

# Synchronizace a replikace geodat v prostředí Esri platformy

Markéta Solanská

Katedra geoinformatiky, Přírodovědecká fakulta, Univerzita Palackého v Olomouci,  
17. listopadu 50, 779 00 Olomouc, Česká republika,  
marketa.solanska@gmail.com

**Abstrakt** Tato práce hodnotí možnosti dostupných replikačních řešení a na základě toho navrhuje databázové řešení s ohledem na možnosti a požadavky katedry. V rešerší části byly vymezeny pojmy synchronizace, replikace a související pojem verzování a popsána replikace včetně variant synchronní, asynchronní, jednosměrné, obousměrné, kaskádové, logické i fyzické. Byly rozebrány požadavky na databázové ukládání dat jednotlivých produktů ArcGIS a byla podrobně popsána technologie ArcSDE, která se v ArcGIS produktech používá pro připojení k databázi. Na základě rešerše byl vybrán databázový systém PostgreSQL, který je možno použít v kombinaci s produkty ArcGIS, což bylo jedním z hlavních požadavků pro výběr databázového systému. Byl sestaven návrh databázového řešení, který zohledňuje všechny požadavky katedry a možnosti daných technologií. Bylo vytvořeno testovací prostředí na serveru poskytnutém katedrou, na němž byly dané procesy otestovány. Na základě toho byl pak sepsán podrobný popis toho, jak nastavit replikaci ve variantě streaming a Slony-I. Návrh zahrnuje také možnost použití nástroje pgspool pro rozložení zátěže mezi servery v databázovém clusteru.

**Klův slova:** replikace, synchronizace, verzování, databázový systém, PostgreSQL, ArcSDE, ArcGIS

**Abstract.** The main goal of this thesis is to evaluate options of replication solutions which are available and based on this research design a database solution which considers possibilities and requirements of the Department of Geoinformatics. In the theoretical part terms replication, synchronization and versioning are defined including description of synchronous, asynchronous, master-slave, multimaster, cascade, logical and physical replication. The requirements of ArcGIS products for storage of data in database were considered and ArcSDE Technology which is used by ArcGIS products for database storage of spatial data was described. Based on the research database management system PostgreSQL was chosen because it is supported by ArcGIS products. The design of the database solution was created based on all requirements and the main processes were tested. Based on that a manual of the proposed replication solution setup was written. Two replication options were tested

- PostgreSQL native streaming replication and replication using PostgreSQL extension Slony-I. The design includes a description of usage of pgpool utility used for load-balancing.

**Keywords:** replication, synchronization, versioning, database management system, PostgreSQL, ArcSDE, ArcGIS

## 1 Úvod

Dnešní trend je ukládat a ponechávat stále více dat pouze v digitální podobě. Mnoho dokumentů už se vůbec netiskne do papírové podoby, což podporuje i trend elektronických schránek a podpisů. S přibývajícím množstvím dat je však třeba řešit komplikace, které informace uložené pouze v elektronické podobě přinášejí. Počítačové experti řeší například otázky, kam ukládat tak velké množství dat, jak data efektivně aktualizovat, jak zabránit poškození dat až už způsobených lidským faktorem či chybou hardware. V případě, že se poškodí disk, můžeme často během okamžiku přijít o všechna data, někdy však pro ztrátu dat stačí pouze stisknout tlačítko na klávesnici.

Vhodným způsobem uchovávání dat je ukládání do databáze s následnou replikací. Replikací je myšlena pokročilá funkcionální, která zajišťuje kopii dat na více serverů. Nabízí ji většina dnešních databázových serverů, zajišťuje větší robustnost databáze a vysokou dostupnost dat. Replikaci lze využít ve všech odvětvích, která pracují s daty. Výjimkou není ani geoinformatika, která často pracuje s velkými objemy dat, které nesou informaci o geografické poloze. Právě reprezentace geografické polohy, skrze textový zápis souřadnic daných bodů, může způsobit razantní zvýšení objemu dat.

