

Földtani adatok kartografált, interaktív megjelenítése a weben open-source eszközök segítségével

Simó Benedek

Bevezetés

A hazai földtani adatok weben való publikálása a 2000-es évek második felétől indult meg a Magyar Állami Földtani Intézetben. Az első kísérletek még Geomedia környezetre támaszkodtak, majd egy szoftverváltást követően az adatok szolgáltatása már ESRI környezetben történt. A 2012-ben Magyar Földtani és Geofizikai Intézet (MFGI) néven létrejött kutatóközpont napjainkig ezt a technológiát használja elsődlegesen földtani térképeknek webes publikációjára.

A hardverek folyamatos fejlődésével és a nyílt forráskódú (open-source) szoftverfejlesztési modell elterjedésével a 2000-es évek közepétől új térinformatikai eszközök jelentek meg a piacon, az asztali szoftverektől a szerveroldali komponensekig (Neteler és Mitášová 2008). Ezek az eszközök mára igen jelentős szerepet töltenek be a modern térinformatikában, ezért érdemes megvizsgálni, hogy alkalmasak-e arra, hogy kiváltsák, vagy részben átvegyék a jelenleg működő rendszer feladatait.

Az alkalmazhatóság kutatása mellett több indokot is szükséges megemlíteni. Ezek közül az egyik egyértelműen gazdasági vonatkozású; a nyílt forráskódú rendszerek kialakításakor ugyanis nem szükséges beruházási költséggel és később szoftverkövetési díjjal számolni (bár természetesen nem szabad megfeledkezni az esetlegesen felmerülő kapcsolódó fejlesztési költségekről). A másik érv, amely az open-source rendszerek alkalmazása mellett szólhat, hogy szabadon testreszabhatóak, mely egyértelmű előnyt jelent a fizetős szoftverekkel szemben (Siki 2009). Az előző két általános érvelés szemben a harmadik sokkal inkább az MFGI-re vonatkozik: könnyebbé válhatna a teljesítendő feladatok végrehajtása számos olyan nemzetközi (az Európai Unió finanszírozásában létrejövő) projektben, amelyekben nyílt forráskódú eszközöket

alkalmaznak (a jelenlegi környezet használatával az ezekhez kapcsolódó munkafolyamatok sokszor nehézkesek vagy nem megoldhatóak). Egy nyílt forráskódú (open-source) WebGIS rendszer kidolgozása tehát gazdasági, rugalmassági és kompatibilitási előnyökkel is járhatna.

Jelen cikk egy általános WebGIS rendszer bemutatása után az MFGI-ben jelenleg működő rendszer jellegzetességeit vázolja, majd a rendszer nyílt forráskódú eszközökkel kialakítható alternatíváját és a megvalósításhoz szükséges migrációs lépéseket mutatja be, végül pedig a kutatás tanulságait foglalja össze.

WebGIS architektúra felépítés

Egy általános WebGIS rendszer a felépítését tekintve alapvetően hármas tagozódású. Az adattárolás egy térbeli adatok kezelésére alkalmas relációs adatbázisban történik. Erre az adattartalomra épülve egy szerveroldali komponens biztosítja ezek elérését (szabványos) webszervizek formájában. A rendszer harmadik összetevője a böngészőben történő interaktív megjelenésért és funkcionalitásért felelős; ez biztosítja a kartografált papírtérképekhez leginkább hasonló kimenetet a böngészőben.

Napjainkra szinte minden adatbázis-kezelő biztosít egy olyan adattípusú mezőt, melyben az egyes rekordokhoz kapcsolódóan geometriai információ eltárolására van lehetőség, szöveges vagy bináris formában. A fizetős szereplők világában az Oracle és a Microsoft SQL Server jár élen, míg az nyílt forráskódú eszközök között a PostgreSQL adatbázis-kezelő PostGIS kiterjesztésével biztosítja a leginkább sokrétű működést. (Mitchell 2005)

A szerveroldali modulok, vagyis térképszerverek világában is többféle nyelven íródott programokat találunk; ezek rendkívül széles skálán

mozognak a Microsoft .NET technológiájától kezdve (ArcGIS Server) az open-source eszközök világában előforduló Java (Geoserver) és C-ben (UMN Mapserver), valamint a C++ nyelven íródott (QGIS Server) implementációk mellett.

