

Univerzita Palackého v Olomouci

Přírodovědecká fakulta

Katedra geoinformatiky

Bc. Libor KIMPL

**STANDARDY PRO DOHLEDOVÉ CENTRUM
SENZOROVÉ SÍTĚ**

Magisterská práce

Vedoucí práce: RNDr. Vilém Pechanec, Ph.D.

Olomouc 2012

Čestné prohlášení

Prohlašuji, že jsem zadanou magisterskou práci vypracoval samostatně pod vedením RNDr. Viléma Pechance, Ph.D. Všechny použité materiály a zdroje jsou citovány s ohledem na vědeckou etiku, autorská práva a zákony na ochranu duševního vlastnictví.

V Olomouci 24. dubna 2012

Děkuji vedoucímu práce RNDr. Vilému Pechancovi, Ph.D. za celkovou pomoc, připomínky a poskytnuté materiály.

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Libor KIMPL**
Osobní číslo: **R100767**
Studijní program: **N1301 Geografie**
Studijní obor: **Geoinformatika**
Název tématu: **Standards pro dohledové centrum sensorové sítě**
Zadávací katedra: **Katedra geoinformatiky**

Z á s a d y p r o v y p r a c o v á n í :

Cílem diplomové práce je rozbor a následná aplikace dostupných standardů pro senzory a sensorové sítě (SOS, SML...), standardy pro přenos a uložení dat (GML/GML aplikační schémata) v prostředí prostorové databáze. V rozboru budou diskutovány závazné a volitelné prvky standardů, doporučené způsoby jejich implementace a procedury pro jejich naplnění. V praktické části, na základě provedeného rozboru, student provede návrh a tvorbu databázové struktury na platformě PostgreSQL/PostGIS pro uložení dat z katedrální sensorové sítě (KSS) zaměřené na S-O monitoring. Současně dojde k vytvoření dílčího prototypu kontrolního panelu, který bude obsahovat prostředí pro zjišťování nastavení a vzdálenou konfiguraci jednotlivých prvků KSS a pro import dat ze senzoru/primárního úložiště. Student vyplní údaje o všech datových sadách, které vytvořil nebo získal v rámci práce, do Metainformačního systému katedry geoinformatiky a současně zálohu údajů ve formě validovaného XML souboru. Celá práce (text, přílohy, výstupy, zdrojová a vytvořená data, XML soubor) se odevzdá v digitální podobě na CD (DVD) a text práce s vybranými přílohami bude odevzdán ve dvou svázaných výtiscích na sekretariát katedry. O diplomové práci student vytvoří webovou stránku v souladu s pravidly dostupnými na stránkách katedry. Práce bude zpracována podle zásad Voženílek (2002) a závazné šablony pro diplomové práce na KGI (2010).

Rozsah grafických prací: **dle potřeby**
Rozsah pracovní zprávy: **max. 50 stran**
Forma zpracování diplomové práce: **tištěná**
Seznam odborné literatury:

DELIN, Kevin, et al. **Environmental Studies with the Sensor Web: Principles and Practice**. Sensors [online]. 2005 by MDPI. Vydáno 28.2.2005. Dostupný z WWW: <<http://www.mdpi.com/1424-8220/5/1/103/>>. ISSN 1424-8220.
Peng, Z-R., Tsou, M-H. (2003): **Internet GIS : distributed geographic information services for the Internet and wireless networks**. Hoboken, Wiley and Sons, 2003 679 s. ISBN 0471359238
KRESSE, Wolfgang; FADAIE, Kian. **ISO Standards for Geographic Information**. Germany : Springer-Verlag Berlin Heidelberg, 2004. 322 s. ISBN 3-540-20130-0.
Pokorný J.: **Databázové architektury: současné trendy a jejich vztah k novým požadavkům praxe**, in Sborník příspěvků 20. ročníku konference Moderní databáze, KOMIX s.r.o., Praha, ISBN: 80-239-7109-3, pp. 5-14, 2006
Voženílek, V. (2002): **Diplomové práce z geoinformatiky**. Vydavatelství Univerzity Palackého, Olomouc, UP, 31 s.
Interní dokumenty projektu AWSN URI OGC Standards and Specifications
www.opengeospatial.org/standards www.postgresql.org/ www.w3.org/XML/

Vedoucí diplomové práce: **RNDr. Vilém Pechanec, Ph.D.**
Katedra geoinformatiky

Datum zadání diplomové práce: **13. června 2010**
Termín odevzdání diplomové práce: **30. dubna 2012**

UNIVERZITA PALACKÉHO OLOMOUČ
PŘÍRODOVĚDECKÁ FAKULTA
KATEDRA GEOINFORMATIKY
tr. Svobody 50, 77146 Olomouc

L.S.

Prof. RNDr. Juraj Ševčík, Ph.D.
děkan

Prof. RNDr. Vít Voženílek, CSc.
vedoucí katedry

V Olomouci dne 13. června 2010

OBSAH

ÚVOD	8
1 CÍL PRÁCE	9
2 STANDARDY PRO SENZORY A SENZOROVÉ SÍTĚ	10
2.1 Sensor Web Enablement (SWE).....	10
2.1.1 Sensor Model Language (SensorML).....	12
2.1.2 Observation and Measurements (O&M).....	14
2.1.3 Transducer Markup Language (TML).....	16
2.1.4 Sensor Observation Service (SOS).....	20
2.1.5 Sensor Planning Service (SPS).....	22
2.1.6 Sensor Alert Service (SAS).....	27
2.1.7 Web Notification Service (WNS).....	30
2.1.8 Interakce mezi SWE službami a SWE kódováními.....	37
2.2 Další technologie, standardy a protokoly pro vytváření senzorových sítí.....	39
2.2.1 6LoWPAN.....	39
2.2.2 IEEE 802.15.4.....	39
2.2.3 ZigBee.....	41
2.2.4 JenNet.....	42
2.2.5 WirelessHART.....	42
2.2.6 ISA100.11a.....	43
2.2.7 Z-Wave.....	43
2.2.8 ANT TM	43
2.2.9 ONE-NET.....	44
2.2.10 DASH7.....	44
2.2.11 Wibree.....	45
2.2.12 Extended Environments Markup Language (EEMML).....	45
3 ZPRACOVÁNÍ PROUDŮ DAT	46
3.1 Systémy řízení proudů dat.....	46
3.2 Speciální architektury.....	48
4 STANDARDY PRO PŘENOS, ULOŽENÍ A FILTROVÁNÍ DAT	49
4.1 Geography Markup Language (GML).....	49
4.1.1 GML schéma – Obecná pravidla a základní komponenty.....	50
4.1.2 GML schéma – Xlinky.....	52
4.1.3 GML schéma – Prvky.....	53
4.1.4 GML schéma – Základní geometrie.....	53
4.1.5 GML schéma – Souřadnicové referenční systémy.....	53
4.1.6 GML schéma – Topologie.....	54
4.1.7 GML schéma – Dočasné informace a dynamické prvky.....	54
4.1.8 GML schéma – Definice a slovníky.....	54
4.1.9 GML schéma – Jednotky, míry a hodnoty.....	54

4.1.10	GML schéma – Směry	54
4.1.11	GML schéma – Pozorování	54
4.1.12	GML schéma – Coverage	56
4.2	Simple Features	56
4.2.1	Well-known text (WKT)	56
4.2.2	Well-known binary (WKB)	58
4.3	Filter Encoding	58
4.3.1	Filtry a typy filtrů	60
4.3.2	Možnosti filtru (Filter Capabilities)	63
4.3.3	Filter Encoding schéma	63
5	NÁSTROJE PRO SPRÁVU DAT ZE SENZOROVÉ SÍTĚ	64
5.1	Testované aplikace	64
5.1.1	Webový prohlížeč měřených dat Fiedler-Mágr	64
5.1.2	MEMSIC Wireless Sensor Network	68
5.1.3	EasyWeather	78
5.1.4	HOBO Event Data Logger	80
5.1.5	MiniCube	81
5.2	Další nástroje	83
5.2.1	SpyGlass	83
5.2.2	TinyViz	83
5.2.3	Surge Network Viewer	83
5.2.4	MonSense	84
5.2.5	NetTopo	84
5.2.6	Octopus	84
5.2.7	TOSGUI	85
5.2.8	MSR Sense	85
5.2.9	MeshNetics WSN Monitor	85
5.2.10	GSN	85
6	DOHLEDOVÉ CENTRUM – SNĚHOMĚRNÝ VAROVNÝ SYSTÉM	86
6.1	Softwarové požadavky aplikace	86
6.2	Návrh databázové struktury	86
6.3	Tvorba a naplnění databázové struktury	87
6.4	Rovnice pro výpočet vodní hodnoty sněhu	91
6.5	PHP skripty a funkčnost aplikace	93
7	DISKUZE	94
8	ZÁVĚR	95
	SUMMARY	
	POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE	
	PŘÍLOHY	

ÚVOD

Standardizace obecně znamená převod obsahu zaznamenávaných údajů do tvaru, který vyhovuje přijatým pravidlům pro daný vědní obor a studované objekty. Patří mezi nejdůležitější kroky tvorby informačního systému ^[11]. Převod popisu přírodních objektů, jevů a procesů a jejich atributů je obtížnější, než u řady jiných vědních oborů. Vyplývá to z mnoha objektivních i subjektivních faktorů. Mezi objektivní faktory patří zejména dlouhodobý a kontinuální charakter přírodních procesů či pouze kvalitativní charakter a nízký stupeň přesnosti apod. Ze subjektivních faktorů se vedle individuálního přístupu a rozdílné erudice pracovníků projevuje i vliv jistého tradicionalismu a terminologické nejednotnosti. Velké problémy nastávají v případě využívání starších podkladů, kde je nezbytné pečlivé posouzení kvalifikovanými a zkušenými odborníky. V zásadě platí, že standardizace musí zajišťovat spolehlivost a co nejširší využitelnost zaznamenávaných údajů. Jde o obecný problém, který se velmi významně uplatňuje v současném světě ve všech oblastech lidských aktivit, zejména v souvislosti s globalizací ^[11].

Senzorové sítě jsou dostupné počítačové sítě pro mnoho prostorově zaměřených zařízení využívajících senzory k monitorování podmínek na různých místech. Podmínek jako jsou například teplota, zvuk, vibrace, tlak, pohyb nebo znečištění ^[28].

Tato magisterská práce poskytuje přehled a rozbor standardů iniciativy Sensor Web Enablement, standardů pro přenos, uložení a filtrování dat a dalších technologií, protokolů a standardů pro vytváření senzorových sítí. Obsahuje jejich obecný popis, popis jejich rozhraní, parametrů a operací. Dále poskytuje přehled existujících nástrojů umožňujících správu senzorových sítí, prohlížení naměřených dat a vizualizace a popis vytvoření vlastní aplikace dohledového centra.

1 CÍL PRÁCE

Cílem diplomové práce je rozbor a následná aplikace dostupných standardů pro senzory a senzorové sítě (SOS, SML...), standardy pro přenos a uložení dat (GML/GML aplikační schémata) v prostředí prostorové databáze. V rozboru budou diskutovány závazné a volitelné prvky standardů, doporučené způsoby jejich implementace a procedury pro jejich naplnění.

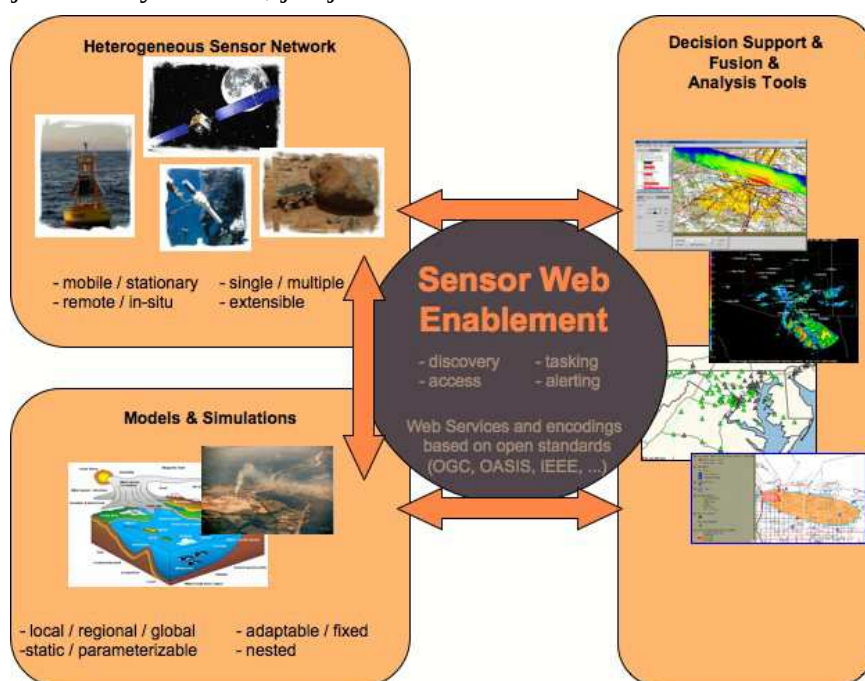
V praktické části, na základě provedeného rozboru, student provede návrh a tvorbu databázové struktury na platformě PostgreSQL/PostGIS pro uložení dat z katedrální senzorové sítě (KSS) zaměřené na S-O monitoring. Současně dojde k vytvoření dílčího prototypu kontrolního panelu, který bude obsahovat prostředí pro zjišťování nastavení a vzdálenou konfiguraci jednotlivých prvků KSS a pro import dat ze senzoru/primárního úložiště.

2 STANDARDY PRO SENZORY A SENZOROVÉ SÍTĚ

2.1 Sensor Web Enablement (SWE)

Sensor Web Enablement (SWE) je iniciativa Open Geospatial Consortium (OGC) definující rozhraní služeb, které umožňují interoperabilní využití sensorových zdrojů tím, že umožní jejich **objevování, přístup k nim, úkolování jich**, stejně jako **oznamování událostí** a **varování**. OGC SWE iniciativa definuje pojem „sensor web“ jako infrastrukturu umožňující přístup k sensorovým sítím a archivovaným datům ze sensorů, která mohou být objevena a lze k nim přistupovat pomocí standardních protokolů a rozhraní programových aplikací (API) ^[40].

Působí jako prostřední vrstva mezi hmotným majetkem a automatizovanými nástroji nebo nástroji ovládanými lidmi, jak je znázorněno na obrázku č. 1 níže ^[27].



Obr. 1: Sensor Web Enablement ^[27]

SWE udává mnoho příležitostí pro propojení sensorů snímajících v reálném čase s internetem. To má mimořádný význam pro zvládání katastrof, monitorování životního prostředí, řízení dopravy, veřejnou bezpečnost, bezpečnost zařízení, nástrojů, průmyslové řízení, vědu, správu zařízení a mnoho dalších oblastí.

Modely, kódování a služby SWE architektury umožňují uskutečnění interoperabilní a škálovatelné, na služby orientované sítě různých sensorových systémů a klientských aplikací. Funkce, na které se OGC zaměřuje v rámci sensor webu ^[28] jsou:

- objevení sensorových systémů, pozorování a procesů pozorování, které splňují aplikační nebo uživatelské požadavky,

- určení sensorových vlastností a kvality měření,
- přístup k parametrům senzoru, které automaticky umožňují softwaru zpracovat a lokalizovat pozorování,
- získávání pozorování v reálném čase nebo pozorování časových řad v standardním kódování,
- dotazování se senzorů na pozorování zájmu,
- předplacení a zveřejnění upozornění, která vydávají senzory nebo jejich služby na základě určitých kritérií.

V rámci iniciativy SWE je dosaženo sensor webu, pomocí kterého lze přistupovat ke všem typům senzorů skrze Web, prostřednictvím vytvoření několika kódování pro popis senzorů a sensorových pozorování a prostřednictvím několika standardů definujících rozhraní pro webové služby. Sensor Web Enablement iniciativa OGC obsahuje následující OpenGIS specifikace:

1. **Sensor Model Language** – standardní modely a XML schéma pro popis sensorových systémů a procesů; poskytuje informaci potřebnou pro objevení senzorů, lokalizaci sensorových pozorování, zpracování pozorování nízkourovňových čidel a výpis požadovaných vlastností.
2. **Observations & Measurements** – standardní modely a XML schéma pro kódování pozorování a měření ze senzorů, archivovaných i vytvořených v reálném čase.
3. **Transducer Markup Language** – konceptuální model a XML schéma pro popis parametrů převodníku a podporu toku dat v reálném čase z a do sensorových systémů.
4. **Sensor Observation Service** – standardní rozhraní webové služby pro dotazování, filtrování a získávání pozorování a informací o systému. Toto je prostředník mezi klientem a úložištěm pozorování nebo sensorovým kanálem v reálném čase.
5. **Sensor Planning Service** – standardní rozhraní webové služby pro podávání požadavků na uživatelem řízené akvizice nebo pozorování. Toto je prostředník mezi klientem a prostředím řízení sbírky senzorů.
6. **Sensor Alert Service** – standardní rozhraní webové služby pro publikování a předplacení upozornění ze senzorů.
7. **Web Notification Services** – standardní rozhraní webové služby pro nesoudobé doručení zpráv nebo upozornění z SAS a SPS webových služeb a dalších prvků služeb pracovních postupů ^[28]. Tato služba je užitečná, pokud je vyžadováno mnoho spolupracujících služeb k zajištění požadavku klienta, nebo když vznikají závažné prodlevy při plnění požadavku klienta.

2.1.1 Sensor Model Language (SensorML)

Sensor Model Language (SensorML) je rámec pro popis vlastností senzoru pomocí Extensible Markup Language (XML). Rozsah SensorML zahrnuje všechny senzory, čidla dálkového průzkumu Země a in-situ senzory, které získávají měření prostřednictvím přímého kontaktu. U tohoto standardu existuje velké překrytí se standardy obraznosti skupiny ISO 19100. SensorML se může stát implementací ISO 19130 ^[1].

Sensor Model Language (SensorML) specifikace poskytuje obecný rámec pro popis nějakého procesu a řetězce procesů. Podstatou SensorML je definovat procesy a jejich komponenty spojené s měřením a transformací pozorování po měření. V rámci SensorML jsou senzory a převodníky modelovány všechny jako proces, který může být propojen a účastnit se procesového řetězce. Účely SensorML ^[24] jsou:

- poskytnout popis senzorů a senzorových systémů pro správu zásob,
- poskytnout informace o senzoru a zpracování ve prospěch zdrojů a pozorování,
- podpora zpracování a analýzy senzorových pozorování,
- podpora geolokace sledovaných hodnot (měřená data),
- zajistit funkční charakteristiky (např. přesnost, práh apod.),
- zajistit explicitní popis procesu, kterým se zabývalo pozorování (tj. počet řádků),
- poskytnout spustitelný řetězec procesů pro odvozování nových datových produktů na vyžádání (tj. odvoditelné pozorování),
- archivovat základní vlastnosti a předpoklady týkající se senzorových systémů.

SensorML poskytuje XML schéma pro definování geometrických, dynamických a pozorovacích vlastností senzoru. Kořenem pro všechny SensorML dokumenty je senzor a popis senzoru rozdělený na hlavní části informace ^[1], včetně:

- identifikace senzoru,
- omezení dokumentu, jako je stupeň utajení nebo omezení použití,
- platformy, ke které je senzor připojen,
- umístění senzoru ve srovnání s referenčním souřadnicovým systémem,
- pozorování, která lze měřit,
- vlastnosti odpovědi senzoru, jako je citlivost,
- geometrické a časové charakteristiky vzorků,
- provozovatele a Sensor Planning Service,
- popisu senzoru, jako sériové číslo,
- popisu dokumentace, jako autor a úpravy.

Základním konceptem SensorML je tzv. *AbstractProcess*. Z tohoto typu jsou odvozeny všechny ostatní typy procesů. Odvozené typy procesů lze rozdělit do dvou skupin: **fyzické procesy** (např. čidla) a **nefyzické procesy** (např. kroky zpracování, které

jsou provedeny z naměřených hodnot). *AbstractProcess* obsahuje vlastnosti, které jsou relevantní pro všechny odvozené typy procesů. *AbstractProcess* obsahuje:

- *název*
- *popis*
- *vstupy*: pro fyzické snímací zařízení jsou vstupy pozorované jevy, pro nefyzické procesy (např. výrobní kroky) jsou vstupy jakákoliv přichodící data
- *výstupy*: data, která jsou vytvořena podle procesu (např. naměřená hodnota nebo vypočítaný výsledek)
- *parametry*
- *metadata*

Metadata pro proces lze rozdělit do pěti různých podskupin ^[36]:

- *všeobecné informace*: identifikátory procesu a seznam podle klasifikátorů, které usnadňují objevení senzoru (např. typ senzoru zamýšlené aplikace)
- *omezení*: informace o bezpečnosti a právních omezeních, jakož i informací o době, po kterou je popis procesu platný
- *vlastnosti*: jakékoli vlastnosti nebo schopnosti senzoru, které mohou pomoci k jeho zjištění, a to včetně možností jako je frekvence měření nebo přesnost, nebo i vlastností jako rozměry snímače
- *reference*: informace, kde se nachází další dokumentace o procesu a kontaktní údaje (např. provozovatel senzoru)
- *historie*: přehled akcí, které popisují historii senzoru, tento seznam může obsahovat informace jako jsou údržba, změny parametrů atd.

Nefyzické procesy

Nefyzické procesy mohou být chápány jako matematické operace. Základním prvkem nefyzického procesu je tzv. *ProcessModel*. Kromě *AbstractProcess*, *ProcessModel* obsahuje vlastnost metody, která popisuje matematickou operaci procesu.

Složitější procesy mohou být popsány pomocí *ProcessChain* typu. Tento typ umožňuje kombinaci *ProcessModels* a dalších *ProcessChains*, aby popsal sled dílčích procesů. Za účelem dosažení tohoto cíle, *ProcessChain* obsahuje seznam *ProcessModels* a *ProcessChains* obsahuje i popis, jak jsou propojeny jejich výstupy (zdroje) a vstupy (cíle) ^[36].