U webových map se musí řešit velký počet dotazů do databáze, protože například každé posunutí výřezu či přiblížení, resp. oddálení výřezu mapy, je samostatným dotazem, který musí kapacita serveru zvládat. Například pokud bude uživatel procházet plánovanou 100km trasu posouváním výřezu mapy po 10 km, může to serveru způsobit velkou zátěž. Replikaci ocení uživatelé pracující na společném projektu, distribuovaná pracoviště i společnosti s velkým množstvím důležitých dat, jejichž dostupnost je rozhodující pro jejich fungování.

## 2 Použité metody a programové komponenty

Konfigurace replikace zahrnovala studium návodů jednotlivých nástrojů pro replikaci, výběr vhodných programových komponent a jejich následné praktické nastavení. To bylo testováno průběžně na několika počítačích.

Jako databázový server byl zvolen *PostgreSQL* s plnou podporou pro správu prostorových dat, která je zajištěna nábavbou *PostGIS*. Pro replikaci byla zvolena nativní *PostgreSQL streaming replikace* a externí nástroj *Slony-I*. Pro efektivní využívání databáze byl dále vybrán externí nástroj *pgpool*, který zajišťuje snížení zátěže jednotlivých serverů rovnoměrným rozkládáním dotazů od klientů mezi jednotlivé databáze.

Nástroj pro replikaci Slony-I byl testován na operačním systému Ubuntu GNU/Linux 12.4 a zároveň na operačním systému Windows XP.

Nativní PostgreSQL streaming replikace byla testována pouze na operačním systému Linux. Server geohydro.upol.cz byl poskytnut jako testovací server pro tuto práci. Na server byl nainstalován 32bitový operační systém Debian GNU/Linux 7.3, který byl vybrán kvůli jeho stabilitě a jevil se tedy pro server jako vhodný. Tato verze ovšem umožnila instalaci pouze programů verzí PostgreSQL 9.1, PostGIS 1.5 a pgpool 3.1. Vzhledem k tomu, že se nejedná o nejnovější verze zmíněných produktů, byla replikace testována také na osobním počítači ve verzích PostgreSQL 9.3, PostGIS 2.1 a pgpool 3.3. To umožnilo nastudování dalších možností, které nové verze přináší a které byly zohledněny v návrhu replikačního řešení.

Pro testování byla používána ukázková prostorová data vytvořená pro účel této práce a dále byla na server uložena datová sada ArcČR ve verzi 3.0.

### 3 Vymezení pojmů

Databáze je strukturovaná kolekce dat, která slouží pro efektivní ukládání dat a jejich zpětné čtení [1]. V relační databázi jsou data ukládána ve formě tabulek, tedy entit a atributů, které jsou vzájemně propojeny vazbami mezi entitami [2]. Toto logické uložení vazeb mezi tabulkami umožňuje efektivní manipulaci s daty, rychlé vyhledávání i komplexní analýzu [3].

Obvykle se rozlišují pojem databáze, který odkazuje na obecný koncept, a pojem databázový systém nebo přesněji *systém řízení báze dat*<sup>1</sup>, což je konkrétním počítačovým program, který zajišťuje fyzické uložení dat. Moderní SRBD jsou navrženy na principu klient/server, kdy databáze běží jako služba na pozadí a čeká na dotazy od klientů. Server uživatelům umožňuje skrze jazyk SQL přistupovat k databázi, vytvářet a aktualizovat data, stejně jak jako vyhledávat či analyzovat [2].

*Prostorová databáze*, někdy také zvaná *geodatabáze*, není nic jiného než databáze obohacená o datový typ určený pro ukládání prostorové informace o prvku, prostorové indexy a sadu funkcí vhodných pro správu prostorových dat. Dnes umožňují ukládat prostorová data například databázové systémy PostgreSQL 9.x, Microsoft SQL Server, Oracle Database, MySQL nebo SQLite.