A kliensoldalon sokáig hódítottak a beépülő modulok segítségével operáló megoldások (ezek Adobe Flash Playert vagy Microsoft Silverlight technológiát használtak az interaktív megjelenítésre), de arányuk mára jelentősen lecsökkent, és a Javascript nyelv egyre hatékonyabb működése és ezzel párhuzamosan napjaink modern böngészőinek megjelenése egyre inkább a tiszta Javascript kódra épülő könyvtárak népszerűsödését eredményezte. Mind a fizetős, mind az open-source világban a jelenleg legnépszerűbb függvénykönyvtárak (ArcGIS Javascript API és Openlayers3 API) ezzel a technológiával készültek. (Dhokal 2016)

A jelenleg működő rendszer

A webes publikálásra szánt földtani adatok jellegüket tekintve alapvetően vektoros adattárolási struktúrával rendelkeznek; típusukat tekintve lehetnek pontszerű térképek (fúrásokat, mérési adatokat tartalmazó ponttérképek), vagy felületi kiterjedéssel rendelkező állományok (felszíni földtani, vagy aljzattérképek), de vonal menti tematikák is előfordulhatnak (pl. izovonalas ábrázolású térképek).

Az adattárolás, térképszerver, kliensoldali megjelenítés hármas egysége ennek megfelelően alakul az MFGI-ben, mely jelenleg az ESRI termékcsalád egyes komponenseire épül. A térbeli adatok kezelése Microsoft SQL Server segítségével történik, melyet az ESRI ArcSDE eszköz tesz teljes értékű térinformatikai adatbázis-kezelővé. Ezekre az adatbázisokra épülve történik a földtani térképek szerkesztése, mely eredményeképpen

egy megjelenítést tartalmazó bináris leíró fájl jön létre (mxd kiterjesztéssel). Ezt felhasználva az ArcGIS Server képes egy webes szolgáltatást indítani, mely tartalmazza a kartografálás során alkalmazott megjelenésre vonatkozó információkat. A webes megjelenítés során a kliensoldalon (vagyis a böngészőben) futó ArcGIS Javascript API-ra épülő kód ezeket a szervereket hívja meg az egyes alkalmazásokban.

Ennek a jelenlegi rendszernek nagy előnye, hogy az adatok kartografálását követően az elkészült állományok – programozás vagy szöveges fájlok manuális módosítása nélkül – publikálhatók az interneten, méghozzá pontosan azzal a jelkulccsal, amivel azt az asztali környezetben kialakították. Az open-source eszközök egyik legnagyobb kihívása talán pont ez lehet: a különféle projektek által létrehozott WebGIS komponensek sokszor nehezen kapcsolhatóak össze az asztali szerkesztéstől a webes megjelenítésig, és a stílusok átörökítése sokszor egyáltalán nem lehetséges, vagy sok manuális munkát igényel.

Az open-source eszközökkel kialakítható rendszer

Mivel a nyílt forráskódú eszközökkel egy WebGIS rendszer teljes vertikumában sokféleképpen felépíthető, így elsőként meg kell vizsgálni, hogy milyen alapvető elvárásoknak kell megfelelnie a kialakítandó rendszernek; figyelembe kell venni a térinformatikai adattárolás módját és a kialakult munkafolyamatokat. Ahogyan az előző részben az már kiderült, az MFGI-ben a feldolgozott, kész térinformatikai adatok tárolása adatbázisokban történik, majd erre épül a térképek szerkesztése, és a webes publikálás kartográfiai szempontból magas színvonalon történik. Így elsősorban azt szükséges megvizsgálni, hogy a nyílt forráskódú szoftverek sokaságából össze lehet-e állítani egy ilyen igényeket kielégítő rendszert, amely hordozza az ESRI termékekre épülő vertikumhoz hasonló minőségi és kényelmi jellegzetességeket.

A nyílt forráskódú GIS világában talán az adatbázis-kezelők közötti választás a leginkább kézenfekvő.

Habár több adatbázis-kezelő is rendelkezik valamilyen szintű funkcionalitással térinformatikai adatok kezelésére (MySQL, MongoDB, SQLite), mégis kiemelkedik közülük a PostgreSQL, a PostGIS kiterjesztéssel. (Chen 2008) Ez a projekt nemcsak széles funkcionalitással rendelkezik (a benne elérhető térbeli adatokra alkalmazható függvények száma jelenleg ezer felett jár), hanem aktív felhasználói és fejlesztői közösséggel rendelkezik, így mára szinte szabványnak tekinthető az open-source világban. Nem kérdés tehát, hogy az MFGI esetében ezzel a választással jó alapot lehet teremteni a kialakítandó WebGIS rendszernek.