Fyzické procesy

Fyzické procesy jsou procesy, které se vztahují k prostoru a času. Typickými příklady takových procesů jsou detektory, spouštěče, sensorové systémy a vzorkovníky. Na rozdíl od nefyzických procesů, fyzické procesy mají prostředky k vyjádření jejich časových a prostorových vztahů (např. podle referenčních rámců a podle pozice v těchto referenčních rámcích). Navíc rozhraní fyzických procesů lze označit (hardware i software rozhraní) na základě OSI Reference Model.

Základní typ fyzického procesu je *Component*. Jedná se o nejmenší jednotku, která nemůže být dále rozdělena. Složitější fyzické procesy mohou být popsány pomocí typu *System*, který shromažďuje jednotlivé komponenty, stejně jako jiné systémy. Příkladem tohoto vztahu je meteorologická stanice (systém), která obsahuje několik snímacích zařízení, jako jsou teploměry, čidla rychlosti větru atd. (komponenty) ^[36].

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/sensorml>

2.1.2 Observation and Measurements (O&M)

Observation and Measurements (O&M) specifikace poskytuje standardní model pro reprezentaci a výměnu výsledků pozorování. O&M je především konceptuální model popisující vztah mezi různými aspekty procesu sběru dat. Podle O&M, pozorování (observation) je událost, která se vyskytuje v určitém časovém okamžiku a vytváří hodnotu pro pozorovaný jev. Kromě času a hodnoty měření je O&M schopna popsat další vlastnosti měření, např. proces používaný ke generování naměřené hodnoty, stejně jako umístění a kvality měření. O&M pokládá naměřenou hodnotu za přibližnou hodnotu atributu pozorovaného prvku zájmu (FOI). Další informace jsou považovány za metadata pro další analýzu a interpretaci dat. Tudíž pozorování slouží jako poskytovatel majetkové hodnoty pro prvek zájmu. O&M schémata nemají umožnit pouze definici pozorování, ale i jevů. Na základě těchto definic mohou být navrženy slovníky, aby definovaly jevy používané v určité aplikační doméně. Takový slovník je základem pro obecnou představu o senzorických datech ^[19].

Observations and Measurements standard definuje konceptuální schéma XML kódování pro pozorování a pro prvky, které jsou zapojené do vzorkování během samotného pozorování. Je rozdělen právě do dvou částí, kde v jedné části je popsáno klasické pozorování a v části druhé je popsáno vzorkování prvků. Model není striktně vázán na prostorové informace, ale může být použit i pro získávání jiných druhů dat, které vznikají určitým pozorováním. O&M je jednou z klíčových norem v souboru OGC Sensor Web Enablement.

Funkce pozorování je vázána na výsledek hodnoty jevu, na kterém bylo pozorování provedeno. Vlastnost pozorování je shodná s vlastností pozorovaného jevu. Pozorování používá ke stanovení hodnoty výsledku určitý postup, který může být realizován: pozorovatelem, senzorem, analytickým odvozením, simulací nebo číselným odvozením. Hodnoty pozorování mohou mít mnoho datových typů, od primitivních datových typů (kategorie, měření) až po komplexní datové typy (čas, umístění a geometrie). Výsledek pozorování je spojen s konkrétním časem nebo časovým obdobím, ve kterém je pozorování prováděno a výsledná hodnota je konkrétní vlastností přiřazenou k prvkům pozorování.

Model pro Observations and Measurements

Klíčovou myšlenkou modelu je to, že výsledkem pozorování je odhad hodnoty vlastnosti prvku pozorování, a že metadata pozorování slouží k podpoře hodnocení, interpretaci a využití výsledků ^[41]. Model pozorování využívá tzv. pozorovatelsky centrické hledisko (prvek, který je pozorován se nachází v centru pozorování), to je v kontrastu s praxí, kterou používají tzv. senzorově orientované modely. Ty používají tzv. poskytovatelsky centrické hledisko (senzor, kterým dochází ke sběru dat je v centru a sbírá data kolem sebe). Pozorování slouží jako tzv. poskytovatel jednotlivých hodnot pro pozorované prvky. Typ pozorovaných prvků (jaké typy prvků mohou být pozorovány) je definován v normě ISO 19109. Typ prvků pozorování určuje sadu vlastností, které dané prvky mohou mít (název, vlastnictví apod.). Je nezbytné definovat před začátkem samotného pozorování, jaký typ prvků budeme pozorovat, aby nedošlo k případu, že pozorovaný prvek nebude nabývat určitých vlastností pozorování ^[41].

Pozorování se skládá z následujících klíčových vlastností:

- *featureOfInterest* – je doména určující prvek, který je reprezentací pozorovaného cíle, jedná se o skutečný objekt reálného světa, který je pozorován.
- *observedProperty* – popisuje jev, pro který výsledek pozorování poskytuje odhad jeho hodnoty. Musí to být vlastnost, která souvisí s typem prvku zájmu. Stručně řečeno se tomu říká koncept poskytovatele majetkové hodnoty.
- *procedure* – je popis procesu použitého ke generování výsledku.
- *result* – obsahuje hodnotu generovanou procedurou. Obecně se jedná o senzor, ale může to být lidský pozorovatel nebo i simulační model.

Dalšími vlastnostmi jsou:

- *samplingTime* – čas, po který platí výsledek prvku zájmu.
- *resultTime* – doba, po kterou byl aplikován postup spojený s pozorováním. Pokud se *samplingTime* a *resultTime* shodují, *resultTime* může být vynechán.
- *parametr* – je parametr specifický pro událost. Obvykle se používá pro záznam parametrů týkajících se životního prostředí nebo parametrů vzorkování specifické události. Např. vzorkování výšky měření teploty vzduchu.
- *metadata* – umožňuje připojit jakýkoliv druh metadat k pozorování.
- *resultQuality* – umožňuje kódování kvality informace.

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/om>

2.1.3 Transducer Markup Language (TML)

Transducer Markup Language (TML) standard je metoda a formát zprávy pro popis informace o převodnících a jejich systémech a pro zachycování, výměnu a archivování současných, minulých a budoucích dat. TML poskytuje mechanismus pro účinné a efektivní zachycování, přenos a archivování dat převodníku v jednotné podobě bez ohledu na zdroj. TML využívá konceptuální model a XML schéma pro popis převodníků a podporu toku dat v reálném čase z a do sensorových systémů ^[28]. Transducer Markup Language definuje:

- soubor modelů popisujících charakteristiky chování převodníku,
- efektivní metodu, jak transportovat data z převodníků a připravit je na fúze prostřednictvím prostorových a časových sdružení.

Senzorová data jsou často pouze výtvozem sensorového vnitřního zpracování než skutečným záznamem změny jevu. Účinky tohoto zpracování na cílené jevy lze charakterizovat jako funkce. TML odpovědní modely jsou normalizované XML popisy známého hardwarového chování. Tyto modely mohou být použity k zvrácení účinku narušení hodnoty a navrácení se k původní hodnotě změny jevu. TML nabízí modely pro latenci snímače a integrační čas, šumové číslo, prostorovou a časovou geometrii, frekvenční odezvu, odezvu v ustáleném stavu a impulsní odezvu. Tradiční XML zapouzdří každý datový prvek v sémanticky smysluplný tag. XML je proto pro výměnu dat velmi vhodné.

Transducer Markup Language TML je jazyk pro zachycení a charakterizování nejen samotných dat z převodníků, ale také pro informace nezbytné pro zpracování a pochopení údajů pro případné příjemce dat. Senzory a vysílače mohou být zachyceny a charakterizovány v rámci TML, což vede k použití termínu „převodník“ spíše než k „senzor“. TML zpracovává nejen statická data, ale i přímo přenášená data z převodníků ^[29].

Popisy důležité pro pozdější zpracování a pochopení dat zahrnují kalibraci, provozní podmínky, nastavení zařízení, vlastnosti a charakteristiky převodníků, vztahy převodníků ve vzájemném systému a chování systému a jsou obsaženy v popisu TML. Logické modely, modely chování, přenosové funkce a jiné informace důležité pro zpracování údajů jsou všechny obsaženy v rámci TML.

TML je schopen tagovat data v přesném čase tak, aby bylo možné přesně vědět, kdy byl fyzikální jev naměřen na jednotlivých hladinách měření, a zachycuje také latenci nebo zpoždění informací na slušném rozlišení. To umožňuje přesné stanovení doby, kdy byl vzat datový bod, stejně jako pomoc při interpolaci mezi datovými body a rekonstrukcí událostí ^[29].

Interoperabilita a integrace dat převodníku

Hlavním účelem TML je umožnit rozšířené sdílení a zároveň poskytovat větší porozumění dat, např. že můžeme získat lepší představu o našem světě prostřednictvím převodníku. Chceme-li spravovat komplexní úkol zpracovávající a sdružující milióny senzorických vstupů, musíme využít počítače ke koordinaci a sdružení dat a reprezentaci pouze zajímavých dat shrnutým způsobem tak, aby lidé mohli provádět důležitá rozhodnutí na vyšší úrovni a včas. Pro usnadnění tohoto úkolu musíme vytvořit kvalitní a jednoznačný komunikační jazyk pro převodníky a počítačové stroje tak, aby se chovaly jako homogenní systém. K dosažení tohoto musí být komunikační jazyk úplný, konzistentní a co možná nejúčinnější. Častokrát se tyto vlastnosti vzájemně vylučují ^[29].

Interoperabilita a obecné chápání dat převodníku

Pro interoperabilitu potřebujeme:

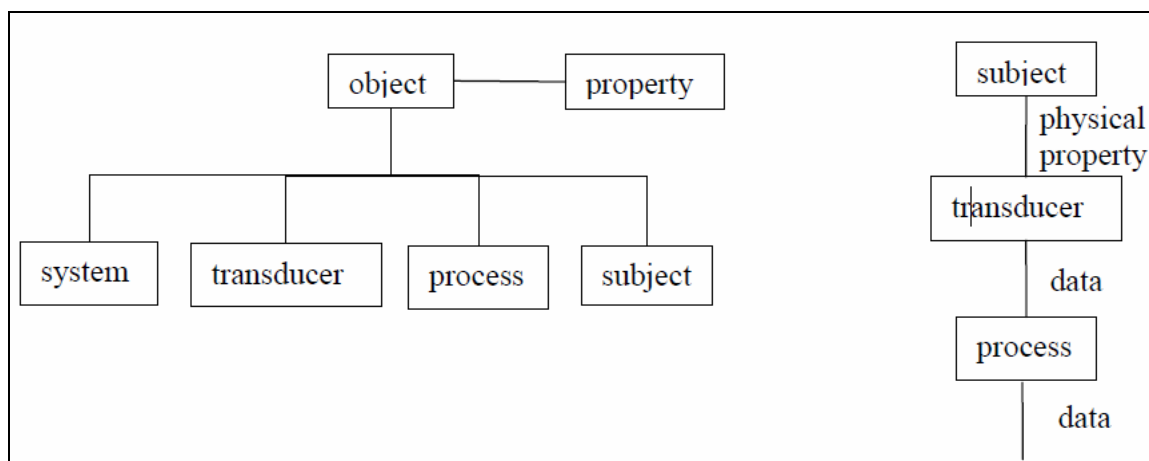
- data, která popisují kódování, strukturu a organizaci přenášených dat,
- data, která popisují strukturu a organizaci dat převodníku.

Pro pochopení dat převodníku potřebujeme:

- data, která popisují objekt měření nebo akce převodníku,
- data, která popisují odpověď převodníku a geometrické charakteristiky,
- data, která popisují veškerá předzpracování údajů,
- data, která popisují vnější vztahy pro převodníky a procesy v rámci systému.

TML poskytuje ucelený soubor obecných normalizovaných popisů k obecnému chápání dat převodníku. Toto obecné porozumění umožní strojům začít vytvářet sdružení dat převodníků na nižší úrovni (jako např. spojování nebo sdružení a registrace prvků). Je důležité, aby datový popis nadále úzce souvisel i s daty. Jedno nemůže být chápáno bez druhého ^[29].

TML umožňuje popis k poskytnutí kontextu dat převodníku. Data z převodníku se týkají fyzické vlastnosti (nebo jevu), která se týká fyzického předmětu. Předmětem vlastnosti je, že rozhraní převodníku mohou, ale nemusí být známy v době psaní TML datových popisů. Pokud nejsou, pak mohou být popsány později. Data do a z převodníku mohou být předem zpracována. Nakládání s údaji, které budou zpracovávány, nebo které byly zpracovány, vyžaduje kompletní popis zpracování. Tyto vztahy jsou znázorněny na obrázku č. 2. Převodníkový systém se může skládat z jednoho nebo více objektů. Objekt sám může být převodníkovým systémem nebo převodníkem, procesem nebo předmětem. Tento vztah je také znázorněn na obrázku č. 2.



Obr. 2: TML konceptuální komponenty ^[29]

Tok dat v reálném a minulém čase

Vzhledem k tomu, že převodníky jsou rozhraními mezi reálným světem a námi modelovanou situací, jsou jejich odpovědi reprezentací neustále se měnícího světa. Tento měnící se svět je reprezentován aktuálním tokem dat převodníku. TML zachytí tento tok dat, který představuje skutečné události zachycené převodníky a udržuje relativní a absolutní časové a prostorové vztahy těchto údajů tak, aby bylo možné prověřit události přesně tak, jak se staly buď v reálném čase nebo na jiném místě a v jiný čas. TML data představují nepřetržitý tok údajů z jednoho převodníku, případně různého množství převodníků, vše prokládané náhodně ve zhruba chronologickém pořadí. Záleží na zařízení pro sběr dat, jak jsou data přesná a správná, a že následkem schopností převodníku nejsou data znehodnocena. Úkol datové integrace a porozumění je ponechán na zpracovatelích převodníku ^[29].

Statická a dynamická data

TML používá programovatelné datové struktury, které se přizpůsobí jednotlivým převodníkům. Dále také používá obecný model nebo popis různé složitosti dat. Hlavním cílem TML je výměna dat převodníku co nejpresněji a nejefektivněji. Toho je dosaženo výměnou dvou typů dat:

- Dynamických dat, která představují časově proměnný svět.
- Popisu dat tak, aby TML zpracovatel byl schopen pochopit různé složitosti údajů pro konkrétní systém převodníků.

Popis dat je nutný k pochopení dat a údajů potřebných k pochopení datových popisů. Vzhledem k velmi časté možnosti, že se data datového popisu mění, spoléhá se na tok dat

pro převedení časově citlivých parametrů senzorových dat. Je nezbytné, aby bylo vedeno logické spojení mezi těmito dvěma typy dat.

Struktura TML

TML nabízí samostatný lehký XML obal pro efektivní přepravu dat převodníku, jakož i všechny potřebné informace pro dekódování, zpracování, analýzu a pochopení dat převodníku. Převodníkový systém, prostorová a časová metadata jsou také charakterizováni a předáváni spolu s daty, což umožňuje analýzu dat převodníku a zpracování dat využívajících pouze informace provedené v TML datovém toku ^[29]. TML dosahuje interoperability prostřednictvím několika klíčových vlastností. Mezi ně patří:

- schopnost připojit TML modely chování a reakce na standardní definici převodníku,
- schopnost propojení s a využití libovolného výstupu senzoru,
- normalizace nesouhlasných datových souborů pomocí TML modelů chování.

TML se skládá z celého popisu systému a dat převodníku z nebo do tohoto systému. Obrázek č. 3 znázorňuje celkovou strukturu TML, včetně popisu systému a dat. Popis systému obsahuje souhrnný popis převodníku, procesu, vztahu a kódování seskupení (cluster encoding). Toto není nutně kompletní TML model, ale je určen pouze pro ilustraci celkové struktury TML popisu ^[29].

```
<?xml version="1.0" encoding="UTF-8"?>
<tml>
  <system name="SYS01">
    <sysClk>
      <period>
        <values>0.001</values>
      </period>
    </sysClk>
    <transducers>
      <transducer name="SYS01_CAMERA">
        <logicalDataModel>
          <dataSet id="CAMERA_IMAGE"></dataSet>
        </logicalDataModel>
      </transducer>
    </transducers>
    <process name="PROCESS_1" uid="SYS01_PCS">
      <input>
        <procDataSet name="CAMERA_IMAGE"></dataSet>
      </input>
    </process>
    <relations>
      <posRelation UidRef="SYS01_CAMERA" refSystem="SYS01_GPS"/>
      <spatialCoord coordName="latitude">
        <values bindUid="SYS01_GPS_LAT"/>
      </spatialCoord>
    </posRelation>
    </relations>
    <clusterDesc name="SYS01_GPS_LLA">
      <dataUnitEncoding>
        </ dataUnitEncoding >
      </clusterDesc>
    </system>
  <!--start data stream-->
  <data clk="263084829" ref="SYS01_TIMESTAMP">2006-03-02T14:39:41.04Z</data>
  <data clk="263085859" ref="SYS01_GPS_LLA">0.577041 -2.035342 0.000000</data>
  <data clk="12344" ref="SYS01_CAMERA_RGB24_BASE64">ABabdkS2836875..... ARBKDK==</data>
  <!--continue data stream-->
</tml>
```

Obr. 3: TML statická data a datový tok ^[29]

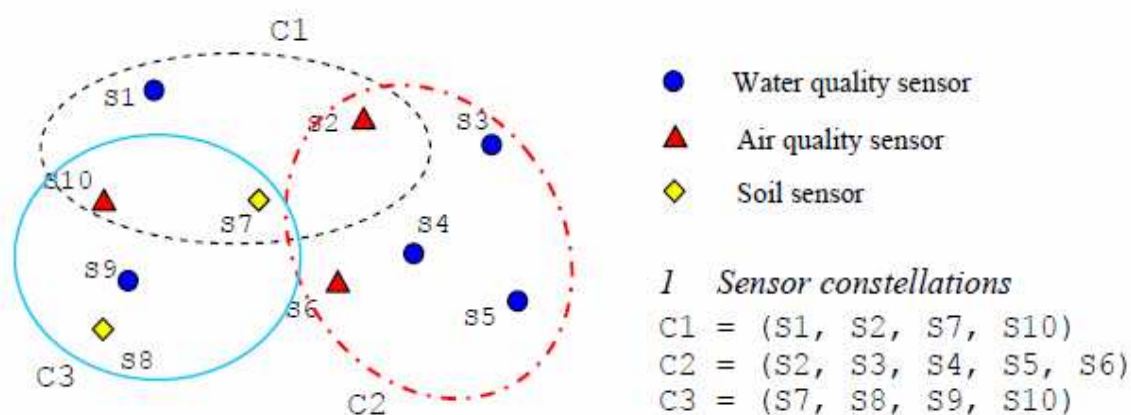
Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/tml>

Pozn.: V rámci iniciativy Sensor Web Enablement jsou standardy SensorML, O&M a TML tři standardy kódování pro popis senzorů a sensorových pozorování.

2.1.4 Sensor Observation Service (SOS)

Práce na předchůdci SOS, na Sensor Collection Service (SCS), začala během testování OGC Web Services 1.1 (OWS 1.1) a zaměřila se především na požadavky interoperabilního rozhraní pro statické senzory a sensorové sítě. Během iniciativy OWS 1.2, úsilí SCS zaměřeno na požadavky pro dynamické vzdálené senzory (a sensorovou síť) a definovalo více explicitní rozhraní využívající SensorML. SOS vychází z předchozích úsilí SCS a také obsahuje rozhraní zaměstnávající TML ^[25].

Sensor Observation Service poskytuje rozhraní API pro správu nasazených senzorů a získávaných dat ze senzorů a zejména „observation“ dat. Ať už z in-situ senzorů (např. monitorování vody) nebo z dynamických senzorů (např. satelitní snímkování). Měření ze systémů senzorů přispívají k tomu, že geoprostorová data jsou dnes nejvíce používána v geodetických systémech. Obecné schéma in-situ senzoru je ilustrováno na obrázku č. 4, kde rozmístěné senzory (S_n) různých typů jsou seskupeny do několika soustav (C_n), které jsou pak přístupné přes nějakou službu (např. SOS) ^[25].



Obr. 4: Obecné schéma pro In-situ senzory ^[25]

Cílem SOS je poskytnout přístup k pozorováním ze senzorů a sensorových systémů standardním způsobem, který je konzistentní pro všechny sensorové systémy včetně dálkového, in-situ, pevného a mobilního senzoru. Jedná se o náročný úkol, protože uživatelé dat ze senzorů byli historicky rozděleni na ty, kteří se zabývají především in-situ senzory a ty, kteří se zabývají dálkovými senzory. Terminologie, perspektiva a očekávání těchto dvou širokých skupin jsou odlišné. SOS využívá specifikace Observation and Measurements (O&M) pro modelování pozorování senzoru a TransducerML a SensorML specifikace pro modelování senzorů a sensorových systémů ^[25].

Přístup SOS definuje společný model pro všechny senzory, sensorové systémy a jejich pozorování. Tento model je „horizontální“, protože platí pro všechny domény, které používají senzory pro sběr dat. Specifické detaily domény jsou zapouzdřeny ve druhé vrstvě (features-of-interest, observed properties, sensor descriptions), která umožňuje základní „pozorování“, která mají být zpracována ^[25].

Stejně jako ostatní OGC webové služby, je SOS založen na výměně dotazů a odpovědí mezi uživateli (údaje o spotřebiteli nebo výrobcí) a službách ve známé podobě, jak jsou definovány v specifikacích standardu. Dotazy jsou vždy označeny jako: service (služba), version (verze) a request (parametry požadavku) (SOS bude mít v tomto pořadí hodnoty, "SOS", "1.0.0" a název požadavku) a řada dalších parametrů, které jsou pro typ požadavku specifické. XML (Extensible Markup Language) a KVP (Key-Value pairs) dotazy jsou obecně podporovány, vždy za předpokladu odpovědi ve formátu XML. SOS specifikace předpokládá povinné a volitelné dotazy, které jsou rozděleny do tří profilů: **1. Core**, povinné dotazy umožňující zkoumání služeb, popis senzorů a sběr dat. **2. Transactional**, volitelné dotazy podporující získávání nových dat (nové senzory a nová pozorování). **3. Enhanced**, volitelné různé dotazy na získávání dalších informací nebo usnadnění přístupu k datům ^[4].