Pojmy replikace a synchronizace některé zdroje rozlišují, jiné je naopak považují za synonyma. Všechny zmíněné pojmy souvisí se zálohováním dat, tedy kopírováním dat mezi dvěma a více uložisti, a se liší konkrétním důvodem pro použití daného procesu.

O synchronizaci souborů či datových složek je možno mluvit v případě, že existují dva datové zdroje, které je potřeba v daný okamžik sjednotit. Jde tedy o proces, který probíhá jednorázově a to většinou z důvodů potřeby porovnání dvou a více datových uložišť, které je potřeba dostat do totožného stavu. To může například přispět snazší spolupráci více uživatelů nad stejnými daty nebo pomoci

---

<sup>1</sup> angl. Database Management System (DBMS)

uživateli, který pracuje na více počítačích. Proces může proběhnout jednou nebo opakovaně, ať už pravidelně či nepravidelně. U souborů se shodným názvem se porovnává čas posledního zápisu, velikost nebo obsah souboru, naopak soubory, u kterých není nalezena shoda, jsou jednoduše zkopírovány.

Replikace je proces průběžný, který soustavně hlídá, zda ve zdrojových datech nedošlo ke změně, a pokud ano, dané změny zkopíruje na jiné datové uložště. Často je tento proces používán právě ve spojitosti s databázemi, kdy jsou data kopírována z důvodu snížení zátěže serveru, či zvýšení ochrany dat. Replikace je tedy často vyžadována z jiných důvodů než synchronizace, začíná s daty existujícími pouze na jednom uložšti a pro zajištění konzistence dat používá jiných technologií. Více se replikací zabývá kapitola 3.1.

Oba procesy je možno použít jednostranně, tedy kopírovat data pouze z jednoho uložště na druhý a nikoliv opačně, nebo oboustranně, kdy se data kopírují navzájem mezi sebou.

### 3.1 Replikace

Replikace je proces, u kterého jsou data a databázové objekty kopírovány z jednoho databázového serveru na druhý a poté synchronizovány pro zachování identity obou databází. Synchronizací je v tomto případě myšleno kopírování všech změn, které v databázi nastanou. Použitím replikace je možno data distribuovat na různě vzdálená místa nebo mezi mobilní uživatele v rámci počítačové sítě a internetu [4].

Vývojáři mnohých moderních aplikací se musí zabývat přetížením serveru způsobených velkým počtem současných přístupů do databáze. V případě přetížení se prodlouží odezva serveru, data tedy přicházejí k uživateli pomalu, nebo server dokonce úplně spadne.

Mezi časté důvody použití databázové replikace tedy patří zajištění dostupnosti dat<sup>2</sup>, resp. snížení pravděpodobnosti, že data nebudou dostupná [5]. Další důvodem je rozložení přístupů do databáze mezi více serverů, takže nebude docházet ke zpomalení výkonu hlavního serveru [6]. Ke zpomalení serveru dochází také při zálohování, což lze řešit replikací dat na jiný server, na kterém je pak proces zálohování spuštěn.

Všechny databázové servery zapojené do procesu replikace jsou v odborné literatuře nazývány *uzly*, angl. *node*. Tyto uzly dohromady tvoří *replikační cluster*<sup>3</sup>. Při správně nastavené replikaci, jejímž cílem je zajištění vysoké dostupnosti dat (HA), by v clusteru nikdy neměly být méně než tři uzly. Může se totiž stát, že vypadne jeden ze dvou uzlů, čímž dojde k situaci, že data v daný okamžik nebudou zálohovaná.