A rendszer szerveroldali komponense, vagyis a térképszerkesztés választásakor számos kérdés merül fel. A legnépszerűbb projektek, mint a Geoserver és a UMN Mapserver esetében kijelenthető, hogy habár széles funkcionalitással rendelkeznek, és teljesítmény szempontból is erős választásnak bizonyulnak, legnagyobb hiányosságuk az asztali környezetben kartografált térképek stílusainak az átörökítésekor merül fel. Képesek ugyan bonyolultabb stílusok megjelenítésére, de ehhez sokszor komoly manuális munka szükséges, és a végeredmény kartográfiai szempontból még így is sokszor limitált, elsősorban a névrajzi, valamint a felületi kitöltések tekintetében.

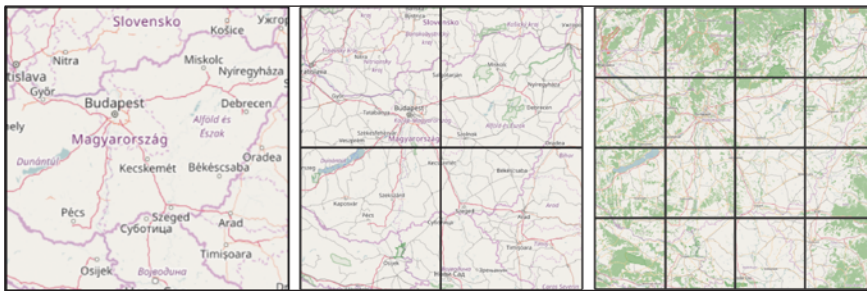
Ezen a ponton szükséges megemlíteni, hogy habár egy WebGIS rendszer esetében nem tekinthető egy asztali szerkesztéshez használt program a rendszer szerves részének, a kartográfiai szempontokat szem előtt tartva ez az elem mégis kiemelt fontossággal bír a rendszer szempontjából. Az MFGI esetében elkészült komplex, kartografált földtani tematikájú térképek, melyek jelkulcsa sokszor igen nagy összetettséget mutat, megkívánja egy ilyen elem integrálását a rendszerbe. A QGIS Desktop, mint asztali GIS-szoftver, képes összetett kartográfiai megjelenítés kialakítására, így megfelelő lehet erre a feladatra, hiszen a nyílt forráskódú GIS-világban a vezető szerepe a PostgreSQL adatbázis-kezelőhöz hasonlítható, és versenytársai minden szempontból elmaradnak mögötte. Ha ez az elem is integrálásra

kerül a rendszerbe, akkor ezek után a legnagyobb kérdés tehát a következő: hogyan lehetséges az ezzel a szoftverrel elkészült térképeket ugyanezzel a jelkulccsal a webre közvetíteni?

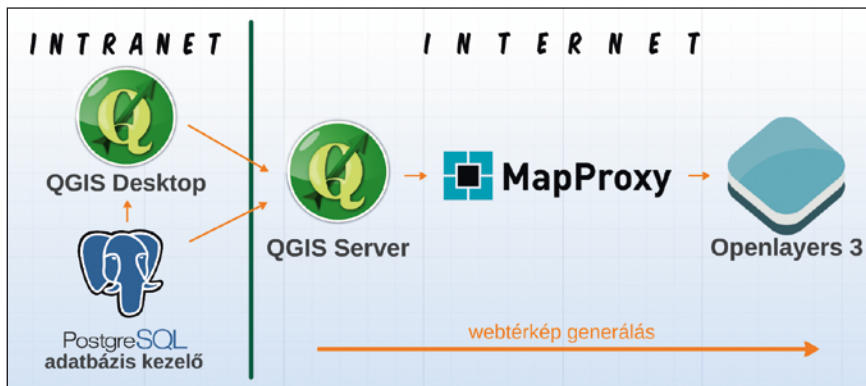
Ahogy a korábban már említésre került, a Geoserver és a UMN Mapserver kötöttek saját, szöveges alapú megjelenítésükhöz, így ezek nem alkalmasak a feladatra. A problémára a megoldást a QGIS Server nyújthatja, melynek legnagyobb előnye, hogy a webes térképek generálásához ugyanazt a renderelési mechanizmust használja, mint a QGIS Desktop, és így képes az asztali környezetben kialakított stílus tökéletes reprodukciójára. A munkafolyamat így az ESRI-alapú rendszerhez hasonlóvá válhat; a QGIS Desktopban szerkesztett térképi állomány elkészültekor keletkező leíró XML-fájl alapján a QGIS Server képes az asztali térképek webes publikálására, szabványos Web Map Service, vagyis WMS formában.

