaz az úgynevezett **Stadtwerke**-ék is.

- **Energielieferant / Energievertrieber:** Magyarul ezt a csoportot „**energiaszolgáltatóknak**” vagy „**energiaszállítóknak**” lehet nevezni. A felhasználó tőlük veszi az áramot, gázt, stb. Ők határozzák meg a saját áraikat és a felhasználók szabadon választhatnak közöttük.

A *Lovion BIS* szoftvert a 2000-es évek elején kezdték el fejleszteni és a kezdeti fázisban (csak) egy **szimpla térinformatikai rendszerként** funkcionált, ahol a közüzemi szolgáltatók meg tudták jeleníteni és (korlátozottan) kezelni az általuk üzemeltett hálózatokat. Ezt a programot akkoriban még *Location Viewer*-nek hívták, ez volt a *Lovion BIS* elődje. A *Location Viewer* alapja a *General Electric* vállalat által fejlesztett **Smallworld GIS** térinformatikai szoftver volt, amely kifejezetten közüzemi szolgáltatóknak („*utilities*”) készült és a mai napig őket szolgálja ki.



5.2.. ábra – A *Location Viewer* alapnézete (Verhoeven, 2021.)

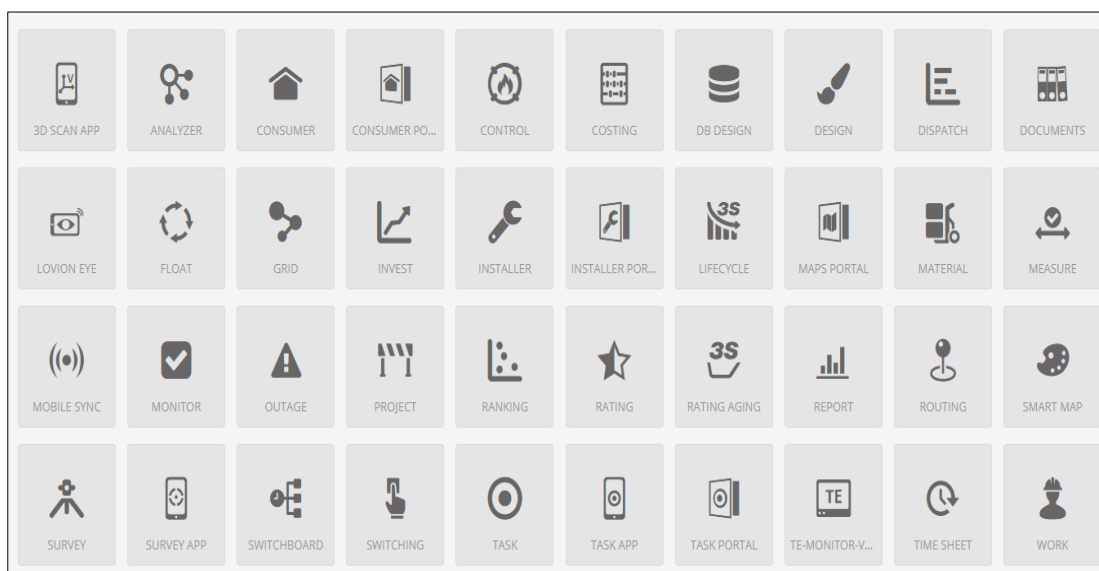
Ebből is látható, hogy a *Lovion BIS* szoftver legfőbb alap-modulja egy **térinformatikai rendszer**. A sokéves fejlesztés során ezen modul köré épült fel a program és **minden további komponens ehhez van igazítva**. A szoftver legújabb használatban lévő verziója a **6.6.5-ös** kiadás, amely a következő külső rendszereket képes integrálni:

- **Térinformatikai rendszerek** – *Geographical Information System (GIS)*: ESRI termékek (*ArcGIS Desktop*, *ArcGIS Pro*), *Smallworld GIS*, *GISMobil*, stb.
- **Web alapú szolgáltatások** → Ide tartoznak többek között az OGC (*Open Geospatial Consortium*) szolgáltatásai, illetve a SOAP (*Simple Object Access Protocol*) szolgáltatásai
- **Vállaltirányítási rendszerek** – *Enterprise Ressource Planning (ERP)*: Wilken, SAP, Schleupen, stb. A kör tovább bővíthető.
- **Grafikai tervező szoftverek** – *Computer-Aided Design (CAD)*: DXF és DWG formátumok expotálására és importlására van lehetőség
- **Dokumentumkezelő rendszerek** – *Document Management System (DMS)*: bármilyen rendszer hozzákapcsolására van lehetőség.
- **SCADA** (*Supervisory Control and Data Acquisition* – felügyeleti szabályozás és adatgyűjtés) **rendszerek**: *HIGH-LEIT*, *PSIcontrol*, *Siemens Leittechnik*, stb.
- **Egyéb szoftverek**: FME (*Feature Manipulation Engine*), Microsoft Office termékek (*Outlook – Exchange*, *Word*, *Excel*)

A szoftver több rész-termékből, azaz **modul**ból áll össze. Ezek a modulok külön-külön is használhatóak, illetve együttesen komplexebb feladatok ellátására képesek. Mivel az egyes szolgáltatóknak eltérő a szervezeti felépítése, profilja és portfóliója, ezért a modulok különböző kombinációi valósulnak meg a gyakorlatban. A *Lovion BIS* jelenleg 40 különböző modullal rendelkezik, amelyek közül a legalapvetőbbek a következők (Verhoeven, 2021):

- *VIEW* → térképes megjelenítő felület
- *EXPLORE*, *REPORT* → adatok megjelenítésére és lekérdezésére alkalmas modulok
- *SMART MAP* → lekérdezések grafikus megjelenítésére szolgáló modul
- *DB DESIGN* → beépített (GUI) adatbázis kezelő rendszer
- *WORK* → feladatok létrehozására alkalmas felület

- *DIPATCH* → feladatok kiosztására alkalmas felület
- *TASK* → feladatok terepi elvégzéséhez szükséges alkalmazás
- *PROJECT*, *INVEST*, *COSTING* → új beruházások tervezéséhez alkalmas modulok
- *MATERIAL* → készletgazdálkodási modul
- *DOCUMENTS* → beépített dokumentumkezelő felület
- *CONSUMER* → felhasználói adatok kezelésére alkalmas modul
- stb.



5.3. ábra – A Lovion BIS moduljainak gyűjtőábrája
(forrás: <https://lovion.de/produkte>)

API-kon (*application programming interface* – alkalmazásprogramozási interfész) keresztül további ügyfélspecifikus, külső modulok is hozzákapszolhatóak a Lovion rendszerhez. Ezek eltérő igények szerint állhatnak össze.

Az egyes modulok alapján különböző komplex vagy egyszerű folyamatokat („Prozess”), illetve részfolyamatokat képes menedzselni a szoftver. Ezek a következők lehetnek:

- Üzemeltetés, karbantartás, hibaelhárítás
- Tervezés, kivitelezés, monitoring

- Vagyongazdálkodás (Asset-Management), erőforrás-gazdálkodás
- Munkafolyamat tervezés, ütemezés és beosztás
- Feladat-kivitelezés és dokumentáció
- Költségvetés-tervezés, kontrollig



5.4. ábra – A Lovion BIS által támogatott folyamatok gyűjtőábrája (forrás: <https://lovion.de/lovion-losungen>)

A **Lovion BIS** szoftverben használt adatok tárolása és kezelése külső relációsadatbázis-kezelő rendszerek hozzákapsolásával (azok szinkronizációjával) **valósítható meg**. Így az ügyfeleknek szabad hozzáférésük van a saját adataikhoz. Jelenleg a következő relációsadatbázis-kezelő rendszereket támogatja a szoftver:

- **Microsoft SQL Server**
- **ORACLE DB**
- **PostgreSQL**
- **SAP-HANA**



5.5. ábra – A Lovion BIS által jelenleg támogatott relációsadatbázis-kezelő rendszerek logói (forrás: <https://lovion.de/strategie/datenbanken>)

A *Lovion BIS* jelenleg **három** különböző **kliensen** (felhasználói felületen) keresztül érhető el:

- *WINDOWS Client* → Ez a normál, asztali felhasználásra alkalmas kliens.
- *APP Client* → Okostelefonokra és táblagépekre fejlesztett kliens.
- *PORTAL Client* → Webböngészőkön keresztül bárholnan elérhető kliens.

A *Lovion BIS 7.1.1*-es verzió bevezetése az *ITS Gruppe* ügyfeleinél ebben az évben kezdődött meg és az átállás folyamatosan megy végbe az egyéni igényeknek megfelelően.

5.3. A relációsadatbázis-kezelő rendszerek *Lovion BIS* szoftverhez történő kapcsolásának metodikája és gyakorlata

A *Lovion BIS* rendszer három alapvető összetevőből épül fel:

- **Szoftver** (adott verziónak megfelelően)
- **Konfigurációs beállítások** (.xml alapú fájlok)
- **Adatbázisok**

Az egész *Lovion BIS* rendszer **XML alapú konfigurációra** épül. Ez azt jelenti, hogy a program egyes összetevői bizonyos .xml kiterjesztésű (vagy azon alapuló) fájlokban található (XML fájlokba írt) konfigurációs beállítások alapján állnak össze.

A **XML** (*Extensible Markup Language*) egy **általános célú leíró nyelv**, amely speciális célú leíró nyelvek létrehozására alkalmas. Fő funkciója, hogy egyéni információs formátumok létrehozását, valamint azok elektronikus megosztását teszi lehetővé. Az úgynevezett **XSL stíluslap** (*eXtensible Stylesheet Language*) az XML stílusnyelve.