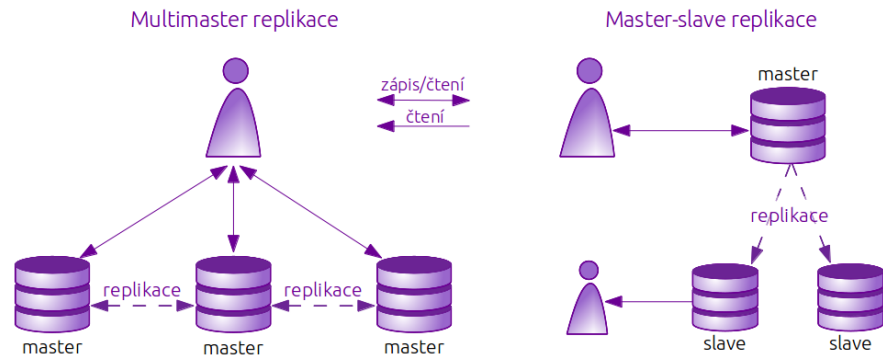
Uzly v replikačním clusteru mohou mít jednu ze dvou základních rolí, nejčastěji nazývaných *master* a *slave*. Master server nebo pouze master je server, který poskytuje data k replikaci, má práva na čtení i zápis a probíhají tedy na něm veškeré aktualizace. Je možno se setkat také s pojmenováním *primary server*,

<sup>2</sup> angl. High Availability

<sup>3</sup> volně přeloženo jako skupina serverů zapojených do replikace

provider, sender, parent nebo source server. Naprosto jiný pojem zavádí MS SQL Server, který tento zdrojový server nazývá publisher (česky vydavatel). Druhý databázový server je nejčastěji nazýván slave, standby, receiver, child nebo subscriber (česky odběratel). Poslední pojem je také používán MS SQL Serverem. Na tento server, který je dostupný vždy jen pro čtení dat, se data kopírují, není však možné na něj změny zapisovat přímo [7].

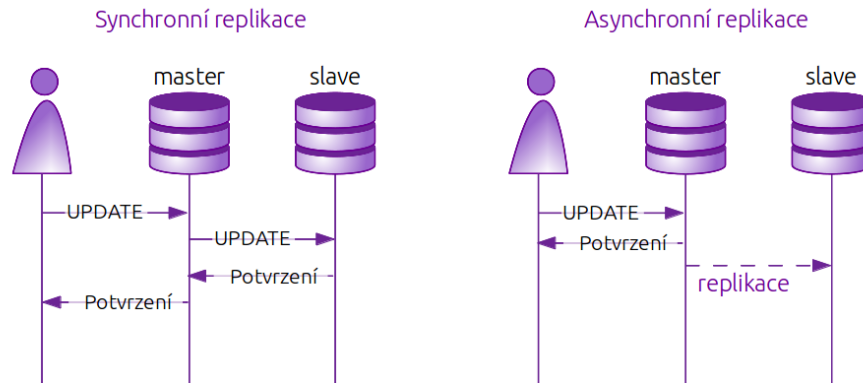
Podle počtu master a slave serverů v replikačním clusteru se rozlišuje, zda se jedná o jednosměrnou nebo obousměrnou replikaci. U tzv. *multimaster* replikace existuje v replikačním clusteru několik master serverů, tedy těch na které se změny zapisují přímo. To je praktické například ve chvíli, kdy je i samotných zápisů tolik, že jeden server tuto zátěž neunes. Zápisy z jednotlivých master serverů se tedy nereplikují pouze na slave servery, ale také na všechny ostatní mastery. Tento způsob s sebou však nese značné komplikace, je potřeba řešit konflikty změn v rámci stejných záznamů, a je tudíž relativně náročný na údržbu. Tato práce se zabývá použitím druhé způsobu, tzv. *master-slave* replikace. Tato replikace používá vždy jen jeden master server v clusteru a dva a více slave servery. Kopie dat tedy probíhá jednosměrně, vždy z master na slave servery. Podle Bella a kol. (2010) mají moderní aplikace často více čtenářů než zapisovatelů, proto je zbytečné, aby se všichni čtenáři připojovali na stejnou databázi jako zapisovatelé a zpomalovali tím jejich práci [6].