Ha a terheléelosztási vagy skálázhatósági szempontokat is figyelembe vesszük, akkor a QGIS Server talán nem mérhető össze a Geoserver vagy a UMN Mapserver teljesítményi paramétereivel, de ez a probléma egy ún. tile-osítási mechanizmussal kiküszöbölhető. Ennek lényege, hogy egy kiindulási méretarányérték felezésével létrejövő méretarány skála elemeihez készül egy-egy vetületi négyzethálórács, és a kartografált térkép ezekre eső elemeit tárolódnak le fájl szinten, rasteres formában. Ezzel mind a szerver terhelése, mind pedig a hálózati adatforgalom csökken, és ezzel párhuzamosan javul a kliensoldalon a felhasználói élmény. (Loechel et al. 2012)

A többféle tile-osításra alkalmazható eszköz közül a Mapproxy az, amely a legszélesebb funkcionalitással rendelkezik (egyedi vetületek kezelése, csak adattartalommal rendelkező területek figyelembe vétele, vagyis az üres tile-okat nem generálja le többször), szabványos adatszolgáltatásokat támogat (WMTS - Web Map Tiled Service, és TMS - Tiled Map Service), és emellett folyamatos aktív fejlesztés alatt áll, ezért jó választás lehet. A QGIS Serverrel közösen a rendszer így képes a földtani térképek szolgáltatására a webre, szabványos formában, így már



1. ábra. A tile-osztás („csempekészítés”) sematikus ábrája



2. ábra. Az MFGI esetében alkalmazható nyílt forráskódú WebGIS rendszer

csak a kliensoldali megjelenítés eszközt kell a rendszerhez integrálni.

A kliensoldali megjelenítésre hivatott (Javascript nyelven íródott) eszközökben mutatkozik talán a legnagyobb sokféleség a WebGIS világában. Rengeteg kisebb, egyszerűbb feladatra alkalmas eszköz érhető el szabadon, azonban két nagyobb szereplő kiemelkedik közülük. Egyszerűbb feladatok megoldására alkalmas lehet a letisztult struktúrával rendelkező Leaflet eszköz, de több hiányossága (szűkösebb adattípus-támogatás, térkép elforgathatóságának hiánya) mellett például nem alkalmas EOV-vetületben megjeleníteni a térinformatikai adatokat, amely az MFGI WebGIS rendszerének kialakításakor alapvető igénynek bizonyul. A másik eszköz a 2014-es év második felére alapjaiban újraírt Openlayers 3-as verziója, mely teljesen új alapokra helyezte a kliensoldali megjelenítést. A világ szinte valamennyi vetületét képes kezelni, számtalan adatformátumot támogat, és emellett tudja hasznosítani a modern böngészők által implementált WebGL renderelési mechanizmust, mely által a megjelenés minden korábbinál gyorsabban lehetséges. (Santiago 2015) Az Openlayers 3-as verziójával lehetséges a Mapproxy eszköz által szolgáltatott TMS vagy WMTS formában szolgáltatott adatok megjelenítése,

míg az adatokhoz való hozzáférés a QGIS Server vagy a PostgreSQL elérésével lehetséges aszinkron kérések (AJAX) segítségével. Utóbbi lényege, hogy az adott oldal frissítése nélkül lehetséges az adatok cseréje és frissítése a szerverrel, így az alkalmazás képes a szerveroldalról dinamikus adatfrissítésre.

Végeredményben kijelenthető, hogy az MFGI webes térképszolgáltatási céljainak egy olyan rendszer felelhet meg leginkább, melyben az adatok tárolása PostgreSQL környezetben történik, erre épülve zajlik a térképek szerkesztése QGIS Desktop környezetben, majd az eredmény QGIS Server, illetve Mapproxy segítségével válik elérhetővé webes szerverek formájában, és ezeket az Openlayers komponens teszi interaktívan kezelhetővé a böngészőben.