SOS má tři základní povinné „Core“ operace: *GetObservation*, *DescribeSensor*, a *GetCapabilities*. Bylo také definováno několik volitelných, nepovinných operací. Jsou to dvě operace na podporu transakcí, *RegisterSensor* a *InsertObservation*, a šest rozšířených operací zahrnující *GetResult*, *GetFeatureOfInterest*, *GetFeatureOfInterestTime*, *DescribeFeatureOfInterest*, *DescribeObservationType*, *DescribeResultModel* a *GetObservationById* ^[25].

GetObservation

Operace *GetObservation* poskytuje přístup k datům z pozorování a měření senzoru prostřednictvím prostorově-časového dotazu, kterým lze filtrovat jevy ^[25]. Umožňuje získat požadované pozorování, podle souboru podporovaných filtrování a nastavení parametrů, ve formátu Observation and Measurements (O&M) nebo případně v TML ^[4].

DescribeSensor

Operace *DescribeSensor* načítá podrobné informace o senzorech a procesech vytvářejících tato měření ^[25]. *DescribeSensor* dotaz poskytuje informace o senzoru specifikované podporovaným povinným parametrem *procedure*, jeho schopnosti nástrojů, vlastnosti a další podpůrná data ve formátu Sensor Model Language (SensorML) nebo Transducer Markup Language (TML) ^[4].

GetCapabilities

V profilu Core dotaz *GetCapabilities* umožňuje prozkoumat službu vyhledáváním informací: poskytovatel služby, podporované dotazy a příslušné parametry, nabízené postupy, vlastnosti pozorování a prvky zájmu ^[4]. Operace *GetCapabilities* poskytuje

prostředky pro přístup ke službě SOS metadata. Tato operace umožňuje klientům získat službu metadata o konkrétní instanci služby. Nezahrnuje žádný parametr dotazu, protože název prvku udává konkrétní operaci. Odpověď na GetCapabilities operaci je zakódovaný XML dokument ^[25].

Volitelný Transactional profil je velmi důležitý, kdykoli když je služba orientovaná nejen na spotřebitele, ale také na producenta: v tomto případě profil poskytuje uživateli přístupový bod pro nahrávání dat ^[4].

RegistrSensor

RegisterSensor dotaz umožňuje přidat do SOS nový senzor (SensorML nebo TML, jako odpověď DescribeSensor dotazu) a šablonu pozorování (O&M), která dá pokyn službě k novým pozorováním, jenž mají být shromážděny. V odpovědi je vrácen jedinečný identifikátor senzoru ^[4].

InsertObservation

InsertObservation usnadní senzoru registraci nových pozorování poskytováním předem získaného identifikátoru snímače a O&M <observation> prvku obsahujícího naměřené hodnoty ^[4].

Enhanced profily přidávají další funkce službám, jako je možnost opakovaně získat pozorování s předem definovanou GetObservation šablonou dotazu, umožňující uložit šířku pásma (GetResult), nebo možnost získat geometrii inzerovaného prvku zájmu (GetFeatureOfInterest) nebo čas, po který byl prvek zájmu pozorován (GetFeatureOfInterestTime) ^[4]. Dále také XML schéma pro určitý GML typ prvku, který je inzerován v dokumentu možností (DescribeFeatureType), přístup ke XML schématu, které popisuje typ pozorování, jenž je vráceno zvláštním jevem (DescribeObservationType). To je potřeba pro specializované typy pozorování. Nebo také přístup ke XML schématu výsledného prvku, který bude navrácen, pokud klient požádá o určitý výsledný model (DescribeResultModel) a přístup k pozorováním na základě identifikátoru obdrženého klientem před voláním operace (GetObservationById) ^[38].

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/sos>

2.1.5 Sensor Planning Service (SPS)

Sensor Planning Service (SPS) standard je prostředkem pro sběr dat (prostředkem mohou být **senzory** nebo **simulace**). Stanoví se **oblast zájmu**, pro kterou chceme získat data a jaká **data** chceme získat (nejčastěji např. teplotu, srážky nebo snímky oblasti). Podporována jsou data nejen odlišného druhu s odlišnými vlastnostmi, ale také různé druhy zpracovávajících systémů, které mohou nebo nemusí poskytovat přístup ke sběru

dat, zpracování jednotlivých úloh, archivaci, distribuci, výsledným datům a informacím. SPS je navržený tak, aby byl dostatečně flexibilní a aby zvládnul takto široký okruh konfigurací.

OpenGIS SPS specifikace byla navržena k interoperabilní službě (interoperabilita = schopnost různých systémů vzájemně spolupracovat a poskytovat si služby), kterou uživatel zjišťuje proveditelnost požadavků pro jeden nebo více senzorů/platformem nebo přímo odesílá požadavky k těmto senzorům/platformám ^[42].

Některé senzory nebo sensorové platformy podporují dynamickou konfiguraci za běhu. To může být například konfigurace míry (frekvence) vzorkování nebo řízení pohyblivé sensorové platformy. Zkoušení senzorů interoperabilním způsobem lze provést pomocí Sensor Planning Service (SPS). SPS je rozhraní webové služby, která umožňuje klientům předložit senzorům úkoly ^[3]. Sensor Planning Service (SPS) nabízí standardizované rozhraní pro ovládání senzorů a simulací. SPS zajišťuje operace pro řízení úkolů, ^[39] včetně:

- definování úkolů,
- kontroly proveditelnosti úkolů,
- kontroly stavu běžících úloh,
- aktualizace a úpravy běžících úloh,
- zrušení úkolů.

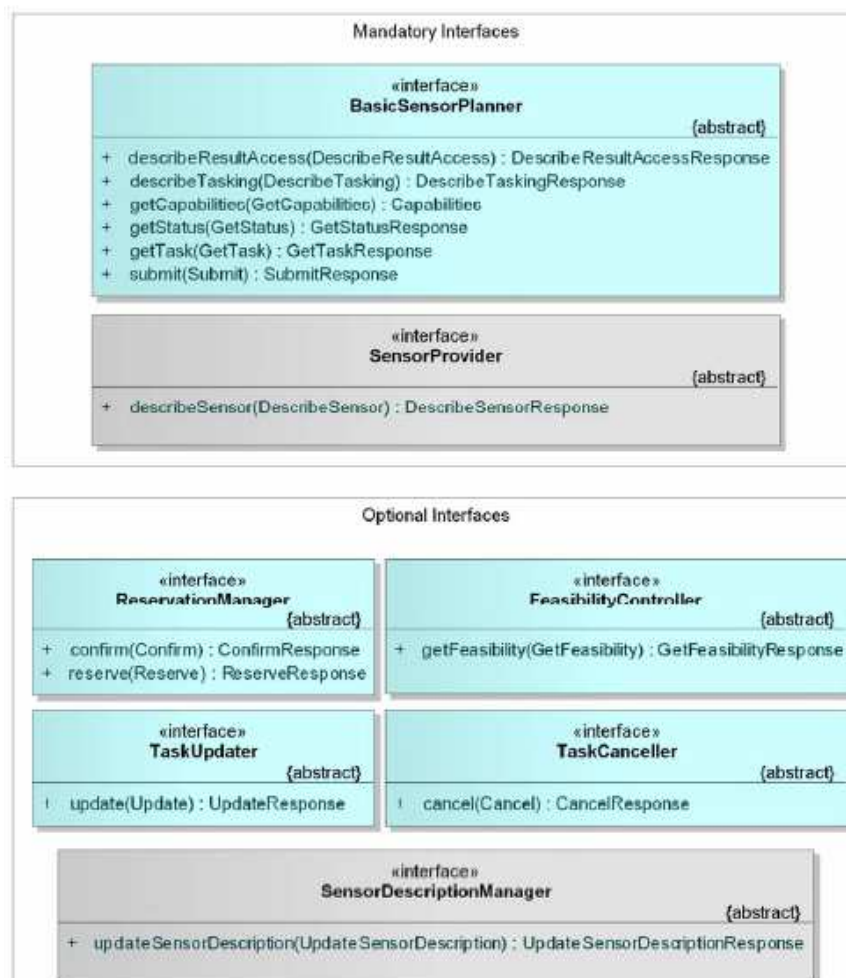
Rozhraní SPS slučuje operace zahrnující celý proces řízení a plánování úkolů senzoru. Obsahuje kontrolu, zda je úkol senzoru proveditelný pomocí *GetFeasibility* operace, podání úkolů (*Submit*), stejně jako sledování stavu předložených úloh (*GetStatus*). Aby bylo možné vybavit klienty dostatečnou informací pro formulaci žádosti o vyhodnocení, může (vlastním popisem) syntaxí pro popis úkolu požádat (*DescribeTasking*). SPS dopředu předložené úkoly zasílá senzoru. Jakkoliv následně získaná data se neshromažďují pomocí SPS. Místo toho, SPS poskytuje prostředky pro dotazování, kde jsou naměřená data dostupná (*DescribeResultAccess*) ^[3].

SPS operace lze rozdělit na informační a funkční. **Informační** jsou *GetCapabilities*, *DescribeTasking*, *DescribeResultAccess* a *GetStatus*. *GetCapabilities*, *DescribeResultAccess* a *GetStatus* dohromady poskytují informace, které uživatel SPS potřebuje vědět, zatímco *DescribeTasking* poskytuje popis informací, které potřebuje vědět snímací systém. **Funkční** operace jsou *GetFeasibility*, *Submit*, *Update* a *Cancel*. SPS operace jsou požadovány uživatelem a prováděny serverem. Kódování používá HTTP GET s KVP kódováním a HTTP POST s XML nebo KVP kódováním. Následující tabulka (Tab. 1) shrnuje operace SPS služby a jejich metody kódování definované v této specifikaci. Tyto operace mají mnoho podobného jako jiné operace OGC webových služeb, včetně WMS, WFS a WCS.

Tab. 1: Kódování SPS operací ^[42]

Operation name	Request encoding
GetCapabilities	KVP and optional XML
DescribeTasking	XML
GetFeasibility	XML
Submit	XML
GetStatus	XML
UpdateRequest	XML
Cancel	XML
DescribeResultAccess	XML

SPS definuje pět rozhraní s jedenácti operacemi, které mohou být požadovány klientem a prováděny serverem SPS. Kromě toho obsahuje dvě rozhraní od SWE Common Service Model [OGC 09-001] - tato rozhraní definují dvě operace. Obrázek č. 5 je diagram UML zobrazující tyto rozhraní (šedé rozhraní jsou definovány v OGC 09-001).



Obr. 5: UML diagram SPS rozhraní ^[26]

SPS rozhraní jsou:

a) *BasicSensorPlanner* (povinné) – Toto rozhraní představuje základní funkce SPS. Obsahuje následující operace:

GetCapabilities

Povinná operace GetCapabilities umožňuje klientům identifikaci dostupných senzorů a podporovaných operací. Struktura a syntaxe vyplývá ze specifikace OWS Common ^[39]. Umožňuje klientům získat službou metadata ze serveru. Odpověď na dotaz GetCapabilities musí obsahovat metadata o serveru včetně specifických informací o SPS, je to vlastně popis možností služby ^[42].

DescribeTasking

Tato operace (povinná) umožňuje klientovi požádat si o informace, které jsou potřebné k přípravě zadání požadavku na snímací systémy, které jsou podporovány SPS a které jsou vybrány klientem. Server vrátí informace o všech parametrech, které musí být uživatelem stanoveny k provedení operace Submit (tedy k potvrzení požadavku). Odpovědí na DescribeTasking je DescribeTaskingRequestResponse ^[42].

Submit

Tato povinná operace předkládá úkol. Závisí na snímacím zařízení, může se provést jednoduchou modifikací snímacího zařízení nebo startem komplexní úlohy ^[26]. Submit operace umožňuje klientům SPS odeslat požadavek. Normální odpovědí na platnou operaci Submit je SubmitRequestResponse ^[42].

GetStatus

Tato nepovinná operace umožňuje klientovi získat informace o aktuálním stavu požadovaného úkolu. Odpovědí na platnou operaci GetStatus je GetStatusRequestResponse. Mohou nastat tyto situace (Tab. 2):

Tab. 2: Možné odpovědi na operaci GetStatus ^[42]

Stav	Význam
not yet started	Ještě nezačal.
in operation	Již začal.
finished	Je ukončen.
cancelled	Byl úspěšně zrušen.
delayed	Je zpožděný SPS.
unknown	Nelze určit jeho stav.

GetTask

Tato funkce vrací kompletní informace o požadovaného úkolu ^[26].

DescribeResultAccess

Operace (povinná) umožňuje klientovi získat informace jak a kde lze k datům, která byla vyprodukována snímacím zařízením získat přístup. Odpověď serveru může obsahovat odkazy na všechny druhy přístupů k datům OGC webových služeb jako SOS, WMS, GVS, WFS nebo k jiným webovým službám. Odpovědí na platnou operaci DescribeResultAccess je DescribeResultAccessRequestResponse ^[42].

b) *SensorProvider* (povinné) - specifikuje následující operaci:

DescribeSensor

Tato operace umožňuje klientovi požádat o podrobný popis senzoru. Žádost může být zaměřena na popis, který byl platný v určitém místě nebo v určitém období v minulosti ^[26].

c) *ReservationManager* (volitelné) - Toto volitelné rozhraní umožňuje klientům rezervovat úkol namísto přímého předložení. To umožňuje zkoušení skupiny SPS. Rezervované úkoly mají omezenou životnost, než vyprší. Po tuto dobu může být tento úkol potvrzen tak, že se provede spuštění služby ^[26]. Rozhraní obsahuje následující operace:

Reserve

Tato operace rezervuje úkol. Rezervace trvá určitou dobu, během níž lze úkol potvrdit.

Confirm

Tato operace slouží k potvrzení rezervovaného úkolu. Po potvrzení rezervovaného úkolu, SPS provede úkol ^[26].

d) *FeasibilityController* (volitelné) - SPS provedení tohoto rozhraní jsou schopné vyhodnotit proveditelnost úkolů. To umožňuje klientům předem ověřit jejich požadavky úkolování ^[26]. Rozhraní obsahuje následující operaci:

GetFeasibility

Nepovinná operace slouží k poskytnutí zpětné vazby klientovi o proveditelnosti požadovaných dotazů. V závislosti na typu snímacího systému SPS, SPS server pak jednoduše ověří, že požadované parametry jsou platné a jsou v souladu s některými obchodními pravidly a nebo to může být složitější operace, kdy server vypočítá využitelnost snímacího zařízení k provedení určitého dotazu ve smyslu místa, času, orientace, kalibrace atd. Normální odpovědí na platnou operaci GetFeasibility je GetFeasibilityRequestResponse ^[42].

e) *TaskUpdater* (volitelné) - Služba, která implementuje toto rozhraní umožňující klientům aktualizace rezervovaného nebo potvrzeného úkolu ^[26]. Rozhraní obsahuje následující operaci:

Update

Tato operace umožňuje klientovi aktualizovat dříve potvrzené požadavky. Odpovědí na platnou operaci UpdateRequest je UpdateRequestResponse ^[42].

f) *TaskCanceller* (volitelné) - Toto rozhraní, pokud je implementováno, klientům umožňuje zrušení rezervovaného nebo potvrzeného úkolu ^[26]. Rozhraní obsahuje následující operaci:

Cancel

Operace umožňuje klientovi zrušit dříve rezervované nebo potvrzené úkoly. Odpovědí na platnou operaci Cancel je CancelRequestResponse ^[42].

g) *SensorDescriptionManager* (volitelné). Specifikuje následující operaci:

UpdateSensorDescription

Tato operace umožňuje klientům aktualizovat popis senzoru ^[26].

Pozn. Detailní popis je dostupný z <http://www.opengeospatial.org/standards/sps>

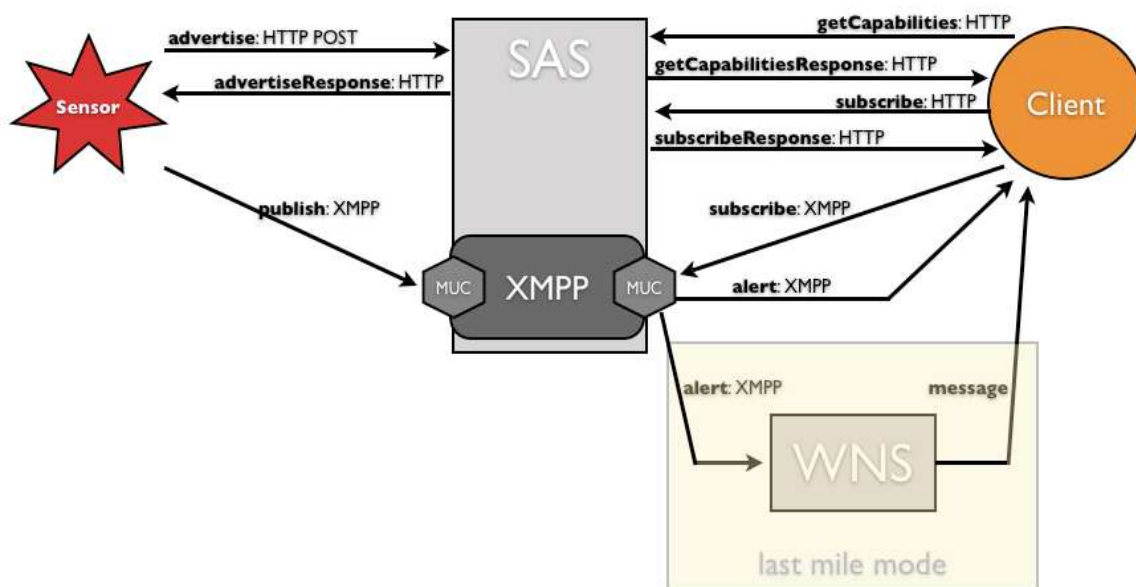
2.1.6 Sensor Alert Service (SAS)

Sensor Alert Service (SAS) je nejnovější částí sady dokumentů OGC Sensor Web Enablement. Sensor Alert Service může být srovnávána se systémem oznamování událostí. Předmětem zájmu je uzel senzoru. Každý uzel má propagovat své publikace na SAS (*advertise*). Pozn.: Ve skutečnosti jsou to „upozornění“, která senzor může posílat, a která musí být registrována spíše než uzel/senzor. Je důležité si uvědomit, že oba senzory, ty které pošlou aktuální hodnotu pozorování jako varování a senzory, které umí poslat upozornění, jako je „nad horní hranicí“ a „pod dolní hranicí“, se mohou registrovat na SAS. To umožňuje připojit více „chytrých“ senzorů k SAS stejně jako ty, které jen posílají aktuální pozorování. Není žádný rozdíl mezi těmito typy snímačů. Pokud čidlo jen vysílá svou aktuální hodnotu sledování, je na SAS, aby kontrolovala své přednastavené tabulky, zda je aktuální hodnota nad nebo pod uživatelem definovanou hranicí. Totéž platí pro ta upozornění, která jsou odesílána ze senzorů, která mohou rozhodovat sama o sobě, pokud je hranice překročena nebo nedosažena ^[23].

Všechny typy upozornění se musí zaregistrovat nebo zveřejnit, aby mohla být potenciálně zaslána. Pokud dojde k události, uzel ji pošle na SAS prostřednictvím operace

publikování. Spotřebitel (zájemce) se může přihlásit k odběru upozornění šířených SAS. Pokud nastane událost, bude SAS o události informovat všechny klienty přihlášené k odběru tohoto typu události. Daná SAS specifikuje rozhraní, které umožňuje uzlům inzerovat a zveřejňovat data pozorování respektive jejich metadata. Je důležité zdůraznit, že SAS sama se chová spíše jako registr než systém oznamování událostí ^[23].

Tradiční OGC webové služby nejsou vhodné pro provádění této upozorňovací služby. Operace SAS můžeme klasifikovat na operace používané producentem dat (senzorem) a na operace, které jsou využívány spotřebitelem dat (klientem). V kontrastu s dalšími OGC službami, SAS nepoužívá pouze HTTP protokol. Některé operace jako například operace GetCapabilities používají HTTP protokol, ale jiné jako například operace přihlášení (subscribe) používají kromě toho Extensible Messaging and Presence Protocol (XMPP) protokol ^[37]. Sensor Alert Service užívá Extensible Messaging a Presence Protocol (XMPP) k provádění postupně založených oznámení funkčnosti. Extensible Messaging a Presence Protocol (XMPP) je IETF ustanovení na základě XML protokolů streamování pro bezprostřední komunikaci a je současně vyvíjeno od roku 1999 s Jabber komunitou ^[23].



Obr. 6: Pohled na SAS a protokoly používané v různých krocích ^[23]

SAS rozhraní (v současné době) specifikuje sedm operací, které mohou být klientem požadovány a SAS serverem provedeny. Mezi tyto operace patří:

GetCapabilities

GetCapabilities (servery vyžadovaná implementace) – Povinná operace povoluje klientovi dotazovat se na a obdržet zpět XML dokument metadata služby, který popisuje schopnosti určitého zavedeného serveru. Tato operace také podporuje vyjednání verze specifikace používané pro interakci klient-server ^[23].

Advertise

Advertise (servery vyžadovaná implementace) – Tato povinná operace dovoluje producentům inzerovat typ publikované informace. V případě, že data jsou produktem nějakého předzpracování, např. senzor hlásí nízký stav baterie, mohou být tato data považována za upozornění. Ve skutečnosti se jedná pouze o zveřejněná pozorování ^[23].

Operace smí být použita senzory schopnými zahájit HTTP volání nebo administrátory senzorů, kteří chtějí senzory registrovat ^[23].