Srovnání multimaster a master-slave replikace

Při návrhu replikace je potřeba se zamyslet také nad tím, zda bude *synchronní* či *asynchronní*. Synchronní replikace neumožní potvrzení transakce modifikující data, dokud všechny změny nejsou přeneseny alespoň na jeden slave server [8]. Tento přístup zajistí, že žádná data nebudou v průběhu zápisu ztracena. V některých případech tento způsob může zbytečně zpomalit rychlost zápisu do databáze, protože je nutno čekat na dokončení zápisu na slave server. Zároveň může způsobit nemožnost zápisu do databáze v případě, že se přeruší spojení se slave serverem nastaveným pro synchronní replikaci. Tento způsob je využíván například při bankovních transakcích, kde je potřeba zajistit, aby všechny operace proběhly na obou stranách. V tomto případě je užití tohoto způsobu zcela nezbytné.

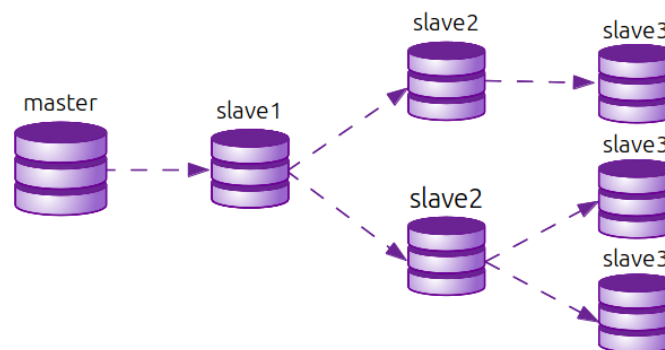
Druhým způsobem je asynchronní replikace, při které se nová data mohou zapisovat na master server, přestože ještě nedošlo k replikaci stávajících dat na slave server [5]. To je sice za běžného provozu rychlejší, v některých případech však může způsobit nekonzistenci dat, například když proběhne transakce na master serveru, který však spadne dřív, než se změna zapíše na slave. V takovém případě se slave změní na master server, ale zároveň se nikdy nedozví o transakci, o které má uživatel informace, že proběhla v pořádku.



Rozdíl mezi synchronní a asynchronní replikací

Dále je možno rozlišovat replikaci podle toho, zda je *logická* nebo *fyzická*. Při fyzické replikaci se kopírují na druhý server bloky binárních datových souborů bez znalosti jejich struktury (sloupce, řádky, ...). Pro tento způsob kopírování dat je potřeba mít na obou serverech stejnou platformu a architekturu. Tento způsob je velice efektivní a často snazší na konfiguraci.

Naopak při logické replikaci se v přenášených datech přenáší samotný SQL příkaz, který se na slave serveru provede stejně jako na master serveru, nebo informace o tom, na kterých řádcích změny proběhly a jaké. Tento způsob je více flexibilní, umožňuje výběr jen několika databází nebo tabulek a není závislý na architektuře ani operačním systému [8].



Ukázka kaskádové replikace

Posledním diskutovaným pojmem je *kaskádová* replikace, která umožňuje připojit další slave k jinému slave serveru místo k hlavnímu master serveru. Kaskádovou replikaci lze využít v případě, že je třeba replikovat data na větší počet slave serverů v clusteru. V případě, že by se všechny slave servery připojovaly k hlavnímu serveru, došlo by u něj k razantnímu snížení jeho výkonu. Kaskádová replikace může být praktická také v okamžiku, kdy se data přenáší na velkou vzdálenost. V případě, kdy je třeba mít několik replik ve velké vzdálenosti od master serveru, je zbytečné, aby se obě kopie přenášely na tak velkou vzdálenost, když druhý slave server lze připojit k prvnímu. Kaskádovou replikaci lze využít také při pádu master serveru, kdy jeden slave povýší na master a druhý je na něj již připojen, aby přijímal repliky.

Každý databázový systém (myšleno SRDB) si volí terminologii a konkrétní nastavení mírně odlišně. Tato kapitola se snaží popsat chápání replikace v co největší míře obecně s ohledem na použití tohoto pojmu v PostgreSQL. Zcela jinou terminologii, i když založenou na stejných principech, zavádí MS SQL Server, který pro export databáze do souboru používá pojem snímková replikace, pro master-slave replikaci pojem transakční replikace a pro multimaster replikaci slučovací replikace.