Migrálás a nyílt forráskódú rendszerre

Egy WebGIS rendszer kialakítása bonyolult feladat lehet abban az esetben, hogyha már létező adatbázisokra és kész térképekre szükséges felépíteni azt. Az MFGI esetében is a meglévő rendszer migrációs lehetőségeinek a vizsgálata volt az első és legfontosabb feladat; ennek egyik eleme az

adatbázisok migrációja, másik pedig a kialakult jelkulcsok átültetése volt.

Az adatbázisok migrációjakor figyelembe kellett venni, hogy a geometria ezekben a táblákban ún. sdebinary formátumban tárolódik, nem pedig az MS SQL Server geometry típusú mezőjében, mivel ezzel a formátummal nem tud hatékonyan együttműködni az ESRI desktop szoftvere, az ArcMap. A migrációhoz ezért elsőként az adatokat egy olyan adatbázisba szükséges másolni, amely az MS SQL Server geometry típusú téradatait tárolásához alkalmazza, erre az ESRI saját eszközei (tooljai) is alkalmasak (akár a Migrate Storage, akár a Copy Features tool). Ezután következhet az adatok migrálása egy open-source adatbázisba, PostgreSQL környezetbe. Ehhez az open-source világ egyik legalapvetőbb eszköze, az ogr2ogr használható fel, mely számos nyílt térinformatikai formátum között biztosít konverziós lehetőséget. Mivel az ArcGIS környezet biztosítja saját toolok fejlesztésének lehetőségét (Python nyelv segítségével), így az ogr2ogr eszköz bevonásával lehetséges egy olyan konverziós eszköz fejlesztése, mely az ArcSDE forrásból átviszi az adatokat PostGIS-szel támogatott PostgreSQL környezetbe.

A jelkulcsi elemek sikeres migrációjához két tényező nyújt segítséget. Egyrészt az MFGI-ben korábban kialakult egy gyakorlat, mely adatbázisra tárolja el az egyes adatbázisok kartografálásához szükséges információkat (Maigut 2004, 2005). Mivel a földtani térképekre elsősorban a felületi jelek jellemzők (Galambos 2004), így ezeknek a szinkódjai megtalálhatóak egy központi jelkulcsadatbázisban (Galambos és Simonyi 2006), melyet a migrációhoz fel lehet használni. A másik tényező, hogy az előző részben tárgyalt nyílt forráskódú WebGIS rendszerhez kapcsolt QGIS Desktop, mint asztali térinformatikai eszköz, képes egy szöveges stílusinformációkat tartalmazó leíró XML-állomány alapján megjeleníteni térinformatikai adatokat. Ezért lehetséges egy olyan eszközt fejleszteni, amely egy megadott XML-struktúrába menti le a központi jelkulcsadatbázisban található értékeket egyes tematikákhoz, mely ezután kartografált módon megjelenik a QGIS-eszközben.

Az adattartalom és a jelkulcsok migrációja tehát egy megoldható és automatizálható feladat, mellyel nagyban megkönnyíthető a korábban vázolt nyílt forráskódú rendszerbe való esetleges átállítás, mely így egy reális opció lehet az MFGI számára.

Összefoglalás

Jelen cikk bevezetőjében az olvasó betekintést nyerhet egy általános WebGIS rendszer felépítésébe, mely után bemutatásra kerül az MFGI-ben jelenleg használt, ESRI termékeken alapuló térképpublikálási technológia. Az írás második fele arra a kérdésre igyekszik választ adni, hogy ugyanerre a térképpublikálási célra lehetséges-e egy open-source eszközökből álló rendszert felépíteni, mely teljes értékű alternatívája lehet a most működő architektúrának. Az új rendszer lehetséges kialakításának a részletezése, valamint a szükséges migrációs lépések taglalása alapján kijelenthető, hogy a vázolt megoldás megvalósítható és beilleszthető az MFGI térinformatikai struktúrájába, mely növeli az Intézet rugalmasságát a WebGIS feladatainak megoldásában.