SAS vrací víceuživatelskou chatovací místnost (Multi-UserChat room (MUC)) používající HTTP. Senzor se pak zaregistruje s tímto MUC, aby mohl publikovat data. Tato registrace je ve skutečnosti (to znamená v terminologii XMPP protokolu) odebrání MUC a používá XMPP. Inzerce obsahuje následující informace ^[37]:

- prvek zájmu,
- operace,
- struktura upozorňovací zprávy.

Odpověď na Advertise se skládá z:

- odebrání koncového bodu (MUC),
- času do vypršení možnosti publikování,
- id publikace.

CancelAdvertisement

CancelAdvertisement (servery vyžadovaná implementace) – Tato povinná operace umožňuje producentům zrušit dříve registrované reklamní nabídky. Operace je používána zejména řídicími jednotkami čidel, které chtějí zrušit registraci senzoru na SAS ^[23]. Producent zašle CancelAdvertisement požadavek pomocí HTTP.

RenewAdvertisement

RenewAdvertisement (servery vyžadovaná implementace) – Tato povinná operace umožňuje producentům obnovit předchozí inzeráty. Operace je užívána hlavně řídicími jednotkami senzorů, které byly registrovány na SAS, ale jejich inzerční doba, jenž měly nastavenou, již na SAS vypršela ^[23].

Subscribe

Subscribe (servery vyžadovaná implementace) - Tato povinná operace umožňuje klientům odebírat upozornění, která jsou inzerována a zveřejněna na SAS. Spotřebitel může definovat vlastní upozornění, na základě inzerovaných a publikovaných

upozornění^[23]. To se provádí pomocí omezujících prvků, jako jsou výsledek filtru, umístění ploch apod.

Klient odešle požadavek k odběru, včetně SubscriptionOfferingID pomocí HTTP k SAS. SAS kontroluje stav odebírání a vrátí odpovídající MUC, stejně jako čas, ve kterém se tento odběr automaticky ukončí. Klienti se musí přihlásit do MUC přes XMPP, aby mohli přijímat upozornění. SAS podporuje následující funkce filtru^[37]:

- prostorové filtrování:

 - v místě*

 - v oblasti (bounding box)*

- pomocí ID nabídky odebírání

- pomocí ID prvku

- filtr obsahu:

 - menší než*

 - větší než*

 - rovno*

 - není rovno*

CancelSubscription

CancelSubscription (servery vyžadovaná implementace) – Tato operace umožňuje klientům zrušit odebírání.

RenewSubscription

RenewSubscription (servery vyžadovaná implementace) – Tato operace umožňuje klientům obnovit odebírání, které vypršelo. V případě, že je bývalý MUC již odstraněn SAS, nový MUC může být otevřen (nebo starý znovu otevřen)^[23].

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/bp>

2.1.7 Web Notification Service (WNS)

Web Notification Service model zahrnuje dva různé druhy oznámení:

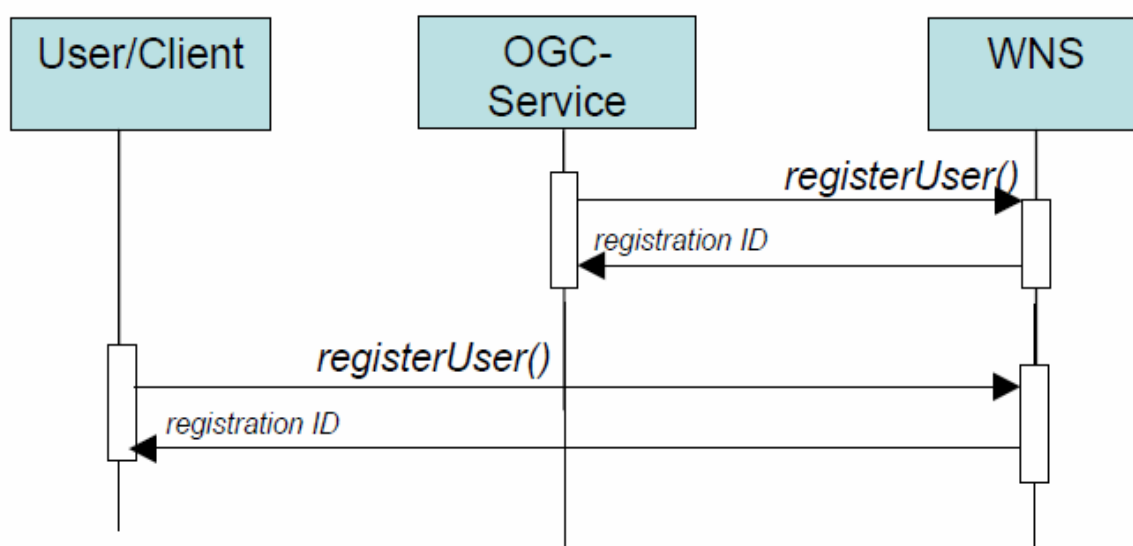
- 1) „**jednosměrná komunikace**“ poskytuje uživateli informaci, aniž by očekávala odpověď.
- 2) „**obousměrná komunikace**“ poskytuje uživateli informace a očekává nějakou asynchronní odpověď. Z tohoto rozlišení vyplývá rozdíl mezi jednoduchou

a sofistikovanou WNS. Jednoduchá WNS poskytuje možnost informovat uživatele nebo službu o konkrétní události. Navíc je schopna dostat od uživatele odpověď^[30].

Základ, na němž bude zasláno oznámení je zdarma ke službě a bude popsán v jejích schopnostech. Škála „způsobů oznámení“ může obsahovat:

- e-mail
- HTTP volání (například HTTP POST: v případě náročných klientů, kteří fungují sami jako webové služby)
- SMS
- okamžité zprávy
- telefonní hovor
- fax
- dopis

WNS musí poskytovat alespoň jeden z popsanych způsobů oznamování. Chceme-li využít možnosti oznámení, uživatelé musí být registrováni předem. Tato registrace bude provedena buď uživatelem, nebo OGC službou, která může fungovat jako proxy pro uživatele, který využívá funkci oznámení (např. SPS). Obrázek č. 7 ukazuje toto chování.



Obr. 7: Diagram procesu registrace^[30]

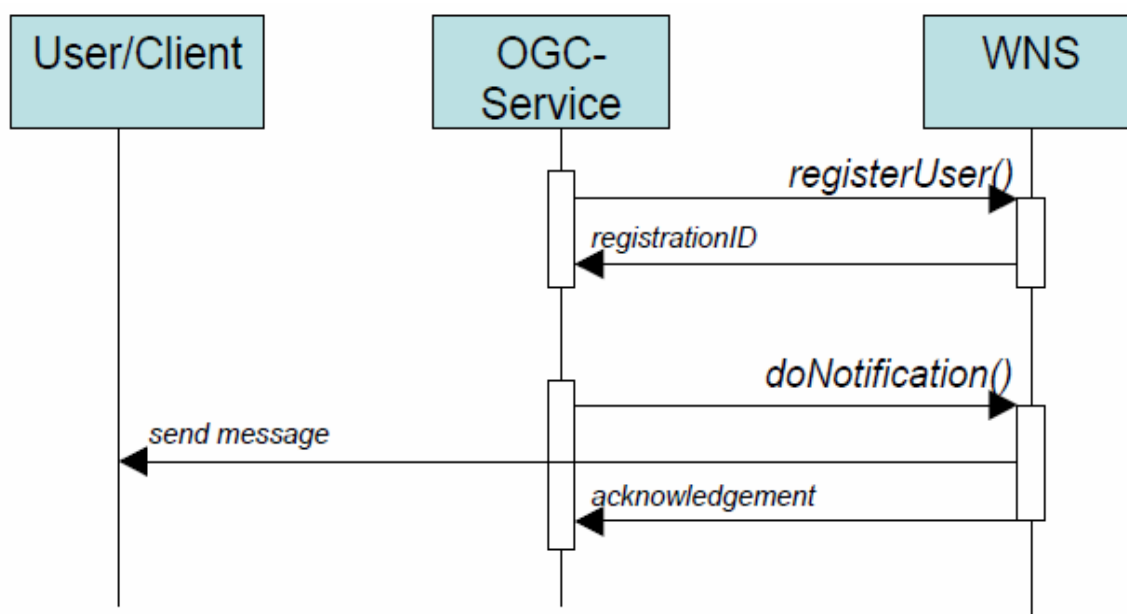
Služba OGC, která využívá WNS musí poskytnout registrační údaje o uživateli (identifikace) a musí poskytnout cíl oznámení (např. informace o e-mailové adrese, telefonní číslo, atd.). Na oplátku služba obdrží registrační ID nebo výjimku jako potvrzení.

Nezávisle na registraci žadatele (uživatele služeb, který se chová jako proxy) zatím není k dispozici žádný uživatelský mechanismus k ověřování dat. Například, pokud

uživatel (klient služby) požaduje nákladný sběr dat, nemůže být (službou) upozorněn, pokud poskytnul při registraci chybnou e-mailovou adresu. V dalším vývojovém kroku bude WNS vybavena správou vnitřního stavu a bude poskytovat potřebné rozhraní, které umožňuje uživatelům/službám zkontrolovat, zda došlo při předchozích operacích k chybě [30].

Jednosměrné oznámení

„Jednosměrná komunikace“ definuje nejjednodušší WNS. Tato verze WNS poskytuje zařízení pro zpracování registrace uživatelů a zasílání upozornění. Po registraci, obrácením na servis, je možnost zaslat upozornění pro uživatele/klienta voláním *doNotification()* operace, jak je znázorněno na obrázku č. 8.



Obr. 8: Diagram jednosměrné WNS [30]

Uznání *doNotification()* žádosti obsahuje zprávu o stavu. Stavy jsou:

- oznámení úspěšně odesláno,
- oznámení neúspěšně odesláno,
- oznámení vypršelo.

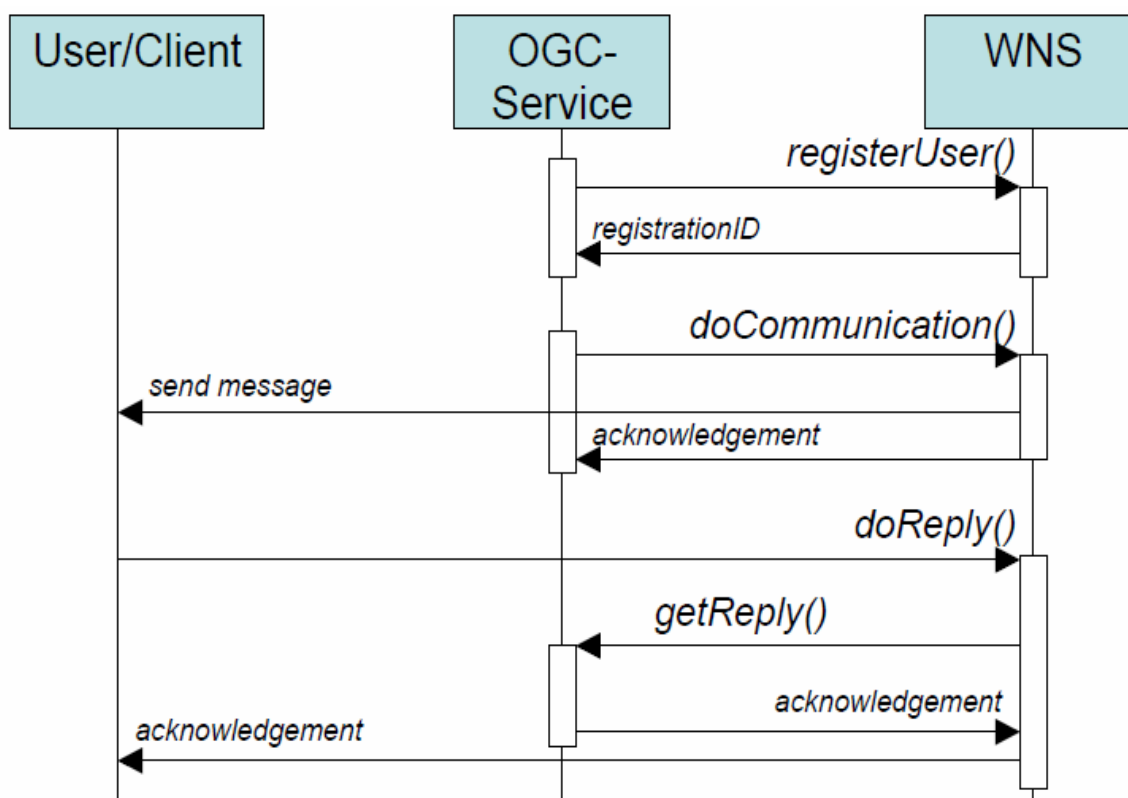
Potvrzení neposkytuje žádné informace o obdržení oznámení, kterým je:

- „Oznámení úspěšně odesláno“ neznamená, že uživatel obdržel oznámení pro všechny události. Je to jen signál, že oznámení bylo úspěšně odesláno na cestu pomocí WNS.

- „Oznámení neúspěšně odesláno“ znamená, že oznámení bylo zasláno, ale bylo vráceno do WNS, což znamená, že bylo nedoručitelné (např. kvůli chybné e-mailové adrese).
- „Oznámení vypršelo“ znamená, že WNS nebyl schopen poslat oznámení ^[30].

Obousměrné oznámení

„Obousměrné oznámení“ představuje rozšířenou WNS. Tyto služby poskytují všechny možnosti popsané výše. Kromě toho, rozšířená WNS poskytuje rozhraní pro obdržení odpovědí oznámení zaslaných uživateli. Proto služba poskytuje funkce pro vytvoření virtuální asynchronní komunikace s uživatelem. Obrázek č. 9 ukazuje tento postup.



Obr. 9: Diagram obousměrné WNS ^[30]

Pro zjednodušení operačního použití v OWS Testbed 1.2, regulační sekvence uvedená výše není plně podporována. Uživatelské odpovědi jsou v současné době posílány přímo do volané OGC služby. To znamená, že OGC služby nebo následné komponenty, musí být schopny přijímat odpovědi.

K uvolnění volané služby od břemene zajištění rozhraní pro různé protokoly oznámení, bude uživatelská odpověď odeslána zpět do WNS v budoucí verzi, podporující celou sekvenci, jak je znázorněno na obrázku č. 9. Proto bude uživatel moci odpovědět přes všechny komunikační protokoly poskytované WNS (např. SMS, HTTP příkazy, e-mail...). Tyto vysoce pokročilé funkce budou stanoveny podrobněji v budoucnu ^[30].

Komunikační model

Oznámení jsou zprávy, které jsou odesílány ze služby pro klienta pomocí libovolných protokolů. Obecně existují dva druhy, které je třeba rozlišovat: *prosté oznámení*, které pouze informuje klienta, že došlo k nějaké události, bez požádání klientovi odpovědi a *komplexní (virtuální) oznámení*. To vyžaduje, aby klient poskytl informace pro některé specifické otázky. Existují dva různé požadavky pro tyto případy: *doNotification* požadavek splňuje potřeby jednoduchého oznámení, zatímco *doCommunication* požadavek vyvolává složité procedury oznámení/odpovědi.

Sama zpráva oznámení je dobře definovanou strukturou, závislou na schopnostech WNS. Je závislá na zařízení. V případě SMS oznámení, bude WNS posílat poměrně malý fragment, zatímco velký XML soubor může být poslán na klientův internetový server. Konverze kódu XML zprávy poskytované službou volání na vhodné oznámení je úkolem Web Notification Service.

Volající do OGC Web Services obecně musí vědět, jaké protokoly jsou k dispozici pro zaslání oznámení konkrétnímu uživateli, vzhledem k jejich odpovědnosti k registraci uživatele s WNS. Prakticky stačí, že předají informace, které uživatel poskytl WNS při registraci, což znamená, že nemají představu, jak by toho konkrétní uživatel mohl dosáhnout obecně. Tato znalost je uložena na WNS, a to výhradně. Pro zjednodušení používání WNS pro jiné webové služby, jediný způsob, jak informovat uživatele, je odesláním kódu XML zprávy do WNS ^[30].

doNotification schéma

V případě prostého oznámení, klientská (volání) služba musí přijmout společné NotificationEventSchema. Toto schéma definuje možný seznam vlastností, které popisují uživatele, které lze zasílat od služby do WNS v požadavku doNotification. Klientská služba musí určit, které oznámení událostí podporuje v rámci svých možností. To musí být sladěno s ostatními OGC webovými službami, které využívají WNS, např. SPS. Schémata budou uložena na centrálním místě a/nebo na každém Web Notification Server. Protože to není povinné, není potřeba měnit stávající OGC webové služby ^[30].

doCommunication schéma

V případě složitých oznámení tzv. komunikace, každá klientská služba musí definovat své vlastní klíče. Klíče by měly být popsány v jeho schopnostech, s cílem zajistit společné sémantické porozumění. Kód XML zprávy zasláné do WNS v *doCommunication* požadavku přetrvává hlavně na následujících parametrech:

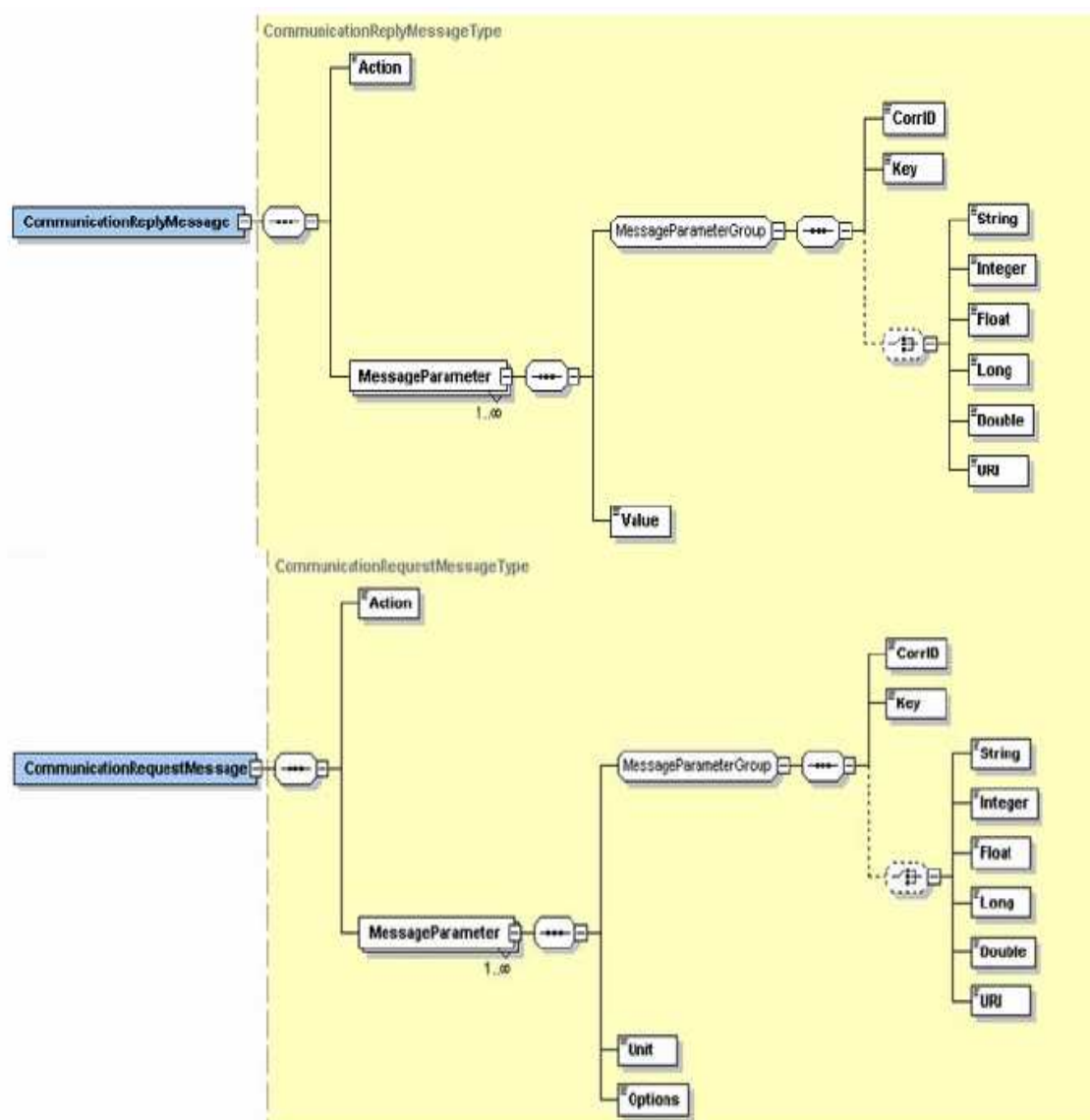
- *Action*: akce definuje operace, které služba potřebuje, například, že potřebuje od uživatele více informací.
- *CorrID*: korelace ID musí být poskytnuta ze strany klientské služby a umožňuje mapování uživatelských odpovědí.

- *Key*: klíč popisuje parametr a druhy interakce, které musí být provedeny, například „potřebná přesnost“.
- *Unit*: jednotka je volitelný parametr, který definuje jednotku požadovaného vstupu.
- *Option*: možnost je volitelný parametr, který obsahuje seznam možných variant vstupu, například "ano" a "ne", seznam čísel: "2,0", "2,5", "3,0", "3,5".
- *Value*: část hodnota se skládá z výběru bloku jednoduchých datových typů, které by mohly být využity pro další popis, na základě těch uvedených v „Key“ uzlu.

Model komunikace odezvy

První část dialogu, zahájení dialogu oznámení webovou službou OGC a následné odeslání zprávy z Web Notification Service je popsána výše. Po oznámení, je na uživateli, aby učinil další kroky. Existuje několik způsobů, jak může Web Notification Service zaslat oznámení. Podporované protokoly jsou popsány v jejich schopnostech. Na druhou stranu, existuje několik způsobů, jak získat odpověď od uživatele. Ačkoli společný protokol používaný pro odpověď bude jistě HTTP (volání *doReply* rozhraní WNS), mohou být podporovány také další protokoly. WNS může poskytnout zpětné volání rozhraní pro SMS, telefonní hovory, fax, apod. Standardizace o tom, jak se tyto protokoly používají, je velmi pokročilou funkcí.