## 4 Aktuální stav a požadavky

Katedra geoinformatiky (UPOL) aktuálně provozuje servery `virtus.upol.cz`, `atlas.upol.cz` a `geohydro.upol.cz`. Poslední z jmenovaných byl poskytnut jako testovací server pro tuto práci a v budoucnu se s ním počítá jako s master serverem pro zde popisované databázové řešení. První dva zmíněné servery jsou aktivně používány, hostují například geoportál publikovaný skrze ArcGIS Server, který je důležitým prostředkem pro prezentaci projektů a dat, která na katedře vznikají. Data ke geoportálu i dalším aplikacím běžícím na těchto serverech jsou ukládána do MS SQL Serveru, přičemž každý ze serverů obsahuje jiné datové sady, které nejsou pravidelně zálohovány, protože jejich aktualizace není příliš častá. Aktuální řešení nepoužívá replikaci dat, data tedy mohou být nedostupná z důvodu výpadku serveru.

Databáze aktuálně obsahují data například z projektů BotanGIS<sup>4</sup>, Virtuální studovna CHKO Litovelské Pomoraví<sup>5</sup>, dále data metadatového systému Micka<sup>6</sup>, data ze sensorové sítě KGI, data ke studentským pracím a také ukázková data určená pro výuku. Je založeno přibližně 10 účtů, které mají přístup pro zápis, a řádově v desítkách účtů s právem čtení, do databází aktuálně není příliš často zapisováno.

Velké množství dat, které má katedra k dispozici, je však stále uloženo ve formátech Shapefile nebo File Geodatabase. Každý kdo má zájem tato data použít, musí je přenést přes různá hardwarová zařízení nebo je zkopírovat po síti. Studenti si musejí dělat kopie dat při každém cvičení, což velice zdržuje

<sup>4</sup> <http://botangis.upol.cz/botangis/mapa>

<sup>5</sup> <http://virtus.upol.cz/>

<sup>6</sup> [gislib.upol.cz/metadata](http://gislib.upol.cz/metadata)

výuku. Často se totiž jedná o velké objemy dat, jejichž kopie může trvat řádově v jednotkách až desítkách minut. Data jsou poté fyzicky uložena na počítačích v učebnách, což mimo jiné dovoluje, aby se k datům dostal kdokoliv, kdo má na učebnu přístup. Není tedy přehled o tom, kdo data využívá. Studenti navíc netuší, s jakými daty pracují a nabývají nesprávných představ o tom, že všechna data jsou vždy uložena ve formátu Shapefile. Zároveň se špatně zajišťuje aktualizace dat, při které, není-li spravována centralizovaně, může docházet k nekonzistenci dat. Při kopírování dat na různá datová úložiště je navíc těžké dodržet licenční podmínky, se kterými jsou data pořizována.

Základním požadavkem byl výběr takového databázového systému, který je široce používán v oblasti geoinformatiky a zároveň je podporován produkty ArcGIS. Požadavem bylo také zhodnocení finanční stránky, replikace je totiž v mnohých komerčních systémech zařazena až mezi nejpokročilejší funkcionalitu a tedy je dostupná až s dražšími licencemi.

Katedra má v zájmu ukládat do databáze mnohem více datových sad, které má k dispozici a které jsou momentálně dostupné pouze ve formátech Shapefile nebo File Geodatabase. Jedná se například o datové sady ArcČR500 verze 2.0 a 3.0, Data200 (ČUZK), CEDA ČR 150, data, která byla uvolněna jako podpora pro Krajinotvorný program MŽP, nebo data dostupná k produktům ArcGIS a Idrisi. Databázové řešení by tedy mělo být navrženo tak, aby uneslo mnohem větší počet připojení a dotazů než v současné době, protože datové sady, které budou nově dostupné skrze databázi, budou používány v řadě cvičení. Plánem je v rámci cvičení studentům umožnit plnohodnotnou práci s daty, tedy povolit jim jak čtení dat, tak zápis do databáze.