A vázolt kutatómunka eredményein alapulva a 2016-os évben elkezdődött az architektúra gyakorlati megvalósítása az MFGI-ben, amely a belső teszteléseket követően várhatóan 2016 végére publikusan is elérhető lesz majd. A nyilvános megjelenésig elsősorban a webes alkalmazás funkciókészletének a bővítése lesz a legnagyobb feladat, és ezzel párhuzamosan történik majd a migrációs eszközök első stabil verziójának a kialakítása is. Egy teljes tematika publikálásának tapasztalatai majd jó alapot nyújthatnak egy későbbi döntéshozásra

a nyílt forráskódú rendszer alkalmazását illetően, mely természetesen stratégiai döntés kell hogy legyen, és további alapos elemzést igényel az elérni kívánt célok tükrében. Mivel a kétféle architektúra mindenképpen növeli az Intézet stabilitását, így a rövidtávú cél a két rendszer egymás melletti működtetése, de hosszabb távon (az esetleges pozitív tapasztalatok tükrében) a kizárólagos alkalmazás lehetősége is megfontolandó lehet majd a jövőben.

Irodalomjegyzék

- Chen R., Xie J. (2008): Open Source Databases and Their Spatial Extensions. In: Advances in Geographic Information Science, Springer, pp. 105–129.
- Dhaka, S. (2016): Why should GIS professionals make a switch to javascript based Web Appbuilder (WAB) from Flex/Silverlight based Map Viewer? <https://www.linkedin.com/pulse/why-should-gis-professionals-make-switch-javascript-based-dhaka>
- Galambos Cs. (2004): Földtani térképek felületi jelei. Geodézia és Kartográfia 56(7): pp. 16–21.
- Galambos Cs., Simonyi D. (2006): Földtani térképeken alkalmazható színadatbázis és felületjel-készlet. In: Balla Z. (ed.): A Magyar Állami Földtani Intézet Évi Jelentése 2005, Magyar Állami Földtani Intézet, Budapest, pp. 193–198.
- Loechel, A. J., Schmid, S. (2012): Caching techniques for high-performance Web Map Services. Proceedings of the AGILE'2012 International Conference on Geographic Information Science, pp. 52–57.
- Maigut V. (2004): Új, digitális földtani alpmű a MÁFI-ban. Geodézia és Kartográfia 56(7): pp. 22–26.
- Maigut V. (2005): Földtani térképek kartografálásának segítése térinformatikai módszerekkel. In: Balla Z. (ed.): A Magyar Állami Földtani Intézet Évi Jelentése 2004, Magyar Állami Földtani Intézet, Budapest pp. 139–144.
- Mitchell T. (2005): Web Mapping Illustrated, O'Reilly, pp. 241–275.
- Neteler, M., Mitášová, H. (2008): Open Source GIS: a GRASS GIS approach. Springer, New York, 424 p. URL: <http://www.grassbook.org>

- Santiago, A. (2015): The book of Openlayers. Leanpub
- Siki Z. (2009): Produktív környezetben használt, nyílt forráskódú komplex térinformatikai megoldások Prezentáció, CASCADOSS műhelymunka-tanácskozás és GRASS tanfolyam, Szeged 2009. január 27–30. URL: http://www.agt.bme.hu/gis/eloadasok/siki_szeged2009sz.pdf (Elérve: 2016. november 2.)
- Thakur, J. K., Singh, S. K., Ramanathan, A., Prasad, M. B. K., Gossel, W. (2012): Geospatial Techniques for Managing Environmental Resources, Springer, pp. 105–111.
- Turczai G. (2005): Földtani térmodell építése – adatbázisok az intra- és interneten. In: Balla Z.: A Magyar Állami Földtani Intézet Évi Jelentése 2004, Magyar Állami Földtani Intézet, Budapest pp. 125–130.

Summary

Visualizing Geologic Data on the Web with Open Source Software Solutions

The Geological and Geophysical Institute of Hungary currently using the ArcGIS software stack for serving geospatial data and interactive maps towards the web. Since the open-source software products could be a feasible alternative, it is important to test its capabilities and the possible migration steps needed. The article demonstrates that with focus on high cartographic standards, it is possible to construct a system for this goal; with PostgreSQL as a RDBMS, QGIS Server as a server side component, Mapproxy as a caching tool and with OpenLayers as the client side for interactivity.



Simó Benedek
tudományos
segédmunkatárs

Magyar Földtani és Geofizikai Intézet

Tájékoztatjuk kedves olvasóinkat, hogy a Magyar Földmérési, Térképészeti és Távérzékelési Társaság programjairól, híreiről rendszeresen tájékozódhatnak honlapunkon is.

www.mfttt.hu

MFTTT vezetősége