V současné době je podporováno pouze zpětné volání rozhraní pro uživatele odpovědi HTTP. Toto omezení neomezuje WNS používat jiné protokoly pro posílání jejího počátečního oznámení. XML část *doCommunication* odpovědi od uživatele, výslovně operace *doReply*, se liší od *doCommunication* požadavku pouze v nahrazení jednotky a možností uzlů hodnotou uzlu, která obsahuje uživatelem poskytnutou odpověď na požadavek ^[30]. Toto je objasněno na následujícím obrázku.



Obr. 10: Srovnání CommunicationReplyMessage (nahore) a CommunicationRequestMessage (níže) ^[30]

Operace

WNS operace lze rozdělit na operace registrace a oznamování. doReply operace je poměrně pokročilou funkcí a nebyla dosud zcela definována.

getCapabilities operace (vyžadována)

Účelem operace getCapabilities je popsat možnosti služby. Z toho vyplývá, pokyny uvedené v OWS 1.2 Service Information Model. V konkrétním případě Web Notification Service, reakce getCapabilities požadavku je obecná informace o službě samotné a konkrétní informace o možných protokolech pro oznámení a povinných provozních parametrech.

Následující otázky musí být zodpovězeny v možnostech:

- Jaké protokoly oznámení jsou podporovány?
- Jaké parametry jsou povinné pro registraci?
- Jaké druhy protokolů reakce jsou podporovány?

registerUser operace (vyžadována)

Operace registerUser umožňuje registrujícím uživatelům přijímat oznámení. Uživatelská adresa a communicationProtocol musí být poskytnuty. WNS musí zajistit správu UserID. UserID jsou poskytována jako Long hodnoty.

doNotification operace (vyžadována)

doNotification požadavek je tzv. zahájení oznámení o uživateli.

doCommunication operace

doCommunication požadavek je tzv. zahájení komunikace s uživatelem. Komunikace přispívá asynchronní strukturou dialogu, to znamená, že WNS pošle oznámení pro uživatele. Toto oznámení znamená pro uživatele, že další kroky je třeba brát (což lze provést automaticky).

doReply operace

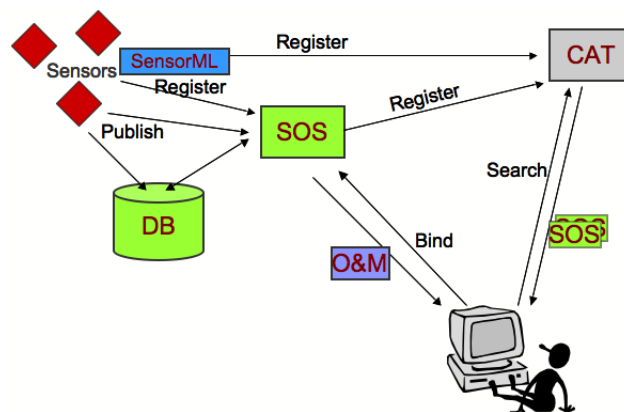
doReply operace je jeden způsob pro uživatele, jak reagovat na dialog oznámení. Může to být metoda, která se nejčastěji používá pro tento účel ^[30].

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/bp>

Pozn.: V rámci iniciativy Sensor Web Enablement jsou standardy SOS, SPS, SAS a WNS čtyři standardy definic rozhraní pro webové služby.

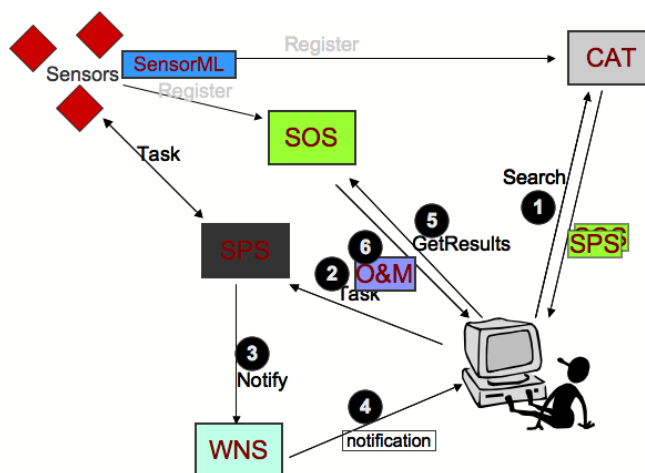
2.1.8 Interakce mezi SWE službami a SWE kódováními

Interakce mezi SWE službami a SWE kódováními jsou zobrazeny na obrázku č.11. Senzory jsou registrovány na SOS službě a publikují zde výsledky pozorování nebo je publikují do databáze, ke které má služba SOS přístup. K objevení senzorů se používá kódování SensorML a SOS registr v katalogu služeb. Uživatel požaduje data pozorování, a tak posílá do katalogu služeb vyhledávací požadavek. Katalog uživateli odpoví seznamem dostupných SOS služeb, které vyhovují požadavku uživatele. Nakonec se uživatel spojí s SOS službou, pomocí které získává data pozorování, kódovaná v O&M.



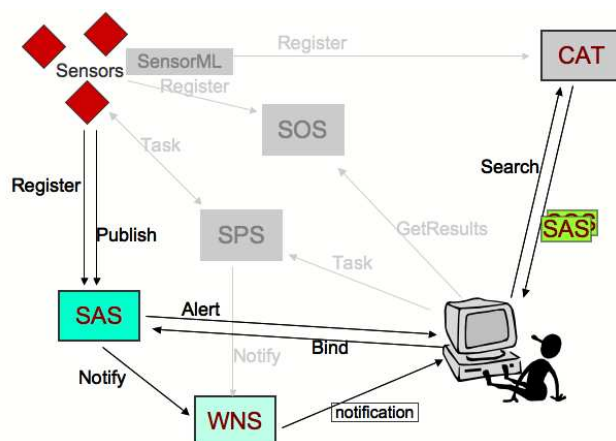
Obr. 11: Interakce mezi SWE službami a SWE kódováními, část první ^[27]

Na následujícím obrázku č. 12 je situace o něco složitější. Lze předpokládat, že v katalogu služeb se nevyskytuje žádná SOS služba, která splňuje požadavky uživatele. V tomto případě může uživatel vyhledat SPS, kterou by mohl senzory úkolovat za účelem vytvoření dat pozorování, která uživatel hledá. Katalog obsahuje odkaz na SPS, kterou může uživatel úkolovat. SPS předá úkoly senzoru. Komunikace mezi senzory a SPS je pro uživatele neprůhledná. Senzory ukládají data do databáze, která je spojena s SOS. SPS informuje uživatele dat, že jsou data dostupná prostřednictvím WNS. Zpráva oznámení obsahuje všechny nezbytné informace pro přístup k datům z SOS.



Obr. 12: Interakce mezi SWE službami a SWE kódováními, část druhá ^[27]

Často se setkáváme se situací, že uživatel nemá zájem o všechny výsledky pozorování senzoru, ale požaduje pouze okamžité oznámení, pokud nastane specifická situace. To ukazuje obrázek č. 13.



Obr. 13: Interakce mezi SWE službami a SWE kódováním, část třetí ^[27]

Opět platí, že klient získá informace o vhodných SAS z katalogu, kde si zvolí vybranou SAS. Senzory průběžně zveřejňují výsledky pozorování do SAS. SAS zpracovává všechny filtrování a upozorňuje klienta v případě, že přednastavené podmínky souhlasí. SAS posílá upozornění přímo klientovi nebo využívá WNS s cílem poskytovat výstrahu.

2.2 Další technologie, standardy a protokoly pro vytváření senzorových sítí

2.2.1 6LoWPAN

Je zkratka pro IPv6 pro *Low power Wireless Personal Area Networks*. Skupina 6LoWPAN definovala zapouzdření a hlavičkové kompresní mechanismy, které umožňují zasílat a obdržet IPv6 pakety od IEEE 802.15.4 založených sítí. IPv4 a IPv6 jsou pracovní koně pro poskytování údajů pro místní počítačové sítě (LAN), metropolitní sítě (MAN) a rozlehlé sítě (WAN), jako je například Internet. Podobně, IEEE 802.15.4 zařízení poskytují komunikaci schopnou snímání v bezdrátové oblasti ^[50].

2.2.2 IEEE 802.15.4

Jedná se o standard, který specifikuje fyzickou vrstvu a řízení přístupu k médiím pro *low-rate wireless personal area networks* (LR-WPANs). Je udržován pracovní skupinou 802.15.

Je základem pro ZigBee, ISA100.11a, WirelessHART a MiWi specifikace, z nichž každá dále rozšiřuje standard rozvojem horních vrstev, které 802.15.4 nedefinuje. Eventuálně může být použit s 6LoWPAN a standardními internetovými protokoly pro vybudování *Wireless Embedded Internet* ^[15].

Architektura protokolu

Zařízení jsou koncipována tak, aby jejich vzájemná interakce probíhala skrze koncepčně jednoduchou bezdrátovou síť. Definice jednotlivých síťových vrstev je založena na modelu OSI (Open System Interconnection).

Fyzická vrstva poskytuje službu přenosu dat, stejně jako rozhraní pro řízení entit fyzické vrstvy, které nabízí přístup ke každé funkci vrstvy a k databázi informací souvisejících s Personal Area Networks (PAN). Pracuje na jednom ze tří nelicencovaných frekvenčních pásem:

- 868,0-868,6 MHz: Evropa, umožňuje jeden komunikační kanál
- 902-928 MHz: Severní Amerika, do roku 2003 umožňoval až deset kanálů, 2006 rozšířen na třicet kanálů
- 2400-2483.5 MHz: použití po celém světě, umožňuje až šestnáct kanálů

MAC (Medium Access Control) vrstva umožňuje přenos MAC rámců pomocí fyzického kanálu. Kromě datových služeb nabízí rozhraní pro správu a sama spravuje přístup k fyzickému kanálu a signalizaci sítě.

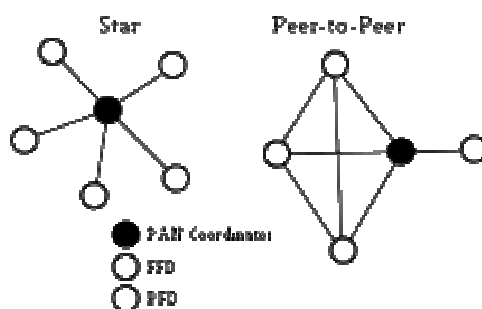
Ostatní vrstvy vyšší úrovně a podvrstvy nejsou touto normou definovány. Existuje specifikace, jako například ZigBee, které navazují na tento standard pro navržení základního řešení.

Síťový model

Standard definuje dva typy uzlů sítě:

- 1) plně funkční zařízení (full function device – FFD), které může sloužit jako koordinátor PAN, stejně jako běžný uzel.
- 2) omezené funkční zařízení (reduced function device – RFD), které komunikují pouze s FFD a nikdy nemohou sloužit jako koordinátor sítě.

Topologie sítě může být peer-to-peer nebo hvězdicová.



Obr. 14: Hvězdicová a peer-to-peer topologie ^[15]

Přenos dat

Základní jednotkou pro přenos dat jsou rámce. Existují čtyři základní typy rámců (data, potvrzení, světelný signál a MAC příkaz) ^[15].

2.2.3 ZigBee

ZigBee je bezdrátová komunikační technologie vystavěná na standardu IEEE 802.15.4. Jedná se o poměrně nový standard, platný od listopadu 2004.

Podobně jako Bluetooth je určena pro spojení nízkovýkonných zařízení v bezdrátových sítích PAN (Personal Area Networks) na malé vzdálenosti do 75 metrů. Díky použití multiskokového ad-hoc směrování umožňuje komunikaci i na větší vzdálenosti bez přímé radiové viditelnosti jednotlivých zařízení. Primární určení směřuje do aplikací v průmyslu a senzorových sítích ^[48].

Pracuje v bezlicenčních pásmech (generální povolení) přibližně 868 MHz, 902–928 MHz a 2,4 GHz. Přenosová rychlost činí 20, 40, 250 kbit/s.

Díky různorodosti předpokládaných aplikací standard definuje tři základní režimy přenosu dat:

- periodicky se opakující (přenos dat z čidel)
- nepravidelné přenosy (externí události, např. stisknutí tlačítka uživatelem)
- opakující se přenosy u nichž je požadavek na malé zpoždění (bezdrátové počítačové periferie – klávesnice a myši).

Struktura komunikačního standardu

Protokol se skládá ze tří základních vrstev. Vrstvy standardu IEEE 802.15.4, nad nimi je definována síťová vrstva (NWK) a aplikační vrstva (APL). Fyzická vrstva specifikuje přístup k přenosovému médiumu. Síťová vrstva realizuje připojení k síti, zabezpečení a směrování paketů. Aplikační vrstva (APL) zajišťuje potřebné služby. Skládá se z aplikační podvrstvy (APS), ZigBee objektů a uživatelských aplikačních objektů.

MAC vrstva (linková) definuje samotný komunikační protokol, jež je založen na přenosu datových rámců. Jsou definovány čtyři typy komunikačních rámců, jež jsou využívány pro přenos dat, řízení či správu sítě:

- Data Frame – rámec využíváný pro všechny přenosy užitečných dat
- Acknowledgement Frame – rámec pro přenos potvrzovací informace, využíván na úrovni MAC pro potvrzování komunikace
- Beacon Frame – rámec používaný koordinátorem k vysílání tzv. *beacons* (používané pro uvádění klientských zařízení do spánkového režimu)
- MAC Command Frame – rámec k nastavování a řízení klientských zařízení v síti ZigBee

Technologie ZigBee postavená na fyzické linkové vrstvě IEEE 802.15.4 definuje tři různé síťové topologie. Základní topologií je topologie hvězdicová s centrálním řídicím uzlem (koordinátorem sítě). Druhým typem je stromová struktura, jež umožňuje zvětšit vzdálenost mezi koordinátorem a koncovým zařízením. Protokol též umožňuje vytvoření

redundancí spojení a vzniká tak topologie typu sítě *mesh*. S její pomocí je možné vytvořit síť prakticky libovolného uspořádání.

Standard ZigBee dělí zařízení na zařízení FFD (*Full Functional Device*) a RFD (*Reduced Functionality Device*). FFD zařízení implementují kompletní protokolový rámec a zajišťují veškeré služby, které standard ZigBee stanovuje. RFD zařízení implementují pouze nezbytné protokolové knihovny z důvodu maximálního omezení hardwarové náročnosti. Tato zařízení mohou pracovat pouze jako koncová. Mohou komunikovat pouze s koordinátorem sítě a jsou omezeny na hvězdicové uspořádání topologie (koncové větve). Koordinátor sítě a směrovače jsou realizovány FFD zařízeními.

Síťová vrstva (NWK) zajišťuje připojení k síti, zabezpečení, směrování a synchronizaci. V případě koordinátora sítě je ještě zodpovědná za start sítě a přiřazování adres nově nalezeným zařízením.

Aplikační vrstva protokolu (APL) se skládá z pomocné aplikační APS podvrstvy, ZigBee objektů (ZDO) a uživatelem definovaných aplikačních objektů. Pomocná aplikační podvrstva je zodpovědná za párování zařízení podle poskytovaných služeb a požadavků. To je realizováno pomocí tzv. párovací (binding) tabulky. ZigBee objekt definuje roli jednotlivých zařízení v rámci sítě (koordinátor, směrovač, koncové zařízení). Dále zajišťuje vyhledávání nových zařízení a jimi poskytovaných služeb. V neposlední řadě zodpovídá za zabezpečení (volí jeho způsob, jako např. veřejné klíče, symetrické klíče). Uživatelské aplikační objekty implementují konkrétní požadavky aplikace dle definovaného ZigBee profilu. ZigBee profil zastřešuje definice možných zařízení, formátů a typů zpráv. Každý profil je určen unikátním 16 bitovým identifikátorem podle specifikace ZigBee Alliance ^[48].

2.2.4 JenNet

JenNet je síťový protokol navržený pro práci s bezdrátovými mikrokontroléry a moduly. JenNet umožňuje snadno vytvářet bezdrátové senzorové sítě (WNS), od jednoduchých až po komplexní stromové sítě. Podporuje hvězdicovou, stromovou a lineární topologii sítě a poskytuje funkce pro robustní komunikaci. Je vyvinut na základě standardu IEEE 802.15.4, který funguje po celém světě v pásmu 2,4 GHz. JenNet také spolupracuje s WiFi a Bluetooth ^[17].

2.2.5 WirelessHART

WirelessHART je technologie bezdrátové senzorové sítě založená na protokolu HART (Highway Addressable Remote Transducer). Tento protokol využívá časově synchronizovanou, sebe organizující a sebe uzdravující mesh architekturu. Podporuje provoz v pásmu 2,4 GHz pomocí IEEE 82.15.4 standardu ^[46].

2.2.6 ISA100.11a

ISA100.11a je standard bezdrátové síťové technologie vyvinutý Mezinárodní společností pro automatizaci ISA (International Society of Automation). Oficiální popis je „Bezdrátové systémy pro průmyslovou automatizaci: řízení procesů a souvisejících aplikací“^[16].

2.2.7 Z-Wave

Z-Wave je soukromý (komerční) bezdrátový komunikační protokol určený pro domácí automatizaci, konkrétně pro dálkově ovládané aplikace v komerčních a užitkových prostředích. Z-Wave komunikace používá nízkoenergetické bezdrátové technologie určené speciálně pro dálkově ovládané aplikace. Protokol je optimalizován pro spolehlivou nízkou latentní komunikaci malých datových paketů, na rozdíl od WiFi a dalších IEEE 802.11 založených bezdrátových LAN systémech, které jsou určeny především pro vysokou šířku pásma datového toku. Z-Wave pracuje na sub-GHz frekvenci kolem 900 MHz. Toto pásmo konkuruje některým bezdrátovým telefonům a dalším zařízením spotřební elektroniky, ale vyhýbá se rušení WiFi a dalších systémů pracujících v přeplněném pásmu 2,4 GHz. Z-Wave je v současné době podporováno více než 200 výrobci z celého světa a objevuje se v širokém spektru spotřebního zboží v USA a Evropě. Norma sama o sobě není otevřená a je k dispozici pouze pro zákazníky Zensys. Z-Wave využívá cílově vedenou mesh topologii a má jeden nebo více kontrolorů, které řídí směrování a bezpečnost. Některé běžné aplikace pro Z-Wave: dálkově řízená kontrola a správa domu, bezpečnostní systémy a domácí zábava^[49].

2.2.8 ANTTM

ANTTM je osvědčený protokol a křemíkové řešení pro ultra nízkoenergetické aplikace senzorových sítí. Je určen pro provoz v 2,4 GHz. Výborně se hodí pro všechny druhy pomalých přenosů dat v topologii senzorových sítí v PAN (Personal Area Network) a WSN (Wireless Sensor Network).

ANT je záměrně navržen tak, aby zjednodušil praktické bezdrátové sítě, vývoj a optimální výkon sítě. ANT uzly sítě mohou pracovat až několik let na knoflíkovou baterku, na rozdíl od ostatních. Je bohatý na funkce. ANT poskytuje spolehlivé datové komunikace, pružné a přizpůsobivé síťové operace a odolnost proti přeslechům^[43].

ANT+ představuje interoperabilní funkce, které mohou být přidány do základního protokolu ANT. Uspodňuje shromažďování, automatický přenos a sledování senzorických dat pro monitorování veškerých osobních informací o osobním zdraví kdekoli a kdykoli. Pomocí ANT+ výrobci, projektující počítačové diagnostiky pro jízdu na kole, měřiče výkonu, monitorování srdečního tepu, atd., zajišťují kompatibilitu s dalšími předními dodavateli. ANT+ zaručuje bezproblémovou digitální bezdrátovou komunikaci, dlouhou výdrž baterie a komunikační spolehlivost^[44].

2.2.9 ONE-NET

ONE-NET je open-source standard pro bezdrátové sítě. Byl vytvořen pro nízkonákladové, nízkoenergetické sítě jako jsou automatické domácnosti, bezpečnost a sledování, kontrola zařízení a senzorové sítě. ONE-NET není vázán na žádný hardware ani software, a proto může být použit s různými vysílači a kontroléry od různých výrobců. ONE-NET používá UHF ISM radiové vysílače a operuje na frekvencích 868 MHz a 915 MHz s 25 kanály použitelnými ve Spojených státech. Standard umožňuje implementaci i na jiných frekvencích. Některé práce se provádí na frekvencích 433 MHz a 2,4 GHz. ONE-NET využívá širokopásmový FSK (Frequency-Shift Keying) k zašifrování dat pro přenos. ONE-NET nabízí dynamický protokol rychlosti přenosu dat se základní rychlostí 38,4 kbit/s. Tato specifikace umožňuje per-node dynamickou konfiguraci rychlosti přenosu dat až na 230 kbit/s. ONE-NET podporuje hvězdicovou, peer-to-peer a multiskokovou topologii. ONE-NET podporuje jednoduché, blokové a proudové transakce. Jednoduché transakce jsou zprávy typu teplota, spotřeba energie i kontrolní zprávy jako zapnutí/vypnutí zpráv. Blokové transakce se skládají z více paketů s obsahem až 58 byte. Bloková transakce může obsahovat až 65 535 byte. Proudové transakce jsou podobné blokovým, ale nevyžadují přenos ztracených datových paketů. ONE-NET je optimalizován pro nízkou spotřebu energie, využívá obvykle AA nebo AAA baterie. Nedílnou součástí ONE-NET je šifrování, neexistuje nic co by nebylo zašifrováno. Používá rozšířený Tiny šifrovací algoritmus (XTEA-eXtended Tiny Encryption Algorithm). ONE-NET je určen pro volné užití a používá zjednodušenou BSD licenci ^[31].