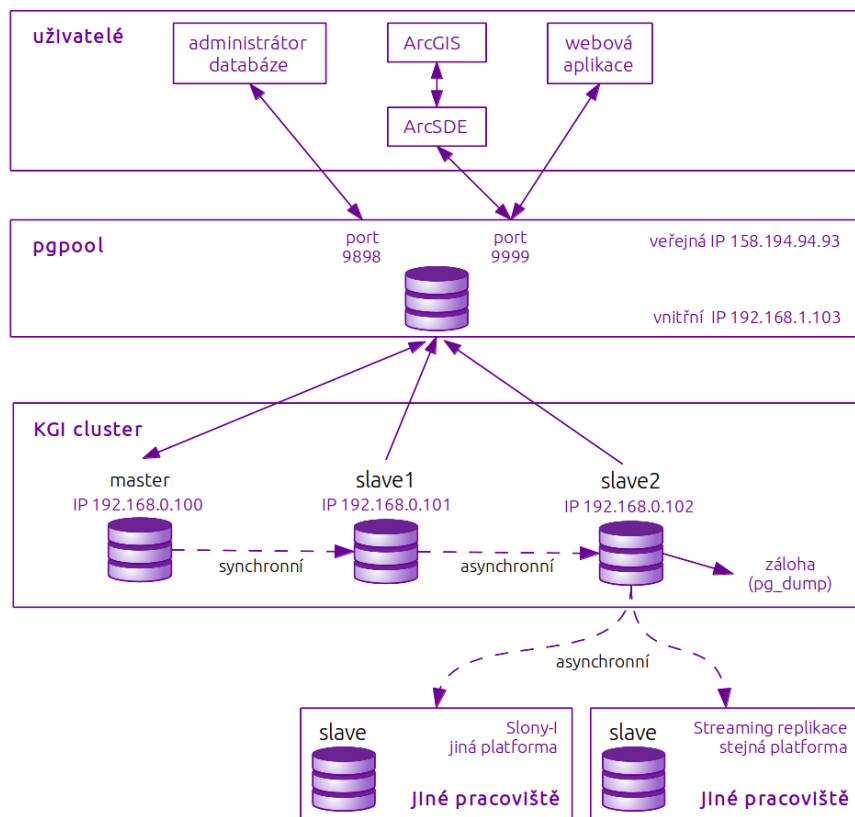
## 5 Návrh replikačního řešení

Po provedení rešerše a zohlednění všech podmínek, požadavků a možností katedry, byl sestaven návrh kompletního databázového řešení založeného na procesu replikace. Z databázových serverů byl vybrán server PostgreSQL hned z několika důvodů. Jedná se o plnohodnotný databázový systém dostupný zdarma se všemi nástroji, je široce používaný v oblasti geoinformačních technologií, je multiplatformní a od verze ArcGIS 9.3 plně podporovaný produkty ArcGIS. Návrh počítá s použitím ArcSDE pro propojení databáze s ArcGIS produkty. Při výběru verzí je nutné zajistit kompatibilitu verzí jednotlivých software, a poté ArcSDE nainstalovat společně s PostgreSQL.

Byl navržen replikační cluster s nejméně třemi servery z důvodů, které již byly diskutovány v kapitole 3.1. Celý cluster poběží na stejné platformě a proto bude možno použít streaming replikaci se všemi výhodami zmíněnými v kapitole 3.1. Byla zvolena jednosměrná master-slave replikace, cluster tedy bude obsahovat jeden master a dva (popř. více) slave serverů. Aby nedošlo ke ztrátě dat v případě, že by master server spadl dřív, než se data zkopírují na slave server, pro první slave (`slave1`) byla zvolena varianta synchronní replikace. Je vhodné, aby servery běžely v lokální síti z důvodu rychlosti a spolehlivosti spojení mezi master a slave serverem.