A *Lovion BIS* rendszer konfigurációs beállításai alapvetően három különböző alcsoportra oszthatóak fel, amelyek a következők:

- **Program-konfiguráció**
- **Projekt-konfiguráció**
- **Felhasználói-konfiguráció**



5.6. ábra – A *Lovion BIS* felhasználói felületei - kliensei
(forrás: <https://lovion.de/en/strategy-en/our-strategy>)

A Lovion BIS két esszenciális konfigurációs fájlal rendelkezik, amelyek XSL stílusnyelven íródtak, azaz XML típusú kiterjesztések. Ezekhez kapcsolódik a többi konfigurációs fájl, amelyek a szoftver felépítésért felelősek:

1. **.lvn (L_oV_{io}N) fájl** → Az .lvn fájl a Lovion BIS rendszer „fő beavatkozási pontja” (*Haupteingriffspunkt*). Az programot indító Lovion.BIS.exe minden esetben egy ilyen fájl meghívásával indul el, mivel ez tartalmazza az adott projekt felépítéséhez szükséges **alapvető konfigurációs beállításokat**. A központi konfigurációs beállításokon kívül, – a program almappáiban található további fájlokra történő hivatkozással – ez a fájl felelős az összes többi **plugin** és **modul konfigurációjáért** is.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LovionProject xmlns:x="http://www.w3.org/2001/XMLSchema-instance" x:noNamespaceSchemaLocation="lvn.xsd"
3 version="2.0" id="project" externalName="Kunde A">
4
5   <Configuration>
6     <CoordinateSystem unit="mm">
7       <SRID id="25832" translationX="-3200000" translationY="0" />
8     </CoordinateSystem>
9     <Folders>
10      <Folder key="config_bis" path="..\config_bis">
11        <Sync externalName="config_bis" syncJob="BaseConfig" syncSubDirs="true"
12          syncMode="ReceiveAndCleanUp" />
13      </Folder>
14      <Folder key="userSettings" path="BIS\DEFAULT_USER_SETTINGS" />
15    </Folders>
16    <Files>
17      <File key="themes" path="BIS\themesConfiguration.xml" />
18      <File key="views" path="BIS\views.xml" />
19      <File key="featureConfiguration" path="CONFIGURATIONS\FeatureConfiguration.xml" />
20      <File key="featureVisibilityConfiguration"
21        path="CONFIGURATIONS\FeatureVisibilityConfiguration.xml" />
22      <File key="defaultExecutionConfiguration"
23        path="CONFIGURATIONS\DefaultExecutionConfiguration.xml" />
24      <File key="showObjects" path="BIS\ShowObjects.xml" />
25    </Files>
26  </Configuration>
27
28  <PlugIns>
29    <!-- BASIS PLUGINS START -->
30    <!-- Verpflichtend für Lovion VIEW und EXPLORE -->
31    <PlugIn id="View" />
32    <PlugIn id="GeoConnect">
33      <Files>
34        <File key="repositories" path="$data_gis_path$\gis_repositories.xml" lookForSyncTags="true">
35          <Sync externalName="gis_repositories.xml" syncJob="PlugIn" syncMode="Receive" />
36        </Files>
37      </PlugIn>
38    </PlugIns>
39  </LovionProject>
40

```

5.7. ábra – Példa egy .lvn kiterjesztésű fájl információtartalmára vonatkozóan
(Gißrau, Hardt, 2020)

2. **.var fájl** → Az adott .lvn fájlban használt globális azonosítók és változók (*Variable*) meghatározására alkalmas fájl. Itt kerülnek megadásra (egyebek mellett) az **adatbázisok** elérésére (kapcsolódására) vonatkozó információk és a központi „**repository**”-k (tárolók) elhelyezésére vonatkozó információk is.


```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <Entries>
3    <Entry key="customer_id" value="kunde_a" />
4    <Entry key="sql_server" value="localhost\MSSQL2019" />
5    <Entry key="lovion_system" value="test" />
6    <Entry key="lovion_system_text" value="Testsystem" />
7    <Entry key="lovion_version" value="663" />
8    <Entry key="data_gis_path" value="..\data_gis" />
9  </Entries>
10

```

5.8. ábra – Példa egy .var kiterjesztésű fájl információtartalmára vonatkozóan
(Gißrau, Hardt, 2020)

A kapcsolódás és a szinkronizálás gyakorlati folyamata:

A külső („harmadik”) rendszerekből (ide tartoznak az adatbázis-kezelő rendszerek is) származó információk általában helyi replikákban (*Lovion Replica*) kerülnek tárolásra, és ezeket „*Batch-Job*” (kötegetelt munka – automatikus utasítássorozat) segítségével szinkronizálja a *Lovion BIS* rendszer.

Mint azt az előzőekben már többször említettem, a *Lovion BIS* programhoz többféle adatbázis-kezelő rendszer is csatlakoztatható. A .var fájlban írt „key=„sql_server”” tag segítségével az is meghatározható hogy pontosan **melyik adatbázis-kezelő rendszer, melyik instanciájából** történjen meg az adatok behívása és szinkronizálása.

A *Lovion BIS* programhoz írt úgynevezett „**DB tools**” szkriptek segítenek abban, hogy egy adott adatbázis a helyi adatbázis-kezelő rendszerbe felvételre (átmásolásra) kerüljön, azaz **helyi példány** állon belőle rendelkezésre. Továbbá ezeknek az eszközöknek (szkripteknek) a segítségével lehet az adatbázisok helyi példányait törölni, frissíteni, azokról kivonatokat vagy biztonsági mentéseket készíteni. A *Lovion BIS* programhoz írt „*DB tools*” csomag az alábbi szkripteket tartalmazza:

- *Restore* → adatbázis importálása (helyi példány készítése)
- *Backup* → adatbázis helyi példányáról biztonsági mentés készítése
- *Drop* → adatbázis helyi példányának törlése
- *Rename backups* → biztonsági mentések átnevezése

```

LovionDatabases

Lovion Databases V.0.1

The following call parameters are used within this script call

■ Mode           : interactive
■ DbSystem        : mssql

Available database functions:

[1] Restore databases
[2] Backup databases
[3] Drop databases
[4] Rename backups (drag&drop only)
[5] Stop script
■ Please, choose:

```

5.9. ábra – Lovion BIS „BD tools” szkriptek (forrás: Lovion BIS termékdokumentáció)

6. A GEOMETRIAI ADATOKAT TÁMOGATÓ ADATBÁZIS-KEZELŐ RENDSZEREK FELHASZNÁLÁSÁNAK VIZSGÁLATA AZ ÜZEMI INFORMÁCIÓS RENDSZEREKBE

Ebben a részben azt szeretném megvizsgálni, hogy a 4. pontban bemutatott relációsadatbázis-kezelő rendszerek közül melyik a legalkalmasabb egy üzemi információs rendszerben történő felhasználásra. Mivel korábban már részletesen bemutatásra kerültek ezek az adatbázis-kezelő rendszerek, ezért itt inkább egy összehasonlító elemzést, illetve alkalmassági vizsgálatot szeretnék elvégezni.

6.1. Az egyes geometriai adatokat támogató relációsadatbázis-kezelő rendszerek összehasonlítása

A 4. pontban bemutatott relációsadatbázis-kezelő rendszerek összehasonlítását nem csupán objektív paraméterek alapján szeretném megtenni. Azt is számításba veszem, hogy melyik adatbázis-kezelő rendszerrel tudok/tudtam a legdinamikusabban dolgozni, például mennyire könnyen találhatóak meg benne egyes függvények, mennyire egyértelműek azok leírásai, illetve, hogy például melyik rendelkezik a leginkább felhasználóbarát kliens alkalmazással, grafikus interfésszel.

A konkrét összehasonlítást a következő táblázat tartalmazza:

ÖSSZEHASONLÍTÁSI SZEMPONTOK	MS SQL Server	PostGIS	Spatialite	Oracle Spatial and Graph
ACID kompatibilis	igen	igen	igen	igen
OGC 06-103r4 szabványhoz illeszkedik	igen	igen	igen	részben
OGC 06-104r4 szabványhoz illeszkedik	részben	igen	igen	részben
nyílt forráskódú	nem	igen	igen	nem
Lovion BIS-el jelenleg kompatibilis	igen	igen	nem	igen
XML támogatás	igen	igen	nem	igen
kliens alkalmazás / grafikus interfész (GUI) kezelhetősége	optimális	teljeskörű	egyszerű	megfelelő
GitHub: stars / forks	8.513 / 8.328	12.213 / 3.844 (PostgreSQL)	3.871 / 668 (SQLite)	2.147 / 1.074 (Oracle)
StackShare: company stacks / developers stacks	1.229 / 15.930	6.213 / 77.890 (PostgreSQL)	774 / 15.024 (SQLite)	265 / 1.531 (Oracle)
DB-engines összesített (overall) helyezés	3.	28. / PostgreSQL: 4.	138. / SQLite: 9.	1. (Oracle)
DB-engines Spatial DMBS helyezés	nem releváns	1.	3.	nem releváns
ÖSSZESÍTETT SAJÁT HELYEZÉS:	3.	1.	2.	3.

6.1. táblázat – A vizsgált térbeli adatok tárolására és kezelésére alkalmas relációsadatbázis-kezelő rendszerek összehasonlítása

Az értékelési szempontokhoz kapcsolódó további magyarázatok:

- **kliens alkalmazás / grafikus interfész (GUI) kezelhetősége** → Mivel mindegyik adatbázis-kezelő valamelyik kliensével dolgoztam már, ezért itt az egyéni tapasztalataimra hagyatkoztam és különböző szubjektív értékeléseket adtam meg.
- **GitHub stars / forks** → Az adatbázis-kezelő népszerűségére vonatkozó adatok, azaz, hogy mennyire aktívan használják világszerte. Ezek a számok azt jelölik, hogy az adott adatbázis-kezelőhöz hány ágon és mennyi feltöltés kapcsolódik. Itt és a következő pontban azért az adatbázis-kezelők törzsprogramjait hasonlítottam össze, mivel ezek adtak releváns képet.
- **StackShare – company stakcs / developers stakcs** → Szintén a népszerűség és a felhasználtság jelzésére alkalmasak ezek az információk. A stackshare.com

közösségi oldal adatai alapján az látható, hogy hány cég és hány fejlesztő használja az adott adatbázis-kezelő rendszert.

- **DB-engines összesített (overall) helyezés** → A <https://db-engines.com/> az egyik legnépszerűbb adatbázis-kezelőket összehasonlító oldal. Ezen oldal összesített rangsorának adatait mutatják ezek a számok.
- **DB-engines Spatial BMBS helyezés** → A <https://db-engines.com/> oldal által kiadott, kifejezetten térbeli adatok tárolására és kezelésére specializálódott adatbázis-kezelő rendszerek rangsora.

6.2. Alkalmassági vizsgálat egy üzemi információs rendszerbe történő integrálásra vonatkozóan

Ebben a részben szeretném kiválasztani azt az adatbázis-kezelő rendszert, amellyel a 7. pontban minta-adatbázist fogok készíteni, majd azt a *Lovion BIS* rendszerbe integrálni. Tehát itt állapítom meg, hogy a 6.1. pontban írt összehasonlítás alapján **melyik rendszer alkalmazható legjobban egy üzemi információ rendszerben történő felhasználásra.**

A 6.1-es részben leírtak alapján szeretnék egy sorrendet felállítani, az adatbázis-kezelő rendszerek alkalmasságának kipróbálására vonatkozóan:

- Első körben a **PostGIS** rendszert szeretném gyakorlatban kipróbálni (tesztelni), mivel ingyenesen hozzáférhető minden funkciója és a *DB-engines* által felállított lista alapján ez a legjobb térbeli adatok kezelésére alkalmas adatbázis (az összesített listán a 28.). A *PostgreSQL* pedig az összesített listán a negyedik helyen szerepel, továbbá ezzel korábban dolgoztam; és a grafikus interfésze (pgAdmin 4) nagyon jól kezelhető és sok funkcióval rendelkezik. A *GitHub* és a *StackShare* adatai alapján ez a legnépszerűbb térbeli adatbázis-kezelő rendszer.
- Ezt követően a **SpatiaLite** következne, mivel az is nyílt forráskódú és az egyszerűsége miatt az egyik legkönnyebben kezelhető ilyen jellegű adatbázis-kezelő rendszer. Kiterjedtek a térbeli funkciói és ezzel is dolgoztam már korábban. Ezzel szemben viszont sajnos csak részben támogatja az *XML* kiterjesztéseket és egyelőre még nincs teljesen kidolgozva a *Lovion BIS*-hez való hozzákapcsolásának a metodikája (nincs rá precedens).

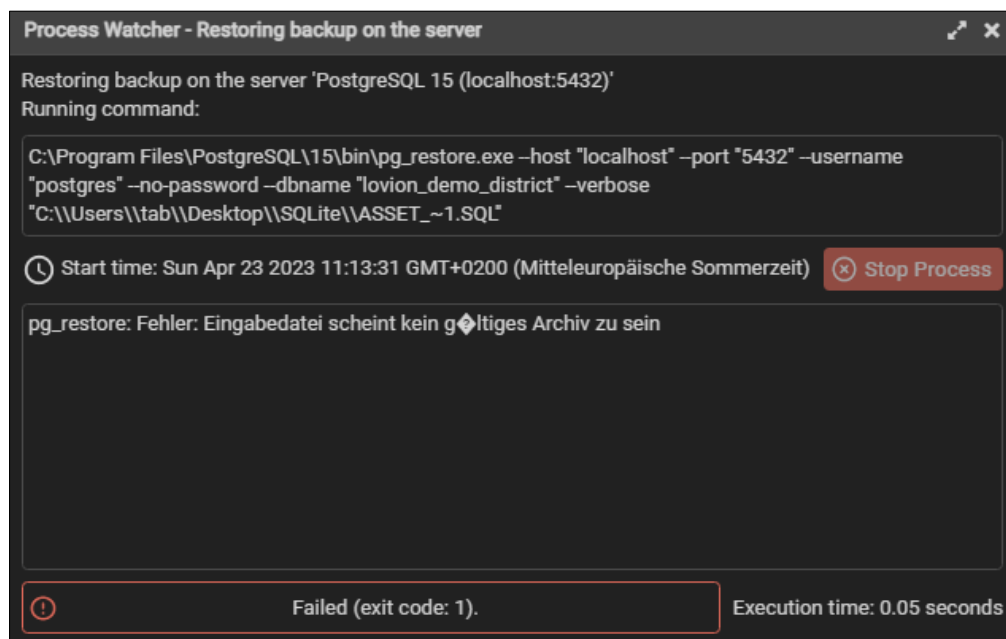
- Amennyiben nem jutnék megfelelő eredményhez a nyílt forráskódú – a *PostGIS* vagy a *Spatialite* – rendszerekkel, úgy ezeket követően először az **MS SQL Server**-rel próbálkoznék, mivel az már gyakorlatban is jól működik a Lovion BIS rendszerrel és széles körben van alkalmazva. Utolsóként az **Oracle Spatial and Graph** kerülne sorra. Bár több szempontból egy nagyon fejlett ilyen jellegű adatbázis-kezelő rendszer, sajnos nem teljes körben (nem minden funkciója) ingyenes, valamint szintén nincs még széles körben alkalmazva a jelenlegi Lovion BIS felhasználók körében.

6.2.1. A PostgreSQL és a PostGIS kompatibilitásának vizsgálata

A *PostGIS* üzemi információs rendszerben (*Lovion BIS*-ben) történő alkalmazásának teljeskörű vizsgálata céljából, első körben szerettem volna egy már használatban lévő adatbázist exportálni a *Lovion BIS*-ből, hogy azt tudjam tesztelni *PostgreSQL-PostGIS* környezetben.

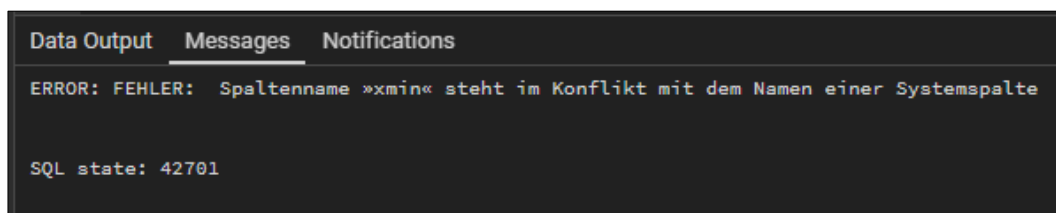
A *Lovion BIS*-hez rendelkezésre áll az ún. „**Lovion BIS 6.6.5 – Demomodell**” teszt-környezet. Ebből a teszt-modellből választottam ki az „*asset_district_demomodel*” nevű adatbázist, amely a mintaterület – Krefeld, Nordrhein-Westfalen tartomány, Németország – közüzemi ellátási körzeteit tartalmazza (poligon típusú geometriai adatokkal). Ennek exportálását az 5.3. részben bemutatott „*DB-tools*” segítségével (*Backup* szkript) tettem meg, majd a következőkben leírtak szerint jártam el:

- A *Lovion BIS 6.6.5 – Demomodell*-hez jelenleg egy *MS SQL Server* adatbázis van hozzákapsolva.
- Az *MS SQL Server*-ben tárolt adatok alap fájlformátuma a *Microsoft SQL Server Query File*, azaz a *.sql* kiterjesztésű. Az adatbázisok eredeti példányairól készíthető biztonsági másolat pedig (*Backup*) *.bak* kiterjesztésű.
- A *PostgreSQL .backup*, *.sql* és *.patch* kiterjesztésű fájlokat képes közvetlenül importálni (*Restore*).
- Ennek megfelelően megpróbáltam az „*asset_district_demomodel.sql*” fájlt közvetlenül behívni (*Restore*) a *pgAdmin4* felületen keresztül, ám ott a következő ábrán látható hibaüzenetet kaptam.



6.1. ábra – A PostgreSQL (pgAdmin4 felületen keresztül) hibaüzenete az „asset_district_demomodel.sql” adatbázis behívásakor

- Mivel így egy tábla sem került importálásra az adatbázisból és nem tudtam tovább dolgozni vele, ezért a következő adatmigrációs eljárással próbálkoztam meg:
 - Az *ODBC Data Source (Open Database Connectivity)* segítségével létrehoztam egy kapcsolatot az *MS SQL Server „MSSQL2019”* instanciájához kapcsolódóan.
 - Az *Ispirer SQLWays Wizard 10* program segítségével végrehajtottam a komplett adatbázis migrációját *MS SQL Server*-ből *PostgreSQL*-be.
- Néhány táblázat azonban így sem került átvételre, így tovább kutattam a hiba eredete után. Kiderült, hogy az volt a probléma, hogy az adatbázis több táblázatában is szerepelnek a „xmin, ymin, xmax, ymax” elnevezésű oszlopok, amelyek így (ezen a néven) nem importálhatóak az adatbázis-kezelő rendszerbe. Mivel a *PostgreSQL-PostGIS* rendszer alapstruktúrája maga is tartalmaz ugyan ilyen elnevezésű elemeket, ezért ebben a formában nem lehet behívni azokat a rendszerbe. Inkonzisztenciához vezet ez az eljárás.
- A hibára úgy sikerült rájönnöm, hogy manuálisan, azaz SQL-parancs segítségével próbáltam meg létrehozni az egyes táblázatokat. A pontos hibaüzenet a következő ábrán látható:



6.2. ábra – A PostgreSQL (pgAdmin4 felületen keresztül) hibaüzenete az „xmin” nevű oszlopot tartalmazó adattábla beolvasásakor

- Megpróbálkoztam az oszlopok átnevezéseivel, de úgy sajnos az adatbázismodell struktúrája összeomlott és a *Lovion BIS 6.6.5 – Demomodell* nem volt képes a továbbiakban megfelelően beolvasni az adatbázist.
- Mivel ez az eljárás egy olyan hiányosságot tárt fel, ami alapján az eredetileg rendelkezésre álló minta-adatbázis – azaz a *Lovion BIS 6.6.5 – Demomodell* adatbázisa – nem, vagy csak egy teljes át-/újraépítést követően használható fel PostgreSQL rendszerben, ezért úgy döntöttem, hogy továbblépek és a második potenciális adatbázis-kezelő rendszerrel, azaz a *Spatialite*-al próbálkozok tovább.

Kiegészítésként megjegyzendő, hogy a PostgreSQL adatbázis-kezelő rendszerek a következő ábrán látható, *.var* fájlba írandó kiegészítéssel kapcsolhatóak a *Lovion BIS* rendszerhez. Ezt a gyakorlatban is kipróbáltam, illetve kerestem olyan (Lovion BIS) ügyfeleket, akik PostgreSQL-ben tárolják az adataikat.