2.2.10 DASH7

DASH7 je open-source standard pro bezdrátové senzorové sítě, který působí v ISM bezlicenčním pásmu 433,92 MHz. Poskytuje letitou životnost baterie, dosah do 2 km, nízkou latenci pro propojení s pohyblivými věcmi, AES 128 bit šifrování s veřejným klíčem a přenos dat až 200 kbit/s. DASH7 následuje ISO/IEC 18000-7 otevřený standard pro bezdrátovou komunikaci, který definuje rozhraní pro radiovou frekvenční identifikaci (RFID) zařízení, které pracují jako aktivní RF tag na frekvenci 433 MHz. DASH7 podporuje tag-to-tag komunikaci, která v kombinaci s dlouhým dosahem a šířením signálu na frekvenci 433 MHz umožňuje snadno nahradit většinu bezdrátových „mesh“ technologií senzorových sítí. DASH7 je využitelný pro širokou škálu aplikací, např. automatizace budov, řízení přístupu, inteligentní energie, služby založené na poloze (Location Based Services – LBS), monitorování pneu, logistika. DASH7 je také využíván Ministerstvem obrany USA a dalšími armádami. Všechny armády NATO jsou nasazeny nebo v procesu nasazování infrastruktury DASH7 ^[7].

2.2.11 Wibree

Wibree, který je známý jako Ultra Low Power (ULP) Bluetooth nebo jako níkoenergetický Bluetooth, je to radiová technologie pro malé, tlačítkové zařízení na baterky, jako hodinky, bezdrátové klávesnice, aplikace pro monitorování zdraví a hrací ovladače. Tyto objekty mohou být připojeny k Bluetooth zařízením, ať už počítačům nebo mobilním telefonům^[45].

2.2.12 Extended Environments Markup Language (EEML)

EEML – Extended Environments Markup Language je značkovací jazyk, který popisuje výstupy dat ze senzorů, často v architektonickém kontextu, ale také v interaktivním prostředí, rozhraní zařízení a dokonce i v Second Life objektech. EEML zpracovává metadata z jakého zdroje byla data pořízena. To má význam pro stroje a lidi při vyhledávání datových toků, které částečně potřebují aniž by znali přesné údaje o jejich zdroji^[9].

Schéma bylo napsáno v XML pro formátování datových toků ze senzorů a zařízení rozmístěných ve fyzickém světě, např. v kancelářích, domech stejně jako ve virtuálním světě, např. Second Life. Data ze senzorů mohou být mapována v reálném čase a vyměňována se simulačními softwary a softwary pro správu zařízení. Současná verze specifikace je 0.5.1 dostupná z <http://www.eeml.org/>

3 ZPRACOVÁNÍ PROUDŮ DAT

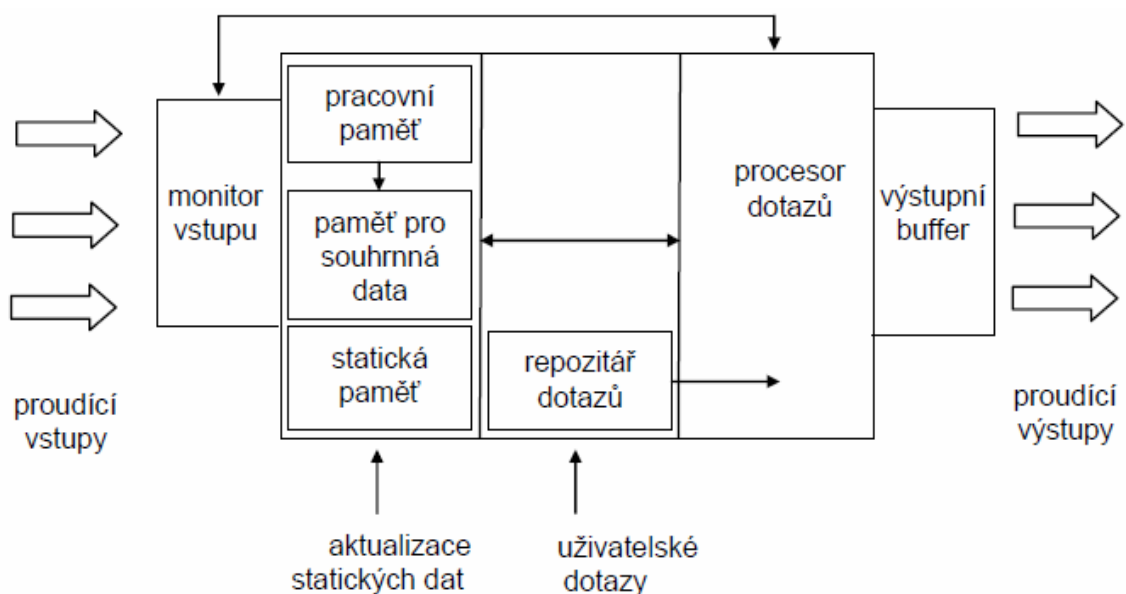
Správa dat přicházejících ze senzorů založená výlučně na tradičním modelu „ukládání a dotazování“ obvykle nemůže efektivně zacházet s objemem a rychlostí proudících dat, jejichž hodnoty mohou existovat jen na krátký okamžik. Tradiční DBMS jsou pro zacházení s proudy dat nevhodné z mnoha důvodů:

- senzorové uzly produkují a odesílají data nepřetržitě bez ohledu na to, existují-li přímé požadavky na tato data,
- dotazy nad příslušnými kolekcemi dat mohou být méně časté než vkládání dat do kolekcí,
- vytvořená data jsou často zpracovávána v reálném čase, protože mohou reprezentovat události, které potřebují okamžitou reakci,
- dotazy se zpracovávají nepřetržitě, protože proudy dat nikdy nekončí, takže mohou „vidět“, jak se mění podmínky systému během jejich volání,
- kvůli omezením na paměť, nemůže být proud dat uložen celý na vnější médium,
- protože proudy dat jsou potenciálně nekonečné, mohou být použity pouze neblokující operátory,
- jestliže množina dat ze senzorů není jako vstup do operátoru celá dostupná, pak se operátor zablokuje. ^[33]

3.1 Systémy řízení proudů dat

V důsledku těchto skutečností se objevily *systémy řízení proudů dat* (Stream Data Management System - SDMS). Příkladem budiž (work-flow) systém Aurora, společný projekt Brandeis University, Brown University, a M.I.T. *Procesor (stroj) pro zpracování proudů dat* (Stream Processing Engine - SPE) je pak příkladem nové databázové architektury, která dovoluje volání dotazů, výpočtů a akcí na proudících datech v reálném čase. Procesor by měl akceptovat proudově orientované dotazy zapsané v notaci SQL a spouštět je nad proudy událostí s výstupem v reálném čase. Zpracování je realizováno z větší části ve vnitřní paměti, operace čtení a zápisu na disk jsou volitelné a mohou být v mnoha případech zvládnuty asynchronně. Na logické úrovni jde o tři druhy paměti (viz obrázek 15): *pracovní* (pro dotazy typu okno), *paměť pro souhrnná data* o dotazech (sumarizace, synopse) a *statická paměť* pro metadata (např. fyzické umístění zdroje, rychlost proudu dat).

Nepřetržité či dlouhotrvající dotazy jsou registrovány v *repozitáři dotazů*, mohou mít zpracování sdílené ve skupině takových dotazů. Jednorázové dotazy, např. na stav proudu realizuje *procesor dotazů* přímo na vstupu. Tento procesor může také na základě zpráv z *monitoru vstupu* znovu optimalizovat plány nepřetržitých dotazů.



Obr. 15: Referenční architektura stroje pro zpracování proudů dat ^[34]

Speciální implementace procesorů zpracovávající nové typy dat jsou dnes trendem ve vývoji nových databázových architektur. Obvykle vedou k zvýšení výkonu. Např. v systému StreamBase vyvinutém Stonebrakerem v r. 2005 je možné analyzovat 140000 zpráv za sekundu, zatímco běžný relační DBMS by zvládl v témže čase pouze 900 zpráv. Důraz na zpracování ve vnitřní paměti pro některé dodavatele SDMS znamenal renesanci DBMS založených právě na tomto přístupu. Globální rozdíly mezi klasickými DBMS a SDMS ukazuje tabulka 3.

Tab. 3: SDMS vs. SŘBD: globální rozdíly ^[34]

	SDMS	SŘBD
zdroje	omezené (paměť, počítání po n-ticích)	bohaté (paměť, disk, počítání po n-ticích)
zpracování dotazu	rozumně složité, blízké zpracování v reálném čase	velmi sofistikované, analýzy
použití	zjištění, jaká data ukládat do databáze SŘBD	audit výsledků dotazu z SDMS

Další variantou architektury je SDMS kombinovaný s DBMS. V tomto případě jsou SDMS umístěny ve více bodech sítě v závislosti na topologii aplikace včetně klasického (relačního) DBMS. Do SDMS vstupují objemné proudy dat a vycházejí z něho redukované. Výstup ze SDMS tvoří vstup do DBMS pro další zpracování.

V současné době je možné pozorovat, že vývoj SDMS je řízen buď aplikacemi nebo technologiemi. V první případě je důležité dosáhnout sofistikovaného vyhodnocení dotazů resp. analýzy dat (bankovní podvody, počasí) ve skoro reálném čase. Druhý případ

zahrnuje efektivní práci s masivními objemy dat z transakcí a měření (data generovaná ze senzorových sítí, satelitů). Pro tvorbu aplikací využívající proudy dat se objevují nové programovací jazyky, např. Hancock vyvinutý v AT&T ^[34].

3.2 Speciální architektury

Zpracování proudů dat může být realizováno v mobilním prostředí, kde proudící data jsou bezdrátově vysílána ke klientským zařízením. V budoucnosti se zřejmě uplatní technologie RFID (Radio-Frequency IDentification). RFID neboli radiofrekvenční identifikace, je současná technologie založená na radiových vlnách pro přesnou identifikaci objektů pomocí značek EPC (Electronic Product Code). Jde o komunikaci vysílacího a přijímacího zařízení, jakési čtečky, s integrovaným obvodem vybaveným anténou (tagem). Událostí pak může být situace, když se nějaká RFID anténa dostane do dosahu senzoru. RFID technologie je použitelná např. pro kontrolu přístupu do objektu, vstupně-výstupní kontroly v knihovnách, evidence stopy pohybujícího se objektu (zboží, dokument, dítě ve škole), ve zdravotnictví (monitorování pacientů – e-health) apod. RFID data jsou opět generována v proudech. Pro proudy RFID dat je typickou úlohou odhalování duplicit a chybných čtení obsažených v proudu. RFID je klíčovou technologií pro realizaci Internetu věcí. Speciálním případem diskutovaných procesorů jsou ty, od kterých se vyžaduje zpracování v reálném čase. Po vzoru dnes již klasických souborů pravidel pro relační databáze či OLAP od E.F. Codda se uvádí 8 pravidel charakterizujících takové procesory ^[34]:

1. Zpracovávat data proudů bez zpoždění, tj. bez požadavku na uložení dat.
2. Podpora neprocedurálního dotazovacího jazyka vyšší úrovně, jakéhosi „StreamSQL“.
3. Umět ošetřit situace poruch v proudu dat (např. chybějící prvky).
4. Garantovat predikovatelné a opakovatelné výsledky.
5. Umět efektivně ukládat, přistupovat a modifikovat informace o stavu zpracování, když je to nutné.
6. Vysoká dostupnost v případě chyb v systému (rychlé zotavení z chyb).
7. Možnost distribuce zpracování a tím škálovatelnosti systému. Distribuce by měla být automatická a transparentní.
8. Zajistit odezvu systému v reálném čase i pro proudy dat vysokých objemů.

4 STANDARDY PRO PŘENOS, ULOŽENÍ A FILTROVÁNÍ DAT

4.1 Geography Markup Language (GML)

Geography Markup Language (geografický značkovací jazyk) je standard (verze 3.2.1) využívající jazyk XML pro popis aplikačních schémat, stejně jako k manipulaci a uchovávání geografických informací ^[22].

GML je určen pro modelování, přenos a skladování geografických dat. V mnoha předdefinovaných schématech GML poskytuje bohatou slovní zásobu, kterou lze použít k vytvoření specifického GML aplikačního schématu. GML slouží jako základ pro geoprostorový web a pro interoperabilitu nezávisle vyvinutých distribuovaných aplikací včetně Location Based Services ^[1].

Klíčové myšlenky, využívané v GML k modelování okolního světa, jsou přejaty ze série standardů ISO 19100 ^[22]. Hlavním cílem ISO 19100 řady standardů je umožnit plnou interoperabilitu mezi různými geografickými informačními systémy. K dosažení tohoto cíle je potřeba vyřešit dva základní problémy. Prvním problémem je definovat sémantiku obsahu a logickou strukturu geografických dat. Toho je dosaženo zavedením stejného aplikačního schématu. Druhá věc je definovat systémově a platformě nezávislou datovou strukturu, která může představovat údaje odpovídající aplikačnímu schématu ^[1]. Geography Markup Language je označován jako standard ISO 19136.

Podle standardu ISO 19109 jsou typy prvků aplikace zachyceny aplikačním schématem. GML aplikační schéma je zadáno v XML a může být sestaveno dvěma různými způsoby:

- Dodržením pravidel definovaných ve standardu ISO 19109 pro aplikační schémata UML a přizpůsobením omezení schémat a pravidel pro GML.
- Dodržením pravidel pro GML aplikační schémata přímo v XML.

GML je XML kódování ve shodě se standardem ISO 19118 pro transport a skladování geografických informací vymodelovaných v souladu s konceptuálním modelovacím rámcem používaným v sérii mezinárodních standardů ISO 19100 a zahrnující prostorové i neprostorové vlastnosti geografických prvků.

Tento mezinárodní standard definuje XML syntaxi, mechanismus a konvence, které:

- poskytují a otevírají rámce pro popis geoprostorových aplikačních schémat pro transport a skladování geografických informací v XML,
- povolují profily, které podporují správnou podmnožinu GML rámce popisných schopností,
- podporují popis geoprostorových aplikačních schémat pro specializované domény a informační pospolitost,

- povolují tvorbu a údržbu provázaných geografických schémat a datasetů,
- podporují skladování a transport aplikačních schémat a datasetů,
- zvyšují schopnost organizací sdílet geografická aplikační schémata a informace, které popisují.

Tvůrci se mohou rozhodnout, zda uloží geografická aplikační schémata a informace v GML, nebo se rozhodnou na požádání konvertovat z jiného úložného formátu a použijí GML pouze k schématům a datovému transportu ^[22].

4.1.1 GML schéma – Obecná pravidla a základní komponenty

GML používá explicitní syntaxi odpovídající General Feature Model definovanému v standardu ISO 19109 v XML dokumentu. Prvky jsou kódovány jako elementy XML se stejným názvem typu prvku. Ostatní rozpoznatelné objekty jsou kódovány jako elementy XML se stejným názvem jako typ objektu.

Každý atribut prvku a jeho role jsou vlastnosti prvku. Vlastnosti prvku jsou kódovány v elementu XML. Prvky jsou reprezentovány:

- v UML jako objekty, kde název typu prvku je použit jako název třídy objektu,
- v GML jako elementy XML, kde název typu prvku je použit jako název elementu.

Vlastnosti prvku jsou reprezentovány:

- v UML přiřazením rolí s třídami typů prvků a jejich atributy, kde vlastnosti sémantiky jsou dány přiřazením názvu role či atributu,
- v GML podelementy (property elements) prvků, kde vlastnost sémantiky je dána názvem vlastnosti prvku.

Hodnota vlastnosti má typ označený:

- v UML třídou přiřazeného cíle nebo datovým typem atributu,
- v GML: v případě vlastností s komplexními hodnotami názvem elementu objektu a v případě jednoduchých hodnot typem nebo textovou hodnotou neobsahující vložený XML znak.

Pravidla psaní

Existuje několik pravidel pro GML schémata pro názvy elementů a komplexní typy. Tato pravidla pomáhají zvýšit srozumitelnost GML a jeho schémat.

- Objekty jsou elementy XML s významovým názvem - UpperCamelCase.
- Vlastnosti jsou elementy XML jejichž názvy jsou psány - lowerCamelCase.

- Abstract elementy mají předponu „Abstract“ (pokud se jedná o objekty) nebo „abstract“ (pokud jde o vlastnosti) představenou před jejich názvem.
- Názvy komplexních typů schémat XML jsou psány UpperCamelCase končící slovem „Type“.
- Abstract schémata XML mají představeno slovo „Abstract“.

Důrazně se doporučuje dodržovat tato pravidla také v GML aplikačních schématech. Pravidla jsou aplikovatelná pouze v jazycích, které rozlišují mezi uppercase a lowercase.

Základní komponenty GML schématu

Cílem základních komponent je zavedení GML modelu a syntaxe. Jedná se o:

- kořenový XML typ, ze kterého jsou odvozeny GML modely a syntax,
- vzor a součásti vlastností GML,
- vzor pro kolekce a matice,
- komponenty pro přidružování metadat k objektům GML,
- komponenty pro tvorbu a definování slovníků.

Základní objekty

AbstractObject

gml:AbstractObject element nemá definovaný žádný typ, a proto je implicitně *anyType*. Používá se jako hlavička XML schématu, která unifikuje obsah a jednoduchý obsah elementů pro datové typy v GML.

AbstractGML, AbstractGMLType

Tento element *gml:AbstractGML* značí jakýkoli GML objekt mající identitu. Chová se jako hlavička skupiny schématu XML, která může obsahovat jakýkoli element, který je prvek GML nebo objekt s identitou. Používá se jako proměnná v obsahových modelech v jádru GML a aplikačních schématech.

Vlastnosti GML

Termín „vlastnost“ je používán pro odkazování na vlastnost GML, která je charakteristikou nějakého GML objektu. Element v GML dokumentu nebo datovém toku je vlastnost GML jestliže a pouze jestliže je to podřízený element objektu GML. Význam každé vlastnosti by měl být naznačen v názvu elementu.

Objekty GML mohou mít nekonečný počet vlastností, navíc ještě ty, které podědí z *gml:AbstractGMLType*. Vlastnost může být definována tak, že má buď jednoduchý, nebo komplexní obsah. Vlastnost s jednoduchým obsahem má XML schéma jednoduché, jako například tyto elementy standardních vlastností: *gml:description* a *gml:name*. Vlastnost s komplexním obsahem má XML schéma komplexní. Vlastnost elementu může mít udána dvěma způsoby:

- Vložením hodnoty vlastnosti přímo. Kupříkladu *gml:name* či *gml:description*.
- Odkazem, kdy hodnoty vlastnosti jsou dostupné jinde a jsou identifikovány atributem *xlink:href*. Tato metoda by měla být používána standardně.

AssociationAttributeGroup

XLink elementy jsou standardní metodou, jak hypertextově odkazovat na XML. Tento element, *gml:AssociationAttributeGroup*, podporuje používání Xlink jako metodu pro označování hodnoty vlastnosti jako odkaz.

abstractAssociationRole, AssociationRoleType

Aby bylo možné podporovat kódování vlastností, které mají komplexní obsah, používají se tyto elementy.

Standardní vlastnosti GML objektů

gml:AbstractGMLType - všechny GML objekty obsažené v modelu dědí vlastnosti tohoto elementu.

gml:description - textová hodnota popisující objekt.

gml:descriptionReference - hodnota této proměnné je vzdálený text popisující objekt provázaný přes *xlink:href*.

gml:name - udává identifikátor nebo označení objektu, většinou popisný název.

gml:id - udává hodnotu, se kterou XML dále operuje. Je povinná pro všechny GML objekty.

4.1.2 GML schéma – Xlinky

Xlinky jsou v GML používány k implementaci vazeb mezi objekty formou odkazů. Jak už bylo řečeno, vlastnosti elementů GML mohou nést atributy Xlink, které podporují kódování vazeb a vztahů. Nejdůležitějším elementem je *xlink:href*, který odkazuje na cíl přes URI. Další Xlink elementy se používají k vyjádření jiných vazeb. Nejužitečnějšími jsou:

xlink:role - popis povahy cíle, zadáno jako URI.

xlink:arcrole - popis funkce či účelu cíle, zadáno jako URI.

xlink:title - popis spojení cílového zdroje, zadáno jako text.

4.1.3 GML schéma – Prvky

GML prvky jsou prvky (cesta, řeka, osoba, vozidlo, region, ...) kódované použitím GML.

gml:AbstractFeatureType - základní model prvku, jemuž lze mimo jiné zadat vlastnosti obalová zóna (envelope) a rozsah.

gml:AbstractFeature - lze chápat jako cokoli, co je GML prvek. Používá se k definování proměnných či šablon.

GML má předdefinovanou širokou škálu typů prvků, topologických typů, časových omezení

4.1.4 GML schéma – Základní geometrie

GML má pro zadávání geometrických tvarů vytvořenou spoustu elementů, čili přístupů, jakým způsobem geometrii zadávat. Nejedná se pouze ryze geografické metody, ale také grafické způsoby.

Zadávat lze geometrie abstraktní, bezrozměrné, jednorozměrné, dvourozměrné a trojrozměrné. Stejně tak lze definovat geometrické kombinace, směsi různých typů či souhrny.

```
<gml:Polygon>
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:posList>0,0 100,0 100,100 0,100 0,0</gml:posList>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>
<gml:Point>
  <gml:posList>100,200</gml:posList>
</gml:Point>
<gml:LineString>
  <gml:posList>100,200 150,300</gml:posList>
</gml:LineString>
```

Obr. 16: Ukázka zápisu bodu, linie a polygonu v GML ^[12]

4.1.5 GML schéma – Souřadnicové referenční systémy

Toto schéma kóduje nejen definice souřadných systémů, ale také jejich transformace a konverze mezi nimi. Lze zadat informace o geodetické poloze, vertikální poloze, dočasné poloze, technické poloze nebo poloze obrázku. Lze definovat kupříkladu i projekční plochy, elipsoidy, nulové poledníky a spoustu dalších vlastností.