Druhý server (`slave2`) bude replikovat asynchronně a zároveň, aby nedocházelo k přetížení master serveru, bude replikace probíhat ze `slave1` na `slave2`, tedy kaskádově. Tím bude řešení zároveň připraveno na výpadek master serveru, protože v případě, že master vypadne, `slave1` bude povýšen na master a `slave2` bude ihned moci replikovat. Ze `slave2` lze dále tvořit pravidelnou, například denní nebo týdenní, zálohu pomocí utility `pg_dump`. Záloha pomocí `pg_dump` tak nebude zatěžovat master server a sama o sobě bude probíhat rychleji, než by tomu bylo na master serveru, který je již tak velmi vytížen dalšími procesy.



### Srovnání multimaster a master-slave replikace

Uživatelé se budou připojovat skrze nástroj pgpool, který se bude tvářit jako jediný databázový server, ke kterému se klienti přihlásí bez ohledu na typ jejich dotazu a on sám pak rozhodne, ke kterému ze serverů klienta přihlásí. Tím bude mít zároveň možnost rozložit zátěž na dostupné uzly v clusteru. Pro ještě větší efektivitu provozu databáze bude pgpool uchovávat databázová spojení a při novém dotazu využije stávajícího spojení, místo aby vytvářel spojení nové.

Vzhledem k tomu, že klienti budou k databázovému serveru přistupovat skrze pgpool, není potřeba aby jednotlivé uzly v clusteru měly veřejnou IP adresu. Plně dostačuje, že servery poběží na lokální síti a pouze pgpool bude na serveru s veřejnou IP, čímž se zajistí, že data budou přístupná z internetu.

Návrh počítá také s externími pracovišti, která budou často číst z databáze a budou mít zájem o zrychlení přístupu k datům tím, že se slave server přesune na jejich pracoviště. Typ replikace se zvolí podle jejich operačního systému a jeho architektury. Pokud se bude jednat o shodný systém, jaký bude použit ve výše popsaném clusteru, pak bude možno použít asynchronní streaming replikaci, naopak pokud se bude jednat o systém jiný, bude použita Slony-I replikace.

## Reference

1. OPPEL, A. J. *Databases: A Beginner's Guide*. New York: McGraw-Hill, 2009, 164 s. ISBN 00-716-0846-X.
2. CONNOLLY, T. *Database Systems: A Practical Approach to Design, Implementation, and Management*. Vyd. 4. Harlow: Addison-Wesley, 2005, 1374 s. ISBN 03-212-1025-5.
3. MOMJIAN, B. *PostgreSQL: Introduction and Concepts*. Boston, MA: Addison-Wesley, 2001, xxviii, 461 s. ISBN 02-017-0331-9.
4. MICROSOFT. SQL Server - Replication. *Microsoft* [online], 2013 [cit. 2013-08-27]. Dostupné z: [http://technet.microsoft.com/en-us/library/ms151198\(v=sql.100\).aspx](http://technet.microsoft.com/en-us/library/ms151198(v=sql.100).aspx).
5. OBE, R., HSU, L. *Postgresql: Up and Running*. Sebastopol, CA: O'Reilly, 2012, 164 s. ISBN 978-144-9326-333.
6. BELL, C., KINDAHL, M., THALMANN, L. *MySQL High Availability*. Vyd. 1. Sebastopol, CA: O'Reilly Media, Inc, 2010. ISBN 978-059-6807-306.
7. RIGGS, S., KROSING, H. *PostgreSQL 9 Administration Cookbook: Solve real-world PostgreSQL problems with over 100 simple, yet incredibly effective recipes*. Birmingham: Packt Publishing, 2010, 345 s. ISBN 978-1-849510-28-8.
8. BÖSZÖRMENYI, Z., SCHÖNIG, H.-J. *PostgreSQL Replication: Understand basic replication concepts and efficiently replicate PostgreSQL using high-end techniques to protect your data and run your server without interruptions*. Vyd. 1. Birmingham: Packt Publishing, 2013, vii, 230 s. ISBN 978-1-84951-672-3.