```

13
14      <!-- PostgreSQL Server -->
15      <Entry key="Server" value="localhost" />
16      <Entry key="Username" value="postgres" />
17      <Entry key="Password" value="root" />
18      <Entry key="Port" value="5432" />
19      <Entry key="AdditionalConnectionParameters" value="" />
20

```

6.3. ábra – A PostgreSQL Lovion BIS rendszerhez történő kapcsolása (forrás: Lovion BIS fejlesztői dokumentáció)

A 6.1. és a 6.2. részekben írtak ellenére sajnos úgy kellett döntenem, hogy másik adatbázis-kezelő rendszer(eke)t is meg kell vizsgálnom a jobb eredmény elérésének érdekében. A szakdolgozat egyik rész céljának tekinthető, hogy itt – a 6.2.1. részben – kiderültek a *PostgreSQL-PostGIS* adatbázis-kezelő rendszer *Lovion BIS*-ben történő felhasználásának korlátai, azaz, hogy már a teszt-környezet (*Lovion BIS 6.6.5 – Demomodell*) esetében is léphetnek fel nehézségek az integráció során.

2024 februárjától szeretném folytatni a tanulmányaimat a salzburgi *UniGIS*-en (*Paris Lodron Universität Salzburg*) és elvégezni a vonatkozó geoinformatikai mesterképzést (<https://unigis.at/unigis-master-of-science/>). Az ott írandó diplomamunkámban mindenképpen ezt a témát fogom továbbvinni / kibővíteni / korrigálni; így a *PostgreSQL-PostGIS* környezet felhasználását is tovább tesztelem majd és megpróbálnám megoldani az imént leírt problémát. A **jelenlegi szakdolgozat** potenciális eredménye érdekében – mint fentebb már írtam – itt viszont **másik rendszer használata** indokolt.

6.2.2. Az SQLite és a SpatiaLite kompatibilitásának vizsgálata

A 6.2.1. rész végén leírtaknak megfelelően szükséges volt egy másik adatbázis-kezelő rendszer alkalmassági vizsgálata is, mivel ott korlátokba ütköztem. Mivel a vizsgált adatbázis-kezelők közül az *SQLite* (illetve a *SpatiaLite*) a másik **ingyenes** (szabad licenszű) rendszer, ezért ezzel folytattam a további vizsgálatokat. Továbbá ez a rendszer a **legegyszerűbb felépítésű** és a **legkönnyebben használható** grafikus felülettel is ez rendelkezik. Úgy gondoltam, hogy a 6.2.1. részben írt **hibák** egyik lehetséges **elkerülési módja**, ha egy **minél egyszerűbb rendszerrel** dolgozok tovább.

Az alábbi lépések mentén haladtam tovább:

- A *SpatiaLite* (illetve az *SQLite*) rendszerek alapvető fájlformátumai a *.sqlite*, *.atlas*, *.db* és *.gpkg* (ezen legutóbbi a *QGIS* alapformátuma). Ezen kívül képesek még más formátumok (pl.: *.csv*, *.txt*, *.geojson*, *.shp*, *.stb*.) importálására is, viszont az *MS SQL Server* által támogatott *.sql*, illetve *.bak* fájlok közvetlenül nem hívhatóak be a rendszerbe.
- A két adatbázis kezelő közötti adatmigrációhoz itt két programot tudtam segítségül hívni:
 - SqlConverter_v1_20: <https://www.codeproject.com/Articles/26932/Convert-SQL-Server-DB-to-SQLite-DB>
 - SQL Converter 1.0.0.0: https://drive.google.com/file/d/1asG_hAqxrFrBPyMmWifmKz8tt7PZlF_1/view

- Itt már sikerült az adatbázist adatvesztés nélkül áthelyezni, tehát az összes tábla, összes oszlopa és rekordja átvitelre került az „asset_district_demomodel” adatbázisnak. A PostgreSQL-nél jelentkezett hiba itt nem volt akadály, azaz az eredeti oszlopnevek megtartása mellett tudtam tovább dolgozni a táblázatokkal.
- A *Spatialite* mellett inntentől a *QGIS* térinformatikai szoftvert is segítségül hívtam (használtam), mivel ebben van egy beépített (*Spatialite* alapú) adatbázis kezelő modul (*DB kezelő*), amivel egy időben grafikusán is megjeleníthetők az adatok.

geom_id	geom_name	rws_id	rws_name	rws_type	geomfield_name	world_id	xmin	xmax	ymin	ymax	sta
10173486-7156-418-842b-1f80d6c20ee		020676c-e9b7-470-b161-dec20a4d18d		geom_id	geomfield_name	world_id	5.35747718	5.32001728	6.8508152	6.8020080403988	5.37164790
102026ec7ac79-4cde-b6d6-8fcaaf7ab166		43ac089b-e955-472b-999a-0120d6b46c		geom_id	geomfield_name	world_id	5.36313028	5.3900473928618	6.8730008087208	6.919940918	5.36485325
1039999b3-fcde-4d14-b0d5-e5a0a0a0a0a0		35-56-38d780a-2096-476d-6477-5a0f6d7f2c8		geom_id	geomfield_name	world_id	5.40257218	5.40257218	6.86198808	6.900229678	5.40404675
104-703e-46d5-4916-8f5c16a5d2		05a5d6d-e9a1-d4d5-8dab-57a0d0159		geom_id	geomfield_name	world_id	5.34120988	5.34120988	6.837795208	6.862847718	5.34120988
1050533a-b4b5-4477-803c-59d6a6a2b1d		058a010-1164-4855-5a65-b170e70365		geom_id	geomfield_name	world_id	5.39334887348118	5.410063818	6.88122230849518	6.930300628	5.41006381
10620497b-ab99-4609-8f89-0a65626e4d		c87a6d3-0706-4955-8a33-04a05a9a0c		geom_id	geomfield_name	world_id	5.392470868	5.428170368	6.91338558	6.931688968	5.42443981
1071a689-c9c3-4053-8730-1942a297b0b		4483d15-299-4c47-ad6d-0430b8f0b6d		geom_id	geomfield_name	world_id	5.35581418	5.3724629505818	6.870246388	6.91544518	5.37246295
110303ab-1e6d-47c1-09d4-00001c4a4d0		9693933-1409-4043-1c1e-d4d2116276b		geom_id	geomfield_name	world_id	5.35581418	5.3724629505818	6.870246388	6.91544518	5.37246295
1173226e-3131-4398-8408-dab1e4e006b		51a7a77-1e63-4bc3-8eab-19d8f164c3		geom_id	geomfield_name	world_id	5.36233043302078	5.39255951069328	6.87643334543438	6.9032041500818	5.39255951
118d2a46-7874-43a3-953a-d90b67045d8		51a1623-b6d6-4886-1e1a-cd0b7b8d1a		geom_id	geomfield_name	world_id	5.360339338	5.433849968	6.836036328	6.87724158	5.38873881
1202a6f13-d4d4-48ac-8a05-94d43171c3		b47ec791-c832-463a-8b7b-261e15d0e9		geom_id	geomfield_name	world_id	5.36005968	5.3948988	6.919848328	6.968838918	5.38873881
12071965-3d46-4047-b10e-17002c0e3d7		b47ec791-c832-463a-8b7b-261e15d0e9		geom_id	geomfield_name	world_id	5.41842091523988	5.4272208	6.91196538	6.938484323988	5.41842091
121a2b05a-e958-402c-84d6-2e0d4e1b6f		2707a7b-4572-4b0e-9840-5e7c533f0a		geom_id	geomfield_name	world_id	5.40257218	5.40257218	6.86198808	6.900229678	5.40404675
123a4080-5a8f-451d-863b-351ac169d4d9		1a100058-97c1-4095-b429-7d489c38080		geom_id	geomfield_name	world_id	5.44280148	5.49237218	6.86198808	6.900229678	5.40404675
12ac2400-8096-4a09-a216-5e119e458d		9e6c7a01-9b2a-4953-a185-fa050cd7a9f		geom_id	geomfield_name	world_id	5.41133828	5.45446658	6.86142438	6.891720428	5.41473507
12d000a1-73a6-4090-ba16-16a2b17f8d3		35-56-38d780a-2096-476d-6477-5a0f6d7f2c8		geom_id	geomfield_name	world_id	5.42101058	5.4777678	6.90426388	6.93941198	5.42101058
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.35581418	5.3724629505818	6.870246388	6.91544518	5.37246295
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.3885200143488	5.433849968	6.836036328	6.87724158	5.38873881
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.42101058	5.4777678	6.90426388	6.93941198	5.42101058
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.4030813648	5.4030813648	6.85215180203488	6.85747978037818	5.40308136
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.412770357702328	5.464491386666666	6.85215180203488	6.85747978037818	5.40308136
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.3302584445717928	5.408801100030878	6.831647015714478	6.86664108254068	5.33025844
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.405157033448748	5.406339761003488	6.85718020323428	6.86000080596988	5.40515703
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.403321348	5.464491386666666	6.85215180203488	6.85747978037818	5.40332134
12d1a5ab-739d-47a0-a3b1-a6d7171132b		1a0330a9-9b67-462b-807c-0b0b2a2559		geom_id	geomfield_name	world_id	5.37946728	5.4170202328	6.8572988	6.88964488	5.41006381

6.4. ábra – A Lovion BIS 6.6.5 – Demomodel, asset_district_demomodel adatbázisa
SpatiaLite rendszerben megjelenítve

7. EGY SPATIALITE-AL LÉTREHOZOTT ADATBÁZIS LOVION BIS RENDSZERBE TÖRTÉNŐ INTEGRÁLÁSÁNAK TESZTELÉSE

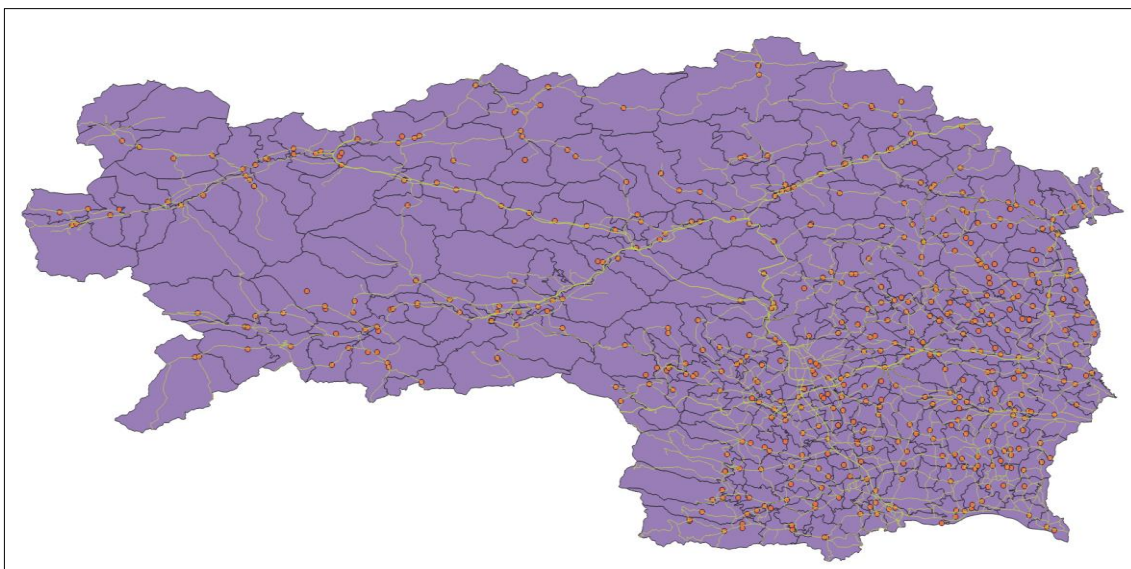
7.1. Az adatbázis létrehozása

A saját adatbázis elkészítése kapcsán egy olyan megoldásra gondoltam, amely valamennyire kapcsolódik a közüzemi/ellátási szolgáltatásokhoz. Ezért úgy döntöttem, hogy készítek egy adatbázist, amelyen látszik a **Stájerország** (Ausztria) beli **hulladéklerakó/-kezelő telepek elhelyezkedése**, az önkormányzati közigazgatási határok és a főbb közlekedési útvonalak mellett. Így terveim szerint az adatbázis – több táblázaton keresztül – tartalmazna **pont**, **vonal** és **poligon** típusú **geometriákat** is.

A saját adattábla összeállítása a következő lépések szerint történt meg:

- 1) Első körben összegyűjtöttem a releváns adatokat a <https://www.data.gv.at/suche/> oldalról, amely szabadon felhasználható adatkészleteket tartalmaz. Ehhez összesen három különböző *.shp* kiterjesztésű (shape) fájlt töltöttem le:
 - **PONT GEOMETRIA** → A stájerországi hulladékgyűjtő-/hulladéklerakó-telepek elhelyezkedését tartalmazó adattábla: „Abfallwirtschaftliche Anlagen Steiermark” (*Abfallwirtschaftliche_Anlagen.shp*). Az import során „*abfallw_anlagen_stmk*”-ra neveztem át ezt a táblát.
 - **VONALAS GEOMETRIA** → A stájerországi fő közlekedési útvonalakat tartalmazó adattábla: „Übergeordnetes Straßennetz Steiermark” (*Verkehrsnetz_hochrangig.shp*). → Az import során „*verkehrsnetz_h_stmk*”-ra neveztem át ezt a táblát.
 - **POLIGON GEOMETRIA** → A stájerországi önkormányzati közigazgatási határokat és a helyi hulladékkezelő szervezetek beosztását tartalmazó adattábla: „Abfallwirtschaftsverbände Steiermark” (*Abfallwirtschaftsverbände.shp*). → Az import során „*gemeinden_stmk*”-ra neveztem át ezt a táblát.
- 2) Létrehoztam egy új *Spatialite* adatbázist „*waste_management_styria_wgs84.sqlite*” néven.
- 3) Importáltam az új adatbázisba az 1) pontban felsorolt három adattáblát. Az importálás során szükséges volt megadni a karakterkódolásokat, ahol **UTF-8**-at vagy **CP1250**-et állítottam be; továbbá az alkalmazandó vetületi rendszereket, ahol pedig minden esetben a **WGS84**-et (*World Geodetic System 1984*, SRID/EPSC: 4326).
- 4) Az importálásokat követően elvégeztem az adattáblák adattartalmának ellenőrzését és összevetését.
- 5) A következő lépésben csatlakoztattam a *QGIS*-hez (verzió: 3.12 Hannover) a „*waste_management_styria_wgs84.sqlite*” nevű *Spatialite* adatbázist.
- 6) Behívtam a három adattáblát, mint három különböző réteget és mentettem a *QGIS* projektet a „*waste_management_styria_wgs84.qgz*” nevű fájlba.

- 7) A megjelenítési beállítások módosításait követően a három adattábla ábrázolása az alábbiak szerint történt meg:



7.1. ábra – A stájermarki hulladéklerakó-telepek elhelyezkedése – az önkormányzati közigazgatási határok és a főútvonalak mellett – QGIS 3.16-ban megjelenítve

- 8) Az adatbázis-módosításokat a továbbiakban a QGIS beépített – *SpatiaLite* alapú – adatbázis kezelőjével (*DB kezelő*) végeztem.
- 9) Létrehoztam a táblázatokhoz tartozó **térbeli indexeket** az alábbi SQL parancsok segítségével:

```
select createspatialindex ('abfallw_anlagen_stmk', 'geometry')
select createspatialindex ('gemeinden_stmk', 'geometry')
select createspatialindex ('verkehrsnetz_h_stmk', 'geometry')
```

Így már rendelkezésemre állt egy olyan összetett adatbázis, amely tartalmazza a főbb geometriai típusokat és komplex térbeli számítások is végezhetőek rajta. Úgy gondoltam, hogy a térbeli adatok kezelésének jobb megértéséhez (megvizsgálásához) célszerű lenne tovább bővíteni ezt az adatbázist és létrehozni egy (vagy több) új adattáblázatot egyedi, számított tartalommal. Így itt a következő célfeladatot határoztam meg: **Azon hulladéklerakó telepek listázása (összeírása), amelyek maximum 5 km-es távolságra találhatók a B320-as (Ennstal Straße nevű) főúttól.**

A célfeladatot a következő lépések szerint valósítottam meg:

- 1) Első körben mentettem a projektet egy új fájlba „waste_management_styria_wgs84_01.qgz” néven, hogy később vissza tudjak majd térni az eredeti állapothoz.
- 2) Klónoztam a „verkehrsnetz_h_stmk” táblázatot „B320”-as néven, az alábbi SQL parancs segítségével:

```
SELECT CloneTable('MAIN', 'verkehrsnetz_h_stmk', 'b320', 1)
```

- 3) Az új, klónozott táblázatba feltöltöttem a B320-as (*Ennstal Straße*) főút adatait az következő SQL parancs segítségével:

```
insert into b320 (pk_uid, strcat, strbez, nametext, geometry)
values (
  (select pk_uid from verkehrsnetz_h_stmk where pk_uid='834'),
  (select strcat from verkehrsnetz_h_stmk where pk_uid='834'),
  (select strbez from verkehrsnetz_h_stmk where pk_uid='834'),
  (select nametext from verkehrsnetz_h_stmk where pk_uid='834'),
  (select geometry from verkehrsnetz_h_stmk where pk_uid='834')
)
```

- 4) A puffer zóna meghatározása előtt meg kellett változtatnom a vetületi rendszert. Mivel a WGS84 geodéziai dátum ellipszoidi koordináta rendszeren alapul, ezért csak fokban lehetséges mérni rajta. Annak érdekében, hogy metrikus rendszerben tudjam meghatározni az övezetet, az ETRS89 – UTM zone 32N (SRID/EPSC: 25832) vetület használatára volt szükség. A transzformációt QGIS segítségével hajtottam végre:

- A „Vektor / Adatkezelő eszközök / Réteg” más vetületbe eszközt nyitottam meg, majd a 25832 EPSG számú vetületet választottam ki és futtattam a parancsot.
- Ezt követően az ideiglenes réteget egy .shp kiterjesztésű fájlba mentettem (b320_etr89.shp néven), UTF-8-as karakterkódolással.
- Így már egyszerűen hozzá tudtam adni azt a „waste_management_styria_wgs84.sqlite” nevű adatbázishoz.
- Az új adattábla esetében is kellett létre kellett hoznom térbeli indexet, amit a következő SQL parancs végrehajtásával tettem meg:

```
select createspatialindex ('b320_etr89', 'geometry')
```



7.2. ábra – A stájermarki B320-as főút ETRS89 (UTM zone 32N) vetület alapján történő leképzése – térkép-kivágat, adattábla adatok, adattábla tartalom

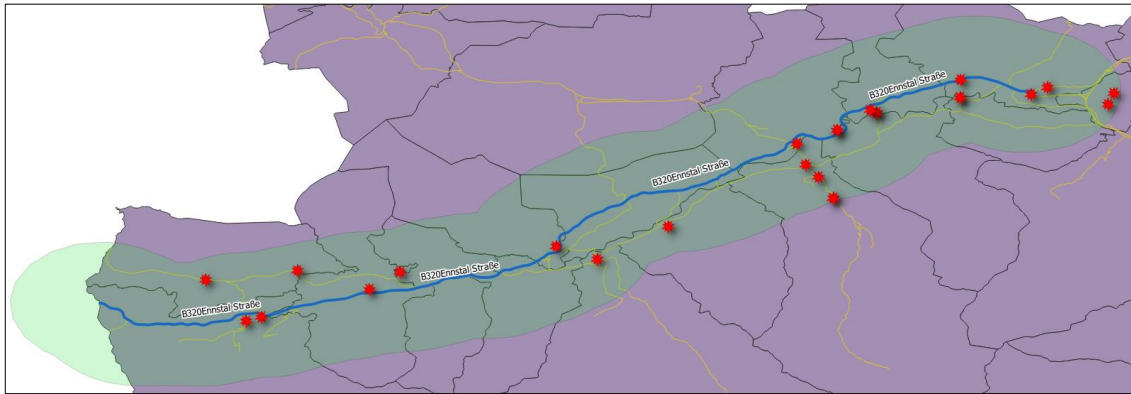
- 5) (A jobb követhetőség és az egyértelműbb megjeleníthetőség érdekében feliratoztam a B320-as utat a megfelelő szelvényen.)
- 6) Itt mentettem a projektet „waste_management_styria_wgs84_02.qgz” néven.
- 7) A következő fontos lépés az övezetgenerálás (puffer-terület képzés) volt. Ehhez a QGIS beépített eszközét (*Vektor / Geoprocessing eszközök / Övezet*) használtam, de a *Spatialite*-ban az következő „ST_Buffer” művelet segítségével is elvégezhető a feladat:

```
ST_Buffer( geom Geometry , dist Double precision
[ , quadrantsegments Integer ] ) : Geometry
```

- 8) Az új réteget mentettem „b320_etr89_buffer” néven egy .shp kiterjesztésű fájlba és hozzáadtam a „waste_management_styria_wgs84.sqlite” adatbázishoz egy új táblaként (EPSG/SRID: 25832, karakterkódolás: UTF-8).
- 9) A B320-as úthoz képzett – 5 km-es – pufferzónában megtalálható hulladék-kezelő telepek meghatározásához is szintén a QGIS beépített modulját (*Vektor / Kutató eszközök / Kiválasztás pozíció alapján*) használtam. A tartalmazás azonban megállapítható a következő SQL parancs segítségével is:

```
ST_Within( geom1 Geometry , geom2 Geometry ) : Integer
```

- 10) Az így kiválasztott adatok alapján – ami összesen 26 darab hulladéklerakó/-kezelő létesítményt jelent – készítettem egy (exportált) shape fájlt „b320_wastem_facilities” néven. Ezt behívtam a „waste_management_styria_wgs84.sqlite” adatbázisba, ami így még egy további adattáblával bővült.



7.3. ábra – A B320-as főút 5km-es pufferzónájában található hulladék-kezelő telepek QGIS-ben megjelenítve

pk_uid	anr	afyp	akuerzel	anlagekurz	betrieb_se	betrieb_bi	a_plz	a_gem	a_str	betreiber	geometry
1	36	6122307	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Liezen		8940	Liezen	Geskusstraße 50	Abfallwirtschaftsverband ...	POINT
2	51	6120203	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Aich		8967	Aich	Langtrum 153	Abfallwirtschaftsverband ...	POINT
3	183	6120303	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Aigen im Ennstal		8943	Aigen/Ennstal	Ketten 71	Gemeinde Aigen im Ennstal	POINT
4	187	6121303	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Gröbming		8962	Gröbming	Mitterberger Straße 852	Marktgemeinde Gröbming	POINT
5	203	6120801	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Irdning-...		8953	Irdning-Donnersbachtal	Winklern 34	Marktgemeinde Irdning-...	POINT
6	225	6124203	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Schladming		8970	Schladming	Salzburgerstraße 673	Manfred Arzbacher GmbH	POINT
7	231	6124301	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Selzthal	1901	8900	Selzthal	Gosserstraße 4	Gemeinde Selzthal	POINT
8	243	6121801	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Irdning-...		8952	Irdning-Donnersbachtal	Falkenburg Donnersbachtal 190	Marktgemeinde Irdning-...	POINT
9	254	6124404	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Stainach		8950	Stainach-Pürgg	Ennsstraße 163	Marktgemeinde Stainach-...	POINT
10	256	6124901	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Weißenbach bei ...		8940	Weißenbach b. Liezen	Hauptstraße 77	Gemeinde Weißenbach bei ...	POINT
11	281	6123601	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Ramsau am ...		8972	Ramsau/Dachstein	Ramsau 161	Gemeinde Ramsau am ...	POINT
12	298	6122202	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Lassing		8940	Lassing	Döllach 95	Gemeinde Lassing	POINT
13	299	6122801	ASZ	Altstoffsammelstelle inkl. PSS Problemstoffsammelstelle	Altstoffsammelzentrum incl. PSS Öblarn		8960	Öblarn	Öblarn 44		POINT
14	384	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Schladming	1976	8970	Schladming	Augasse 169	Martina Sieder	POINT
15	390	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Solk		8961	Solk	Stein an der Enns 46	Christian Tasch	POINT
16	401	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Aigen		8943	Aigen/Ennstal	Ketten 71	Gemeinde Aigen im Ennstal	POINT
17	411	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Wörschach -...		8942	Wörschach	Maltschem 28		POINT
18	449	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Döllach ARA ...		8940	Lassing	Döllach 95	Gemeinde Lassing	POINT
19	450	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Stainach		8950	Stainach-Pürgg	Ennsstraße 163	Wassergenossenschaft ...	POINT
20	456	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Haus	1997	8967	Haus	Markt Haus 208	Marktgemeinde Haus	POINT
21	484	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Trautenfels		8951	Stainach-Pürgg	Trautenfels 72	Marktgemeinde Stainach-...	POINT
22	496	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Ramsau		8972	Ramsau/Dachstein	Rössing 384		POINT
23	497	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Donnersbach		8953	Irdning-Donnersbachtal	Winklern 34		POINT
24	500	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Selzthal	1981	8900	Selzthal	Badgasse 9	Gemeinde Selzthal	POINT
25	529	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Irdning		8952	Irdning-Donnersbachtal	Falkenburg Gewerbestraße 200		POINT
26	536	0	TKV	Tierkörperpersammelstelle	Tierkörperverwertungssammelstelle Liezen		8940	Liezen	Richard-Steinhuber-Straße 26	Stadtgemeinde Liezen	POINT

7.4. ábra – A B320-as főúttól maximum 5km-re található hulladék-kezelő telepek listája SpatiaLite adatbázisban megjelenítve

Úgy gondolom, hogy így sikerült végrehajtanom egy olyan műveletsorozatot, amely érintette a három alapvető geometriai forma (pont, vonal, poligon) mindegyikét és egy igen összetett (összesen hét egymással összefüggő adattáblából álló) adatbázis létrejöttét eredményezte.

Ezt az adatbázist szeretném integrálni a *Lovion BIS* rendszerbe és megnézni, hogy működik-e ott. Azaz, hogy az összes adattábla megjeleníthető-e teljes mértékben és kezelhető-e a programban?

Itt szeretném még megemlíteni a *SpatiaLite* rendszer néhány előnyét (és hátrányát), amelyre az adatbázis felépítése, alakítása során jöttem rá:

- Több, eltérő vetületi rendszerrel rendelkező adattábla, egy adatbázisban történő tárolására és kezelésére is alkalmas, ami egy nagyon nagy előnyének tekinthető.
→ Például: 1. vetület: WGS84 / 2. vetület: ETRS89 (UTM zone 32N)
- A grafikus felülete (*SpatiaLite GUI*) egyszerű, könnyen kezelhető és átlátható, viszont néha ez hátrányt is jelenthet. Például nem kínálja fel automatikusan az adattáblák, oszlopok, stb. neveit, mint pl. az *MS SQL Server* vagy a *PostgreSQL*.
- Sok különböző formátum (pl. *.shp*, *.geojson*, *.csv*, *.xls*) importálható, behívható.
- *PostgreSQL* adatbázis-szerver csatlakoztatható hozzá.

7.2. Az adatbázis importálása a Lovion BIS rendszerbe

A *SQLite-SpatiaLite* rendszer kapcsolata a *Lovion BIS* rendszerrel még nem teljesen megoldott. Annak érdekében, hogy egy *SpatiaLite* rendszerben létrehozott adatbázis működését közvetlenül tudjam tesztelni a *Lovion BIS* rendszerben, a fejlesztői csapattal hosszabb egyeztetésekre lenne szükség. A mostani szakdolgozatomban eredményessége érdekében, a *SpatiaLite GUI*-ban és *QGIS*-ben létrehozott *.sqlite* kiterjesztésű adatbázist egy *MS SQL Server* adatbázison keresztül kapcsoltam a *Lovion BIS* rendszerhez.

Ez a gyakorlatban azt jelentette, hogy a „*waste_management_styria_wgs84.sqlite*” nevű adatbázist át kellett alakítanom egy *MS SQL Server* adatbázissá, amelyet az 5.3. pontban leírtak alapján kapcsolattam a *Lovion BIS 6.6.5. Demomodell*-hez.

Az átalakítást a következők szerint hajtottam végre:

1. Az *SQL Server Management Studio 19 (SSMS)* grafikus felhasználói felületen (GUI) keresztül létrehoztam egy „*lovion*” nevű *MS SQL Server 2022-es* adatbázist.
2. Ezt az adatbázis szerveret csatlakoztattam a *QGIS*-hez a *Data Source Manager* szolgáltatásán keresztül.
3. A *QGIS* felületén keresztül importáltam az *MS SQL Server 2022* „*lovion*” nevű adatbázisába a „*waste_management_styria_wgs84.sqlite*” nevű adatbázis hét darab táblázatát.
4. Ellenőriztem az *MS SQL Server 2022* „*lovion*” nevű adatbázisába importált táblázatok adattartalmát. Egyebek mellett a végeredménynek számító

„b320_wastem_facilities” adattábla geometriai adatait tartalmazó oszlopait is ellenőriztem, ahol a következő SQL parancsokat használtam az *MS SQL Server* (1. SQL parancs), valamint a *Spatialite* (2. SQL parancs) rendszerekben:

<pre>select geom.STAsText() as position from dbo.b320_wastem_facilities</pre>	<pre>select ST_AsText(geometry) as position from b320_wastem_facilities</pre>
---	---

5. Az eredmények teljesen azonosak lettek mindkét adatbázis-kezelő rendszer esetében, így megállapítható, hogy az adatátvitel sikeresen lezajlott.

The screenshot shows two side-by-side query results. The left window is Microsoft SQL Server Enterprise Manager, and the right window is Spatialite. Both show the same data for the 'b320_wastem_facilities' table.

pk_uid	a_gem	betreiber	position
1	36	Liezen	Abfallwirtschaftsverband Liezen
2	51	Aich	Abfallwirtschaftsverband S...
3	183	Aigen/Ennstal	Gemeinde Aigen im Ennstal
4	187	Gröbming	Marktgemeinde Gröbming
5	203	Irdning-Donnersbachtal	Marktgemeinde Irdning-Donnersbachtal
6	225	Schladming	Manfred Arzbacher GmbH
7	231	Selzthal	Gemeinde Selzthal
8	243	Irdning-Donnersbachtal	Marktgemeinde Irdning-Donnersbachtal
9	254	Stainach-Pürgg	Marktgemeinde Stainach-Pürgg
10	256	Weissenbach b. Liezen	Gemeinde Weissenbach bei Liezen Orts- und Infrastrukt...
11	281	Ramsau/Dachstein	Gemeinde Ramsau am Dachstein
12	298	Lassing	Gemeinde Lassing
13	299	Obblam	Gemeinde Obblam
14	384	Schladming	Martina Sieder
15	390	Solk	Christian Tasch
16	401	Aigen/Ennstal	Gemeinde Aigen im Ennstal
17	411	Wörschach	Gemeinde Wörschach
18	449	Lassing	Gemeinde Lassing
19	450	Stainach-Pürgg	Wassergenossenschaft Stainach-Ennsboden
20	456	Haus	Marktgemeinde Haus
21	484	Stainach-Pürgg	Marktgemeinde Stainach-Pürgg
22	496	Ramsau/Dachstein	Marktgemeinde Ramsau am Dachstein
23	497	Irdning-Donnersbachtal	Marktgemeinde Irdning-Donnersbachtal
24	500	Selzthal	Gemeinde Selzthal
25	529	Irdning-Donnersbachtal	Marktgemeinde Irdning-Donnersbachtal
26	536	Liezen	Stadtgemeinde Liezen

7.5. ábra – A „b320_wastem_facilities” adattábla négy oszlopának megjelenítése *MS SQL Server* és *Spatialite* adatbázis-kezelő rendszerekben

6. Ezt követően készítettem egy .bak kiterjesztésű biztonsági másolatot (*Tasks / Back Up...*) az *SQL Server Management Studio* segítségével, „lovion.bak” néven.
7. Ezt a fájlt importáltam a *Lovion BIS 6.6.5. Demomodell* teszt-környezetet tartalmazó PC-n az 5.3. pontban említett, *Lovion BIS*-hez írt „DB tools” szkript segítségével.
8. Itt akadt egy kisebb problémám, amit szerencsére hamar sikerült megoldanom. Az otthon használt *SQL Server 2022 (16.x)* kompatibilitási száma 160-as volt. A *Lovion BIS 6.6.5. Demomodell*-hez kapcsolt *SQL Server 2019* kompatibilitási száma pedig 150-es. Így nem fogadta el a magasabb kompatibilitási számú adatbázist a teszt-modellhez kapcsolt instancia. Ehhez egy transzformációt kellett végrehajtanom, amiben az alábbi linkeken leríták segítettek:

- <https://learn.microsoft.com/en-us/sql/t-sql/statements/alter-database-transact-sql-compatibility-level?view=sql-server-ver16>
- <https://learn.microsoft.com/en-us/sql/relational-databases/databases/view-or-change-the-compatibility-level-of-a-database?view=sql-server-ver16>

9. A *Lovion BIS 6.6.5. Demomodell*-hez kapcsolt instancia („MSSQL2019”) esetében a „lovion” adatbázist átneveztem „waste_management_styria”-ra, az alábbi SQL parancs segítségével:

```
ALTER DATABASE lovion MODIFY NAME = waste_management_styria
```

10. Ezután szükséges volt módosítanom a *Lovion BIS 6.6.5. Demomodell*-hez kapcsolódó következő fájlokat:

- **LovionBIS_66.var** → új adatbázis meghívatkozása

```

63 <!--Work Management-->
64 <Entry key="db_material" value="material_demomodel" />
65 <Entry key="db_monitor" value="monitor_demomodel" />
66 <Entry key="db_task" value="task_demomodel" />
67 <Entry key="db_timesheet" value="timesheet_demomodel" />
68
69 <!--Waste Management Styria-->
70 <Entry key="db_wastem" value="waste_management_styria" />
71
72 <!--Netzmanagement-->
73 <Entry key="db_control" value="control_demomodel" />
74 <Entry key="db_measure_connect" value="measure_connect_demomodel" />
75 <Entry key="db_measure" value="measure_demomodel" />
76 <Entry key="db_outage" value="outage_demomodel" />
77 <!-- <Entry key="db_ranking" value="" /> -->
78 <!-- <Entry key="db_rating" value="" /> --> <!-- Replica in Demomodell -->
79 <Entry key="db_smarttag" value="smarttag_demomodel" />
80 <Entry key="db_supply" value="asset_supply_demomodel"/>
81 <Entry key="db_survey" value="survey_demomodel" />
82 <Entry key="db_switchboard" value="switchboard_demomodel" />

```

7.6. ábra – A „waste_management_styria” nevű adatbázis meghívatkozása a *LovionBIS_66.var* fájlban

- **repositories.xml** → a megjelenítendő adatbázis további rendszerspecifikus összetevőinek meghatározása