4.1.6 GML schéma – Topologie

Existují definovatelné čtyři třídy. Pro základní topologické objekty, a pak pro každou dimenzi po 3D. Podporovány jsou také různé topologické soubory.

4.1.7 GML schéma – Dočasné informace a dynamické prvky

Schémata dočasných prvků obsahují komponenty popisující dočasnou geometrii, topologii, souřadnicové systémy a dočasné charakteristiky geografických dat. Čas je měřen ve dvou měřítkách: intervalové měřítko nabízí základy pro měření doby, ordinální měřítko poskytuje informace pouze o relativní poloze v čase.

4.1.8 GML schéma – Definice a slovníky

Často je vhodné použít definice poskytované externími orgány. Ty mohou být distribuovány různými způsoby – online i offline. Tyto komponenty mohou být používány přímo nebo také jako základ další specializované definice prvků v GML. Jedná se třeba o souřadnicové operace, souřadné systémy, datumy, dočasné referenční systémy nebo měrné jednotky. Stejně tak mohou být přidány slovníky, které obsahují nějakou sadu definic.

4.1.9 GML schéma – Jednotky, míry a hodnoty

Několik GML schémat se také týká požadavků na kvantitativní hodnoty, které používají nějaké měřítko či jednotky míry.

4.1.10 GML schéma – Směry

Toto schéma popisuje směr. Asociované objekty mohou tak získat vlastnosti definující jejich orientaci, směr, kurz, azimut nebo další směrové aspekty geografických prvků.

4.1.11 GML schéma – Pozorování

Tyto modely se zabývají procesem pozorování, často pomocí nějakého záznamového zařízení nebo přímo člověkem. Prvek pozorování popisuje metadata, která jsou sdružována společně s procesem zachycení záznamu. Toto zahrnuje širokou škálu případů, včetně turistických fotografií (ne však samotné fotografie, ale proces tvorby fotografie). Tohoto schématu se většinou užívá pro jednoduchá pozorování. Pro vědecké či technické účely a měření je většinou vyžadováno specializovaného GML aplikačního schématu.

GML Aplikační schéma v oblasti zpracování senzorových dat

GML poskytuje základ pro aplikační schémata specifická pro různé komunity nebo domény, která podporují interoperabilitu dat. Aplikační schémata jsou obvykle navržena podle normy ISO 19103 konformní UML. Poté jsou GML aplikace vytvářeny na základě pravidel uvedených v normě ISO 19136 ^[13].

Existuje implementace GML aplikačního schématu pro standard Observation & Measurements (OMXML). Ukázka kódu OMXML dokumentu pro pozorování jehož výsledkem je míra:

```
<om:OM_Observation
  gml:id="obsTest1"
  xmlns:om="http://www.opengis.net/om/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xsi:schemaLocation="http://www.opengis.net/om/2.0
http://schemas.opengis.net/om/2.0/observation.xsd">

  <gml:description>Observation test instance: fruit mass</gml:description>
  <gml:name>Observation test 1</gml:name>
  <om:type
    xlink:href="http://www.opengis.net/def/observationType/OGC-OM/2.0/OM_Measurement"/>
  <om:phenomenonTime>
    <gml:TimeInstant
      gml:id="ot1t">
        <gml:timePosition>2005-01-11T16:22:25.00</gml:timePosition>
      </gml:TimeInstant>
    </om:phenomenonTime>
    <om:resultTime xlink:href="#ot1t"/>
    <!-- URL identifikující proceduru... -->
    <om:procedure
      xlink:href="http://www.example.org/register/process/scales34.xml"/>
    <!-- enviromentální podmínky během měření-->
    <om:parameter>
      <om:NamedValue>
        <om:name
          xlink:href="http://sweet.jpl.nasa.gov/ontology/property.owl#Temperature"/>
        <om:value xsi:type="gml:MeasureType" uom="Cel">22.3</om:value>
      </om:NamedValue>
    </om:parameter>
    <!-- URN identifikující pozorovanou vlastnost-->
    <om:observedProperty
      xlink:href="http://sweet.jpl.nasa.gov/2.0/phys.owl#Mass"/>
    <!-- WFS identifikuje objekt závislý na provedeném pozorování -->
    <om:featureOfInterest
      xlink:href="http://wfs.example.org?request=getFeature&featureid=fruit37f"/>

    <om:result
      xsi:type="gml:MeasureType"
      uom="kg">0.28</om:result>
    <!-- XML Schema typ výsledku je označen jako hodnota xsi:type atributu -->

  </om:OM_Observation>
```

Pozn.: Následující odrážky vysvětlují barevné rozlišení kódu XML dokumentu, které platí i pro níže uvedené části kódů.

- XML elementy
- vlastnosti XML elementů (typ výsledku, název), xlinky
- hodnoty vlastností XML elementů
- hodnoty XML elementů
- odkazy na použité standardy a definice
- poznámky

4.1.12 GML schéma - Coverage

GML také definuje schémata pro tvorbu coverage a reprezentací.

Pozn.: Detailní popis je dostupný z <http://www.opengeospatial.org/standards/gml>

4.2 Simple Features

Simple Features jsou OpenGIS standard (verze 1.2.0), který specifikuje digitální ukládání geografických dat (bod, linie, polygon, multi-bod, multi-linie, atd.), jak s prostorovými, tak i s neprostorovými atributy^[18].

Simple Features jako ISO 19125 se skládá ze dvou částí s názvem ISO 19125-1 (Simple feature access - Part 1: Common architecture) a ISO 19125-2 (Simple feature access - Part 2: SQL option). První část standardu definuje simple features model. Jedná se o abstraktní model v tom smyslu, že je nezávislý na konkrétní platformě. Část druhá definuje simple features přístup k databázi pomocí rozhraní SQL. OGC navrhla základní dokumenty ISO 19125^[1]. Simple Features jsou založeny na 2D geometrii s lineární interpolací mezi vrcholy. Obecně platí, že 2D geometrie je jednoduchá, neobsahuje-li překřížení prvku sebe samým. OpenGIS Simple Features specifikace definuje různé prostorové subjekty, které mohou být použity k vytvoření nové geometrie z existující geometrie. Plné znění standardu je dostupné na <http://www.opengeospatial.org/standards/sfa>.

Definuje dva formáty pro popis geografických dat (WKT a WKB).

4.2.1 Well-known text (WKT)

WKT je textový značkový jazyk určený pro popis vektorové geometrie geografických objektů, prostorových referenčních systémů, popřípadě transformačních parametrů mezi jednotlivými souřadnicovými systémy.

1) *geometrické objekty*

Geometrické objekty, které lze pomocí WKT popsat jsou:

- body,
- linie,
- polygony,
- TIN,
- polyhedrony.

Multigeometrické typy dovolují použít v jednom objektu více geometrických typů stejné dimenze. Pro uložení geometrie různých dimenzí slouží kolekce geometrie.

Souřadnice mohou být 2D (x, y), 3D (x, y, z), 4D (x, y, z, m), kde m je hodnota lineárního referenčního systému nebo 2D s hodnotou m (x, y, m). Klíčové slovo EMPTY definuje tzv. „prázdnou“ geometrii. Prvek, který neobsahuje žádné lomové body (souřadnice).

Zápis ve formě WKT je často používán v OGC specifikacích, PostGIS například obsahuje funkce pro konverzi geometrie z/do formy WKT.

Příklad geometrie zapsané ve formě WKT:

```
POINT(6 10)
LINESTRING(3 4,10 50,20 25)
POLYGON((1 1,5 1,5 5,1 5,1 1),(2 2, 3 2, 3 3, 2 3,2 2))
MULTIPOINT(3.5 5.6,4.8 10.5)
MULTILINESTRING((3 4,10 50,20 25),(-5 -8,-10 -8,-15 -4))
MULTIPOLYGON(((1 1,5 1,5 5,1 5,1 1),(2 2, 3 2, 3 3, 2 3,2 2)),((3 3,6 2,6 4,3 3)))
GEOMETRYCOLLECTION(POINT(4 6),LINESTRING(4 6,7 10))
```

2) *prostorové referenční systémy*

Prostorový referenční systém obsahuje v zápisu WKT informace o geodetickém datu, geoidu, souřadnicovém systému a kartografickém zobrazení. WKT je používán řadou GIS aplikací a knihoven, např. ESRI používá WKT ve formátu Shapefile (soubor *.prj). Příklad definice systému S-JTSK ve formě WKT:

```
PROJCS["krovak",
  GEOGCS["bessel",
    DATUM["unknown",
      SPHEROID["Bessel_1841",6377397.155,299.1528128],
      TOWGS84[570.8,85.7,462.8,4.998,1.587,5.261,3.56]],
    PRIMEM["Greenwich",0],
    UNIT["degree",0.0174532925199433]],
  PROJECTION["Krovak"],
  PARAMETER["latitude_of_center",0],
  PARAMETER["longitude_of_center",0],
```

```

PARAMETER["azimuth",0],
PARAMETER["pseudo_standard_parallel_1",0],
PARAMETER["scale_factor",1],
PARAMETER["false_easting",0],
PARAMETER["false_northing",0],
UNIT["meter",1]]

```

3) transformace

Formát WKT může popisovat metodu transformace a transformační parametry pro konverzi mezi dvěma prostorovými souřadnicovými systémy.

Databáze podporující WKT: MySQL, PostgreSQL s PostGIS, SQLite se SpatiaLite, Oracle, MS SQL Server.

4.2.2 Well-known binary (WKB)

WKB je binární forma používaná pro převod a uložení informací v databázovém systému jako je např. PostGIS. Formát je kontrolován konsorciem Open Geospatial Consortium (OGC) a popsán ve specifikacích *Simple Feature Access* a *Coordinate Transformation Service*. Všechny souřadnice se zde ukládají jako hodnoty Double precision (64bitové číslo s plovoucí desetinnou čárkou). Příklad geometrie zapsané ve formě WKB:

```
010100000000000000000000F03F000000000000F03F
```

Hodnota odpovídá hodnotě POINT(1 1) ve formátu WKT.

Pozn.: Rastrová data se ukládají jako **Binary large object (BLOB)**, je označení pro datový typ blíže nespecifikovaných binárních dat v databázi. Obvykle se jedná o obrázky, zvukové záznamy nebo jakákoli jiná data, která jinak bývají uložena v samostatných souborech ^[18]. BLOB je tedy obecný datový typ, který explicitně vyjadřuje, že databáze nemá informace o tom, jak jeho obsah interpretovat. Na rozdíl od jiných datových typů, kdy databáze může číselné a časové údaje poskytovat ve vhodném formátu, případně textové údaje konvertovat do požadované kódové stránky, obsah BLOB položky obdrží klientská aplikace přesně tak, jak byl binárně uložen a o interpretaci se musí postarat sama.

4.3 Filter Encoding

Mnoho OpenGIS webových služeb vyžaduje schopnost podporovat XML založené na filtrování. Tato specifikace definuje Filter Encoding jako mechanismus, který mohou využít webové služby pro výběr podmnožiny instance objektů. Filter Encoding je běžnou součástí, kterou lze použít v mnoha webových službách, jako jsou Web Feature Service

(WFS), Web Coverage Service (WCS), Styled Layer Descriptor. Například WFS může použít Filter Encoding v GetFeature operaci pro specifikaci omezení dotazu ^[5].

Dotazovací jazyk používaný ve Filter Encoding je založen na Common Query Language (CQL) definovaném OGC Catalog Interface Implementation Specification. Common Query Language (CQL) je formální jazyk pro reprezentaci dotazů na informační systémy, jako jsou webové indexy, bibliografické katalogy a muzejní sbírky informací. Je to dotazovací jazyk s minimální syntaxí podobnou SQL WHERE klauzuli v SELECT dotazu ^[20]. CQL je určen k „do what you mean (dělej, co myslíš)“, pro jednoduché každodenní dotazy a zároveň umožňuje prostředky k vyjádření složitých pojmů ^[6]. Tato specifikace definuje běžně užívané XML kódování OGC CQL jako systém neutrální reprezentace predikátu dotazu. Dotaz uvedený v této reprezentaci XML lze jednoduše validovat, analyzovat, a pak převést do cílového jazyka bez ohledu na povinnost provést opatření proti místnímu trvalému objektu skladu. Například XML kódovaný filtr může být transformován do klauzule WHERE v SQL příkazu SELECT, v případě, že objekty jsou uloženy v relační databázi založené na SQL. Může také být přeměněn na XPath a XPointer výraz, v případě, že objekty jsou uloženy v XML dokumentech.

Následující předpoklady o objektech a jejich vlastnostech jsou používány OGC Filter Encoding:

1. Obecné objekty jsou složeny z jednoduchých a/nebo komplexních nebo souhrnných negeometrických vlastností.
2. Objekty jsou zakódovány v XML.
3. Vlastnosti objektu mohou být mapovány buď XML elementy nebo atributy XML.
4. Má-li objekt komplexní nebo souhrnné negeometrické vlastnosti, tyto vlastnosti mohou být mapovány vnořenými soubory XML elementů.

<PropertyName> element se nepoužívá k zakódování názvů vlastností objektu ve filtrovacím výrazu. Názvy vlastností musí být platné XML elementy nebo názvy atributů, jak je definováno v XML. Názvy vlastností mohou být kvalifikovány s prefixem jmenného prostoru, stejně jako v případě, kdy název musí být potvrzen v Namespace ve specifikaci XML. Jednoduché vlastnosti mohou být odkazovány pomocí jejich jména. Nicméně, protože objekty také obsahují složité nebo souhrnné negeometrické vlastnosti, mechanismus pro odkazování na tyto typy vlastností musí být podpořen Filter Encoding. Podmnožina výrazů XPath se používá pro tento účel k podpoře zkrácení relativní cesty umístění. Filter Encoding nemusí podporovat plný XPath jazyk, aby vstupní náklady byly co nejnižší ^[5].

4.3.1 Filtry a typy filtrů

Typy filtrů mohou být následující:

- Prostorové operátory
- Srovnávací operátory
- Logické operátory
- Identifikátory prvku

Následující typy filtrů jsou definovány OGC:

- Aritmetické operátory
- Literals
- Výrazy (Expressions)
- Funkce (Functions)

Prostorové operátory (Spatial Operators)

Prostorový operátor určuje, zda jeho geometrické argumenty splňují uvedené prostorové vztahy. Operátory zahrnují:

- Equals (ogc:BinarySpatialOpType) – geometrický objekt je rovný jinému objektu
`<xsd:element name="Equals" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Disjoint (ogc:BinarySpatialOpType) – geometrický objekt je disjunkt k jinému objektu
`<xsd:element name="Disjoint" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Touches (ogc:BinarySpatialOpType) – geometrický objekt se dotýká jiného objektu
`<xsd:element name="Touches" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Within (ogc:BinarySpatialOpType) – geometrický objekt je uvnitř jiného objektu
`<xsd:element name="Within" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Overlaps (ogc:BinarySpatialOpType) – geometrický objekt překrývá jiný objekt
`<xsd:element name="Overlaps" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Crosses (ogc:BinarySpatialOpType) – geometrický objekt se křížuje s jiným objektem
`<xsd:element name="Crosses" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Intersects (ogc:BinarySpatialOpType) – geometrický objekt průnik s jiným objektem
`<xsd:element name="Intersects" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Contains (ogc:BinarySpatialOpType) – geometrický objekt obsahuje jiný objekt
`<xsd:element name="Contains" type="fes:BinarySpatialOpType" substitutionGroup="fes:spatialOps"/>`
- Dwithin (ogc:DistanceBufferType) – geometrický objekt je v dané vzdálenosti
`<xsd:element name="DWithin" type="fes:DistanceBufferType" substitutionGroup="fes:spatialOps"/>`
- Beyond (ogc:DistanceBufferType) – geometrický objekt je za danou vzdáleností
`<xsd:element name="Beyond" type="fes:DistanceBufferType" substitutionGroup="fes:spatialOps"/>`
- BBOX (ogc:BBOXType) – geometrický objekt je v daném ohraničujícím rámečku
`<xsd:element name="BBOX" type="fes:BBOXType" substitutionGroup="fes:spatialOps"/>`

Pozn.: Kódy popsané u jednotlivých prostorových operátorů jsou části kódu souboru filter.xsd, který patří mezi tři soubory definující OGC Filter Encoding schéma. Kromě souboru filter.xsd je OGC Filter Encoding schéma definováno soubory filterCapabilities.xsd a expr.xsd – viz 4.3.3.

```
<fes:Filter
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/fes/2.0
http://schemas.opengis.net/filter/2.0.0/filterAll.xsd
http://www.opengis.net/gml/3.2
http://schemas.opengis.net/gml/3.2.1/gml.xsd">
  <fes:BBOX>
    <fes:ValueReference>Geometry</fes:ValueReference>
    <gml:Envelope srsName="urn:ogc:def:crs:EPSG::4326">
      <gml:lowerCorner>13.0983 31.5899</gml:lowerCorner>
      <gml:upperCorner>35.5472 42.8143</gml:upperCorner>
    </gml:Envelope>
  </fes:BBOX>
</fes:Filter>
```

Tento zápis kódu ukazuje jednoduchý filtr s prostorovým operátorem BBOX.

Srovnávací operátory (Comparison Operators)

Srovnávací operátor se používá k vyhodnocení matematického srovnání mezi dvěma argumenty. Operátory zahrnují:

- PropertyIsEqualTo (ogc:BinaryComparisonOpType) – vlastnost je rovna hodnotě
- PropertyIsNotEqualTo (ogc:BinaryComparisonOpType) – vlastnost není rovna hodnotě
- PropertyIsLessThan (ogc:BinaryComparisonOpType) – vlastnost je menší než hodnota
- PropertyIsGreaterThan (ogc:BinaryComparisonOpType) – vlastnost je větší než hodnota
- PropertyIsLessThanOrEqual (ogc:BinaryComparisonOpType) – vlastnost je menší nebo rovna hodnotě
- PropertyIsGreaterThanOrEqual (ogc:BinaryComparisonOpType) – vlastnost je větší nebo rovna hodnotě
- PropertyIsLike (ogc:PropertyIsLikeType) – vlastnost je porovnána s hodnotou
- PropertyIsNull (ogc:PropertyIsNullType) – vlastnost je nulová
- PropertyIsBetween (ogc:PropertyIsBetweenType) – vlastnost je mezi hodnotami

Logické operátory (Logical Operators)

Logické operátory se používají ke kombinaci jednoho nebo více podmíněných výrazů. Operátory zahrnují:

- AND (ogc:BinaryLogicalOpType)

```
<xsd:element name="And" type="fes:BinaryLogicOpType" substitutionGroup="fes:logicOps"/>
```

- OR (ogc:BinaryLogicalOpType)

```
<xsd:element name="Or" type="fes:BinaryLogicOpType" substitutionGroup="fes:logicOps"/>
```

- NOT (ogc:BinaryLogicalOpType)

```
<xsd:element name="Not" type="fes:BinaryLogicOpType" substitutionGroup="fes:logicOps"/>
```

Pozn.: Kódy popsané u jednotlivých logických operátorů jsou části kódu souboru filter.xsd, který patří mezi tři soubory definující OGC Filter Encoding schéma.

Identifikátory prvku (Feature Identifiers)

<FeatureID> element (ogc:FeatureIDType) je používán k reprezentaci jedinečného identifikátoru geografického prvku podaného webovou službou.

Aritmetické operátory (Arithmetic Operators)

Následující jsou aritmetické operátory:

- Add (ogc:BinaryOperatorType) – sčítání
- Sub (ogc:BinaryOperatorType) – odčítání
- Mul (ogc:BinaryOperatorType) – násobení
- Div (ogc:BinaryOperatorType) – dělení

Literals

<Literal> element (ogc:LiteralType) je používán k reprezentaci literal hodnot. GML musí být používán k reprezentaci literal geometrických hodnot.

Výrazy (Expressions)

Podle specifikace OGC, je výraz tvořen pomocí následujících prvků:

- Aritmetické operátory: <Add>, <Sub>, <Mul>, <Div>
- <PropertyName>
- <Literal>
- <Function>

Výrazy jsou XML fragmenty vzniklé vnořením prvků.

Funkce (Functions)

<Function> element (ogc:FunctionType) je používán k definování jednotlivých hodnot funkcí. Funkce mohou akceptovat žádný nebo více argumentů jako vstup a generovat výsledek. Vstupní argumenty jsou výrazy (expressions).

4.3.2 Možnosti filtru (Filter Capabilities)

Webová služba, která podporuje Filter Encoding by měla zveřejnit možnosti, které podporuje. Možnosti filtru jsou rozděleny do dvou kategorií: prostorové možnosti (filter capabilities) a skalární možnosti (scalar capabilities). Prostorové možnosti zahrnují všechny prostorové operátory, zatímco skalární možnosti odkazují na neprostorové operátory: logické, srovnávací, aritmetické a funkce ^[5].

4.3.3 Filter Encoding schéma

OGC Filter Encoding schéma je definováno ve třech souborech: "expr.xsd", "filter.xsd" a "filterCapabilities.xsd". Schéma "filter.xsd" zahrnuje "expr.xsd", a nemusí být importováno do schématu, které používá OGC filter encoding. Schéma "FilterCapabilities.xsd" musí být importováno do možností schématu služeb, které používají filter encoding, k inzerování schopností filter encoding těchto služeb ^[5].

5 NÁSTROJE PRO SPRÁVU DAT ZE SENZOROVÉ SÍTĚ

Bezdrátové senzorové sítě (WSNs) jsou rozvíjející se oblastí výzkumu. Aplikace pro senzorové sítě, manipulaci s daty a jejich reprezentaci se staly zásadní součástí senzorových sítí. Vizualizace je klíčovou otázkou pro vývoj a provoz těchto sítí. Údaje vydávané jednotlivými senzorovými uzly jsou shromažďovány bránou a předány základní stanici. Dále pak mohou být předány prostřednictvím připojení k internetu na vizualizační software na vzdáleném počítači ^[32].