```

136 <DataModel path="WASTE_MANAGEMENT" name="WASTE_MANAGEMENT">
137   <RwoDefinition file="datadictionary.xml" />
138   <Enumerations file="enumerations.xml" />
139   <GuiStyle file="guistyle.xml" />
140 </DataModel>

```

7.7. ábra – A „waste_management_styria” nevű adatbázishoz kapcsolódó további rendszerspecifikus összetevők meghatározása *repositories.xml* fájlban

11. Továbbá szükséges volt létrehoznom a 24. ábrán látható *.xml* kiterjesztésű fájlokat (további kiegészítésekkel, mint például a *datamodel.xml*, stb.), valamint az azokat tartalmazó mappastruktúrát.
12. Több variációt és megoldást is kipróbáltam (mind a *LovionBIS_66.var*-ba, mind pedig a *repositories.xml*-be írandó tag-ek tekintetében), illetve többször újraépítettem az előzőekben írt fájlrendszert (teljesen eltérő tartalmú fájlokkal is), de a *Lovion BIS 6.6.5. Demomodell* sajnos ezek után sem ismerte fel az újonnan létrehozott adatbázist (*waste_management_styria*).
13. Ezt követően azzal próbálkoztam meg, hogy egy már korábban is létező adatbázisra (*asset_district_demomodel_665*) cseréltem le az újonnan épített adatbázist, így kijátszva az új adatbázis-környezet felépítéséből származó problémákat.
14. Ezen próbálkozásomat követően, a *Lovion BIS 6.6.5. Demomodell* indításakor az alábbi hibaüzenet jelentkezett:



7.8. ábra – Az „asset_district_demomodel_665” lecserélése után kapott hibaüzenet

Ez a hiba sajnos megakadályozott abban, hogy részletekbe menően tesztelni tudjam az újonnan létrehozott adatbázist a *Lovion BIS* rendszerben. Annak érdekében, hogy ezt a tesztelést el tudjam végezni, más megközelítési módszerre lenne szükség. Ezt majd csak a mesterképzésen (*UniGIS Salzburg*) írandó diplomamunkában fogom tudni megtenni. Ennek körülményeit részletesebben a következő – 8. pontban – fogom kifejtetni.

8. ÖSSZEFOGLALÁS, KONKLÚZIÓ

Úgy gondolom, hogy a 4. és az 5. részek kidolgozásával sikerült összeállítanom egy olyan összefoglaló jellegű dokumentumot, amely ez idáig csak több részletben, illetve nem teljeskörűen állt rendelkezésre magyar nyelven. A 4. rész összefoglalja a térbeli adatok kezelésére alkalmas relációsadatbázis-kezelő rendszerek sajátosságait, az 5. rész pedig bemutatja az üzemi információs rendszereket.

A 6. pontban sikerült nagyon jó eredményeket (illetve részeredményeket) elérnem, mivel sikerült megállapítanom a vizsgált relációsadatbázis-kezelő rendszerek erősségeit és gyengeségeit, továbbá arra is fény derült, hogy melyik rendszer milyen esélyekkel alkalmazható egy üzemi információs rendszerben. Ez a rész alapozta meg a gyakorlati feladat elvégzését is. Itt derült fény arra is, hogy a *PostGIS* miért nem alkalmas feltétel nélkül a *Lovion BIS* rendszerben való használatra és hogy milyen potenciális módosítások lehetnek szükségesek az abba való integrálás (együttes alkalmazás) érdekében.

A gyakorlati feladat által kitűzött céljaimat (7. pont) sajnos nem sikerült teljes mértékben megvalósítanom, de ennek ellenére több sikeres részeredményt értem el itt is. Több olyan részlet is szem elé tárult, amely hasznos abból a szempontból, hogy miként lehet használni az *SQLite-Spatialite* adatbázis-kezelőket a *Lovion BIS* rendszerben; illetve, hogy az *MS SQL Server* adatbázis-kezelő rendszeren milyen módosítások/változtatások végezhetőek ilyen szempontból.

Ennek ellenére, a siker elérése érdekében – azaz, hogy részletekbe menően tesztelni tudjak egy általam (*Spatialite* adatbázis-kezelővel vagy esetleg *PostGIS* adatbázis-kezelővel) épített/átalakított adatbázist a *Lovion BIS* rendszerben – más gyakorlati módszert és eljárást kellene választanom. Mint korábban már említettem, szeretném elvégezni a releváns mesterképzést a salzburgi *UniGIS* intézetnél. Az ott írandó diplomamunkámként mindenképpen a mostani szakdolgozatomat fogom folytatni, kiegészíteni, javítani és a gyakorlati feladat által kitűzött céljaimat egy másik módszer alapján – teljeskörűen – megvalósítani. Annak érdekében, hogy a tesztelést sikeresen tudjam elvégezni a *Lovion BIS* rendszerben, a diplomamunkám megírása során a következő munkafolyamatot fogom majd követni:

- A *Lovion BIS 6.6.5 Demomodel*-ből közvetlenül fogok exportálni egy – *MS SQL Server* típusú/formátumú – minta-adatbázist (például a „*asset_district_demomodel*”-t, mint itt korábban is tettem).
- Közvetlenül ezt az adatbázist fogom átkonvertálni *Spatialite* rendszerebe.
- Ezen az – már *Spatialite* típusú/formátumú – adatbázison fogok végrehajtani módosításokat (térbeli számításokat), illetve ezt fogom kiegészíteni új (általam létrehozott) táblázatokkal.
- Ezt a módosított adatbázist fogom visszakonvertálni a *MS SQL Server* rendszerbe (azaz azzal kompatibilisre) és ezt fogom közvetlenül visszaültetni (újból integrálni) a *Lovion BIS* rendszerbe.

Tehát ugyan azt az adatbázist fogom visszaültetni a rendszerbe, a releváns módosításokat és kiegészítéseket követően.

Továbbá szintén a mesterképzésen írandó szakdolgozatomban szeretnék foglalkozni a speciális, komplexebb geometriai formák (például.: Spline, NURBS, stb.) vizsgálatával is.

Úgy gondolom, hogy a saját kompetenciáim fejlesztése szempontjából több hozadéka is volt jelenlegi szakdolgozatomnak. Ezek közül a legfontosabbak a következők voltak:

- Sok gyakorlati tapasztaltot szereztem és még mélyebben megismerkedtem (mind a *Lovion BIS* használatával, mind pedig) azzal a metodikával, amely a *Lovion BIS* rendszert összeköt(het)i a különböző relációsadatbázis-kezelő rendszerekkel.
- Sikerült három relációsadatbázis-kezelő rendszer (*MS SQL Server*, *PostGIS*, *Spatialite*) és az azokhoz kapcsolódó kliens alkalmazások (GUI) használatát elsajátítanom.
- Sokat használtam gyakorlatban az SQL nyelvet, új lekérdezéseket és műveleteket tudtam tanultam meg.
- Tovább mélyítettem ismereteimet a *QGIS* térinformatikai program felhasználásával, illetve az adatbázis-műveletek térbeli megjelenítésével kapcsolatosan.

9. IRODALOMJEGYZÉK

- Márkus Béla, 2020: Műszaki Térinformatika (egyetemi jegyzet). Óbudai Egyetem, Alba Regia Műszaki Kar, Geoinformatikai Intézet. Székesfehérvár.
- Giachetta Roberto, Cserép Máté, 2018: Térinformatikai és távérzékelési alkalmazások fejlesztése – Térinformatikai adatok és adatformátumok (egyetemi jegyzet). Eötvös Loránd Tudományegyetem, Informatikai Kar. Budapest.
https://mcserep.web.elte.hu/data/education/2017-2018-2 TTAF/elte_ttaf_data-formats.pdf
- Nagy Gábor, 2010: Spatial Databases by Open Standards and Software 4. – The OGC and the Simple Features for SQL standard. University of West Hungary, Faculty of Geoinformatics. Székesfehérvár.
- Shipeng Sun, 2023: Data Analysis and Visualization with R: Spatial – Chapter 1 Introduction to spatial data in R. Department of Geography and Environmental Science, Hunter College. New York
<http://www.geo.hunter.cuny.edu/~ssun/R-Spatial/intro.html>
- Nagy Gábor, 2010: Spatial Databases by Open Standards and Software 5. – Basics of PostGIS. University of West Hungary, Faculty of Geoinformatics. Székesfehérvár.
- Nagy Gábor, 2018: Digitális domborzatmodellek és pontfelhők alkalmazása a terep modellezésében (doktori (PhD) értekezés). Óbudai Egyetem, Alkalmazott Informatikai és Alkalmazott Matematikai Doktori Iskola. Székesfehérvár.
- Nagy Gábor, 2019: Térinformatikai adatok tárolása adatbázisokban (egyetemi jegyzet). Óbudai Egyetem, Alba Regia Műszaki Kar, Geoinformatikai Intézet. Székesfehérvár.
- John R. Herring, 2011: OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture. Open Geospatial Consortium, Inc. Arlington, Virginia, USA.

- John R. Herring, 2010: OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 2: SQL option. Open Geospatial Consortium, Inc. Arlington, Virginia, USA.
- Kottyán László, 2010: Informatika 11. – Adatbázis szerverek (egyetemi jegyzet). Nyugat-magyarországi Egyetem, Geoinformatikai Kar. Székesfehérvár.
- Miljenko Lapaine, E. Lynn Usery, 2017: Choosing a Map Projection, Publication of the International Cartographic Association. Cham, Svájc
- Microsoft Learn weboldala, 2023: <https://learn.microsoft.com/>. Különböző aloldalak – az adott szövegrészeknél pontos linkkel megjelölve. Microsoft
- Gede Mátyás, 2014: Vektoros rendszerek – Térinformatikai Adatbázis rendszerek (egyetemi jegyzet). Eötvös Loránd Tudományegyetem, Informatikai Kar, Térképtudományi és Geoinformatikai Intézet. Budapest
<https://people.inf.elte.hu/ntb/vrend/index.html>
- Blahota István, 2013: SQLite alapok. Mobil alkalmazásfejlesztés az informatikai tudás innovatív alkalmazásával című, TÁMOP-2.2.4-11/1-2012-0055 kódszámú projekt keretében kiadott oktatási segédanyag. Nyíregyháza
- Oracle Corporation – Help Center: <https://docs.oracle.com/>. Különböző bejegyzések és szócikkek.
- Günther Klützke, 2021: Grundlagen der Versorgungswirtschaft (oktatási segédanyag). ITS Gruppe. Dortmund, Németország.
- Philipp Verhoeven, 2021: Übersicht Lovion Module (oktatási segédanyag). ITS Gruppe. Dortmund, Németország.
- Dr. Mike Gißrau, Roland Hardt, 2020: Workbook – Lovion BIS Administration I (oktatási segédanyag). ITS Gruppe. Dortmund, Németország.