5.1 Testované aplikace

5.1.1 Webový prohlížeč měřených dat Fiedler-Mágr

Webový prohlížeč umožňuje přístup k datům, která posílají telemetrické stanice typu M4016, M4316 a STELA společnosti Fiedler-Mágr vybavené GPRS modemem, na server. Zde jsou tato data archivována v databázi, ke které má uživatel přístup právě prostřednictvím webového prohlížeče. Ten tato data umí zobrazit v uživatelsky přívětivé podobě díky grafům a tabulkám. Podporuje též exporty do několika datových formátů. Další funkce webového prohlížeče závisejí na výši přiděleného přístupového práva uživatele ^[10].

Pro přístup na server webového prohlížeče postačí jakýkoliv počítač připojený k internetu a vybavený internetovým prohlížečem (např. Mozilla Firefox, Google Chrome, atd.). Webová adresa serveru je <https://stanice.fiedler-magr.cz>. Další možnosti pro přístup na server je přes webové stránky firmy Fiedler-Mágr, odkazem Datahosting.

Přihlášení do webového prohlížeče

Přihlášení do prohlížeče se provádí klasickým způsobem, tedy zadáním uživatelského jména a hesla, které bylo uživateli předáno zástupcem firmy Fiedler-Mágr. Každý uživatel prohlížeče má stanovená různá přístupová práva. Nejnižší práva má uživatel, jenž zadává jako uživatelské jméno a heslo stejný údaj „demo“. Takto slouží webový prohlížeč pouze k prezentaci prohlížeče. Uživatel s maximálními právy má větší volnost v přizpůsobování vzhledu prohlížeče a parametrizaci stanic.

Hlavní stránka

Po úspěšném přihlášení se uživateli zobrazí hlavní stránka. Tato stránka je rozdělena na tři části. První částí je tzv. informační pruh (nahore), kde je zobrazeno jméno uživatele, datum a čas, časové pásmo a k dispozici jsou odkazy pro změnu hesla, přepnutí jazyka

a odhlášení uživatele. Druhá část hlavní stránky (vlevo) zobrazuje seznam stanic řazených podle oblastí. Poslední část (vpravo) je hlavní zobrazovací oblast, která po přihlášení zobrazuje aktuality a rady pro uživatele. Hlavní funkcí této části je pak zobrazení detailů stanice vybrané ze seznamu stanic, zobrazení významných hodnot změřených na kanálech a dalších nástrojů, které umožňují práci s daty.

Web prohlížeč měřených dat
Přihlášený uživatel: Brus
Časové pásmo: Europe/Prague
24.01.2012 22:41:56

Popis ovládání Webového prohlížeče
web LOG
Změnit heslo
English
Odhlásit se

tzv. informační pruh

Měřicí stanice: M4016_19800
UPOZORNĚNÍ: Veškerá uváděná data jsou bez právní záruky.
Poslední přenos: Ne 25.12 07:38:31
Poslední data: Ne 25.12 07:38:25

Měřicí kanály:

☒	Jmenovka	Posl. změřená hodnota	24.12.11			Suma	Suma od instalace
			Minima	Maxima	Průměr		
☐	K1: Vlhkost 2m [%]	85,11	83,59	95,99	93,02	-	-
☒	K2: Teplota 2m [°C]	2,3	0,9	3,8	2,1	-	-
☐	K3: VlhkostPůdy1 [%]	31,4	29,7	31,2	30,3	-	-
☐	K13: Teplota AV1 [°C]	0,8	0,5	0,8	0,6	-	-
☐	K14: Teplota AV2 [°C]	2,4	2,4	2,5	2,4	-	-
☐	K15: Teplota AV3 [°C]	4,2	4,1	4,2	4,1	-	-
☐	K16: Napětí AKU [V]	12,91	12,69	12,91	12,90	-	-
☐	Tabulka událostí						

Grafy
Grafy Export do DTA Export do CSV archivovaný měsíc čárový graf
měsíc: 1 rok: 2012

Statistika:
Tabulky Export do DTA Export do CSV posledních 31 dní

Obr. 17: Webový prohlížeč Fiedler-Mágr

Práce s daty

Ze seznamu stanic je potřeba si vybrat tu, se kterou se bude pracovat. Po kliknutí se v hlavní zobrazovací části objeví název stanice a její poslední aktivita. Dále pak je zobrazena ikona stavu stanice (např. nízké napětí akumulátoru nebo alarm na některém měřicím kanálu), nástrojová lišta, ale hlavně jsou zde zobrazeny důležité prvky, se kterými lze dále pracovat. Jsou to tabulka měřicích kanálů, grafy, statistika a tabulka binárních kanálů.

V tabulce měřicích kanálů lze označit ty kanály, které budou zahrnuty do dalších operací s daty. Poslední položkou v této tabulce je tabulka událostí. Po zaškrtnutí této položky a kliknutí na tlačítko *Grafy* se zobrazí tabulka, která zobrazuje všechny události na stanici za zvolený interval. Kromě označovacího políčka obsahují jednotlivé položky

měřících kanálů také jmenovku, poslední změřenou hodnotu a hodnoty za předchozí den (min, max, průměr, suma).

Po předchozím označení kanálu z tabulky měřících kanálů lze přistoupit k dalšímu prvku, kterým jsou grafy. Tento prvek obsahuje tři tlačítka a dvě možnosti výběru (období a typ grafu). Možnosti výběru období jsou: archivovaný měsíc, včera, dnes, týden zpět, posledních 24h dat, posledních 48 hodin dat, posledních 31 dní dat, interval od..do. Možnosti výběru typu grafu jsou: čárový, bodový, vyplněný, sloupcový. Tlačítkem *Grafy* lze zobrazit stránku grafů měřících kanálů tak, jak si uživatel nastaví jeho možnosti. Tato stránka se otevře v novém okně. Další dvě tlačítka umožňují export grafu do datových formátů DTA a CSV.

Dalším prvkem je statistika, která podobně jako grafy obsahuje tři tlačítka, ale jen jednu možnost výběru (období). Možnosti výběru období jsou zde odlišné od výběru období u grafů. Jsou to: archivovaný měsíc, minulý měsíc, aktuální měsíc, posledních 31 dní, interval od..do. Tlačítka jsou jako u předchozího, tedy pro zobrazení stránky statistiky je zde tlačítko *Tabulky*, které otevře nové okno se statistikou a pro export jsou zde tlačítka *Export do DTA* a *Export do CSV*. Posledním zobrazeným prvkem je pak tabulka binárních kanálů, která je velmi podobná tabulce měřících kanálů.

Nastavení

Výběrem stanice ze seznamu se v hlavní zobrazovací části zobrazuje nástrojová lišta, včetně dalších prvků, které byly výše popsány. Tato nástrojová lišta obsahuje nástroje: *Nastavení zobrazení hlavní stránky*, *Nastavení zobrazení stránky grafů*, *Nastavení zobrazení stránky statistik*, *Nakonfigurovat vybrané parametry*, *Finance stanice*, *Operace s daty*, *Sledovací agent* a *Parametry stanice*.

Nastavení zobrazení hlavní stránky umožňuje nastavit počet a typ grafů zobrazených v hlavní zobrazovací oblasti. Kromě grafů lze zobrazit i fotografii měřící stanice. Celkem je k dispozici osm oblastí pro zobrazení. V případě používání předplacené SIM karty, webový prohlížeč měřených dat obsahuje historii výše kreditu, kterou lze získat pomocí nástroje *Finance stanice*. K nástroji *Operace s daty* mají přístup pouze uživatelé s nejvyššími právy. Slouží k manuálnímu nastavení přenosu dat a k mazání dat z databáze.

Nástroj *Sledovací agent* umí při každém připojení stanice k webovému serveru vyhodnotit předem nastavené podmínky, a na základě těchto podmínek rozesílá emailové zprávy. Ideální použití je pro kontrolu výše a platnosti kreditu SIM karty ve stanici, nízkého napětí akumulátoru na měřících stanicích bez síťového napájení, či chyba v přenosu dat. Možností je k dispozici ale mnohem víc: sledování spouštění a vypnutí alarmů, výpadek síťového napájení, upozorňování na chyby na všech nebo pouze vybraném měřícím kanále, překročení stanovených mezí na měřících kanálech, či monitorování binárních vstupů i výstupů ^[10].

Přenos dat z webového prohlížeče do vlastní databáze

Aby bylo možné transportovat data z webového prohlížeče do vlastní databáze, je potřeba nejdříve exportovat data do souboru *.csv. Soubor vyexportovaný z webového prohlížeče nelze použít přímo, a proto je nutné jej upravit nebo kopírováním dat z tohoto souboru vytvořit nový *.csv soubor. V novém souboru musí být hlavička s názvy sloupců a jednotlivé hodnoty sloupců. Ostatní metadata mohou být smazána. Takto upravený soubor lze vložit do databáze.

Před prvním přesunem dat z *.csv souboru je potřeba vytvořit vlastní databázi. Pro tento účel byla zvolena databáze PostgreSQL. Existuje několik možností, jak vytvořit databázi v PostgreSQL. Jednou z nich je použití SQL jazyka. V tomto případě se vytvořená databáze bude jmenovat *FIEDLER* a její kódování bude UTF-8. Ukázka kódu vytvoření databáze pro transportovaná data:

```
sql1: CREATE DATABASE "FIEDLER"
      WITH OWNER = postgres
      ENCODING = 'UTF8'
      TABLESPACE = pg_default
      CONNECTION LIMIT = -1;
```

Pozn.: Pokud je kódování databáze nastaveno na UTF-8, pak stejné kódování musí mít i upravený *.csv soubor.

Dalším krokem je vytvoření tabulky, do které budou vkládány hodnoty sloupců z *.csv souboru. Jednotlivé sloupce vytvořené tabulky budou mít stejné názvy tak, jak jsou uvedeny v hlavičce v souboru. Těmto sloupcům tabulky je ještě potřeba přiřadit datový typ. Ukázka zápisu kódu vytvoření tabulky:

```
sql2: CREATE TABLE "Srazkomer_K1"(
      "Datum" character varying,
      "Čas minima" time without time zone,
      "Hodnota minima" character varying,
      "Čas maxima" time without time zone,
      "Hodnota maxima" character varying,
      "Průměr" character varying,
      "Suma denní" character varying,
      "Suma nasčítaná" character varying
)
WITH (OIDS=FALSE);
```

Konečně je možné importovat data z *.csv souboru do vytvořené vlastní databáze. To se provádí pomocí SQL příkazu COPY. Tento příkaz obsahuje název tabulky, do které jsou data vkládána, názvy jednotlivých sloupců tabulky, cestu k souboru a oddělovač, kterým jsou data v souboru oddělena. Dále příkaz specifikuje typ souboru, ze kterého jsou data kopírována do tabulky (tedy *.csv) a specifikuje, že soubor obsahuje hlavičku s názvy jednotlivých sloupců. Ukázka zápisu kódu pro příkaz COPY:

```
sql3: COPY "Srazkomer_K1"("Datum","Čas minima","Hodnota minima","Čas  
maxima","Hodnota maxima","Průměr","Suma denní","Suma nasčítaná")  
FROM '/SrazkomerK1Srazky.csv'  
WITH DELIMITER ';' '  
CSV HEADER;
```

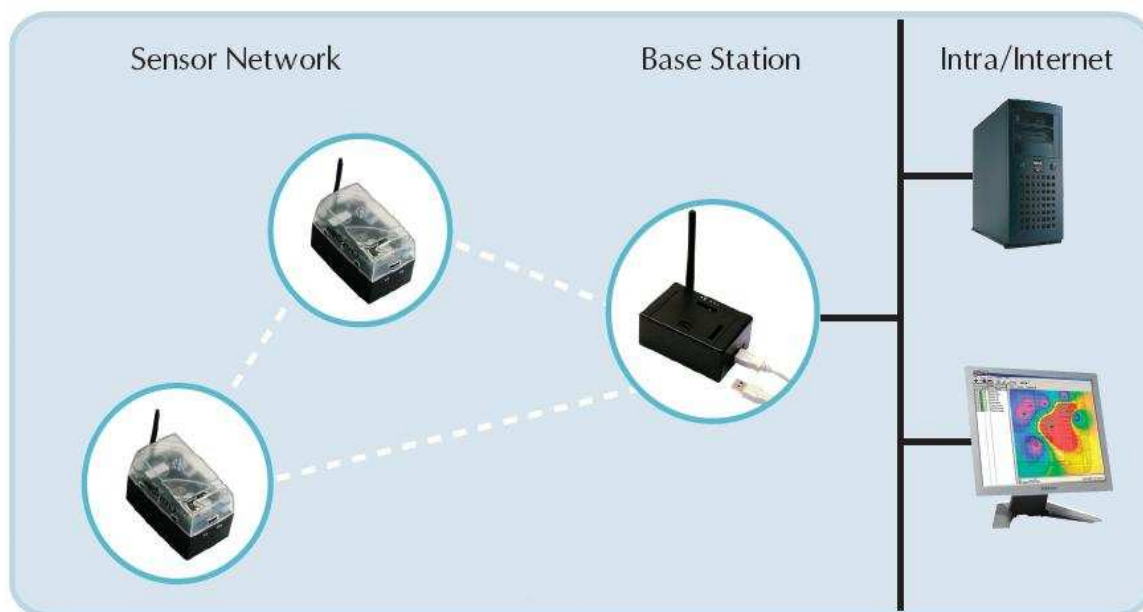
5.1.2 MEMSIC Wireless Sensor Network

MEMSIC Starter Kit poskytuje snadné a cenově efektivní řešení, jak získat z první ruky zkušenosti s bezdrátovou senzorovou sítí. Tato základní sada obsahuje všechny komponenty potřebné pro rychlé nasazení základní bezdrátové senzorové sítě ^[47]. Obsahuje senzorové uzly, bránu a software pro monitorování senzorové sítě MoteView.

Radiová technologie používaná bezdrátovými moduly MEMSIC podporuje domácí a mezinárodní kmitočtová pásma 433 MHz, 868-915 MHz a 2,4 GHz. V pásmu 2,4 GHz podporuje standardy 802.15.4 a ZigBee. Rychlost přenosu dat se pohybuje od 19,2 kbps do 240 kbps.

Hardwarové a softwarové vybavení

Součástí sady jsou dvě senzorové desky MTS400 pro měření teploty, vlhkosti, barometrického tlaku a pro monitorování okolního světla. Tyto univerzální senzorové desky jsou určeny pro širokou škálu aplikací od jednoduché bezdrátové meteorologické stanice po úplnou síť pro sledování životního prostředí uzlů. Použití je možné v zemědělství, průmyslu, lesnictví a dalších. Další součástí sady je MIB520 brána (gateway) s USB portem, která propojuje bezdrátovou senzorovou síť s počítačem nebo Internetem. Sensorové uzly a brána jsou předem nakonfigurovány.

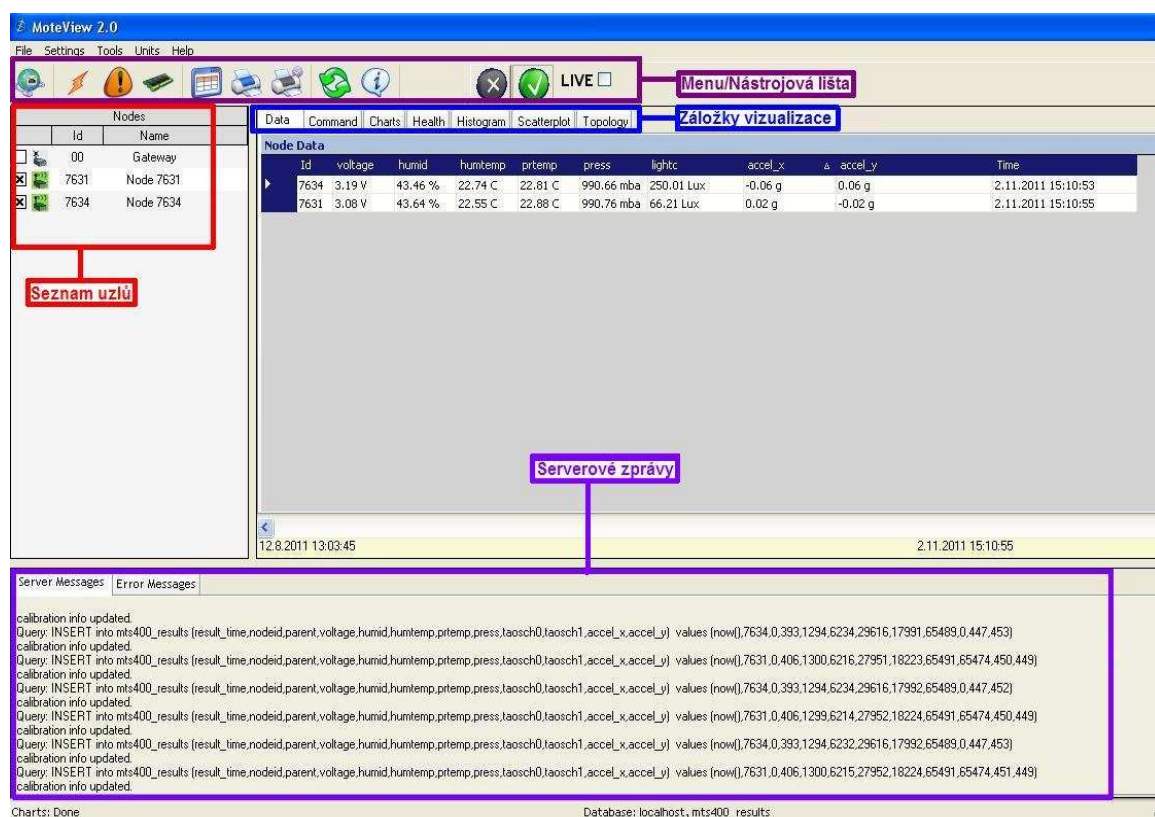


Obr. 18: Architektura senzorové sítě ^[47]

Aplikace MoteView pro operační systém Windows poskytuje intuitivní grafické uživatelské rozhraní pro sledování a správu bezdrátové senzorové sítě. Zobrazením topologie sítě, tabulkami a grafy, stejně jako jednoduchou konfigurací senzorových uzlů umožňuje MoteView uživateli pochopit senzorová data a síť. Pro spuštění aplikace jsou požadovány tyto komponenty: PostgreSQL 8.0, PostgreSQL ODBC a Microsoft .NET 1.1. Tyto komponenty jsou součástí sady.

MoteView je produkt Crossbow Technology. Je navržen jako klientské rozhraní mezi uživatelem a vytvořenou senzorovou sítí. MoteView poskytuje uživatelům nástroje, které zjednodušují rozmístění a monitorování. Umožňuje také snadné připojení k databázi, analýzu a vytváření grafů. MoteView má čtyři hlavní části uživatelského rozhraní:

- Panel nástrojů/Menu – umožňuje uživateli zadat akce a zahájit příkazem dialogy
- Seznam uzlů – zobrazuje všechny rozmístěné uzly a umožňuje konfiguraci zobrazování jednotlivých uzlů
- Záložky vizualizace – umožňuje uživateli prohlížet senzorová data různými způsoby
- Serverové zprávy – zobrazuje seznam událostí a příchozí zprávy



Obr. 19: Uživatelské rozhraní MoteView

Importovat data do MoteView lze pomocí vytvořeného *.sql souboru. Export dat z MoteView lze provést pěti různými způsoby:

- 1) export databáze do souboru *.sql – tento soubor obsahuje SQL příkazy pro obnovení databáze. Obsahuje příkazy pro znovu vytvoření všech tabulek, indexů, pravidel atd. a obnovení všech naměřených hodnot.
- 2) export tabulky do souboru *.sql – soubor obsahuje SQL příkazy pro obnovení tabulky v databázi, jejich indexů a pravidel a obnovení jejich hodnot.

Pozn.: Jednotlivé SQL příkazy je nutné samostatně kopírovat z vytvořeného souboru do okna pro provádění SQL dotazů (Query), aby se daly případně opravit pro potřeby dané databáze.

- 3) export tabulky do Spreadsheet souboru – exportuje obsah tabulky do souboru s oddělenými hodnotami (*.csv). Oddělovačem může být čárka, tabulátor nebo mezera. Pro potřebu exportu lze zvolit časové období hodnot, jednotlivý uzel nebo všechna data a všechny uzly.
- 4) export tabulky do XML – viz obrázek 20
- 5) export do JPEG – umožňuje exportovat aktivní záložku vizualizace. Výsledný obrázek má stejnou podobu jako jednotlivé záložky v aplikaci.


```

<?xml version="1.0" standalone="yes"?>
<NewDataSet>
  <mts400_results>
    <result_time>2011-08-12T13:03:45.6400000+02:00</result_time>
    <nodeid>7631</nodeid>
    <parent>7634</parent>
    <voltage>408</voltage>
    <humid>1272</humid>
    <humtemp>6551</humtemp>
    <prtemp>28396</prtemp>
    <press>18081</press>
    <taosch0>65521</taosch0>
    <taosch1>65508</taosch1>
    <accel_x>450</accel_x>
    <accel_y>448</accel_y>
  </mts400_results>
  <mts400_results>
    <result_time>2011-08-12T13:03:46.0310000+02:00</result_time>
    <nodeid>7634</nodeid>
    <parent>0</parent>
    <voltage>394</voltage>
    <humid>1260</humid>
    <humtemp>6580</humtemp>
    <prtemp>30116</prtemp>
    <press>17848</press>
    <taosch0>65510</taosch0>
    <taosch1>0</taosch1>
    <accel_x>448</accel_x>
    <accel_y>450</accel_y>
  </mts400_results>
  <mts400_results>
    <result_time>2011-08-12T13:03:54.0460000+02:00</result_time>
    <nodeid>7631</nodeid>

```

Obr. 20: Ukázka části kódu exportu obsahu tabulky do XML

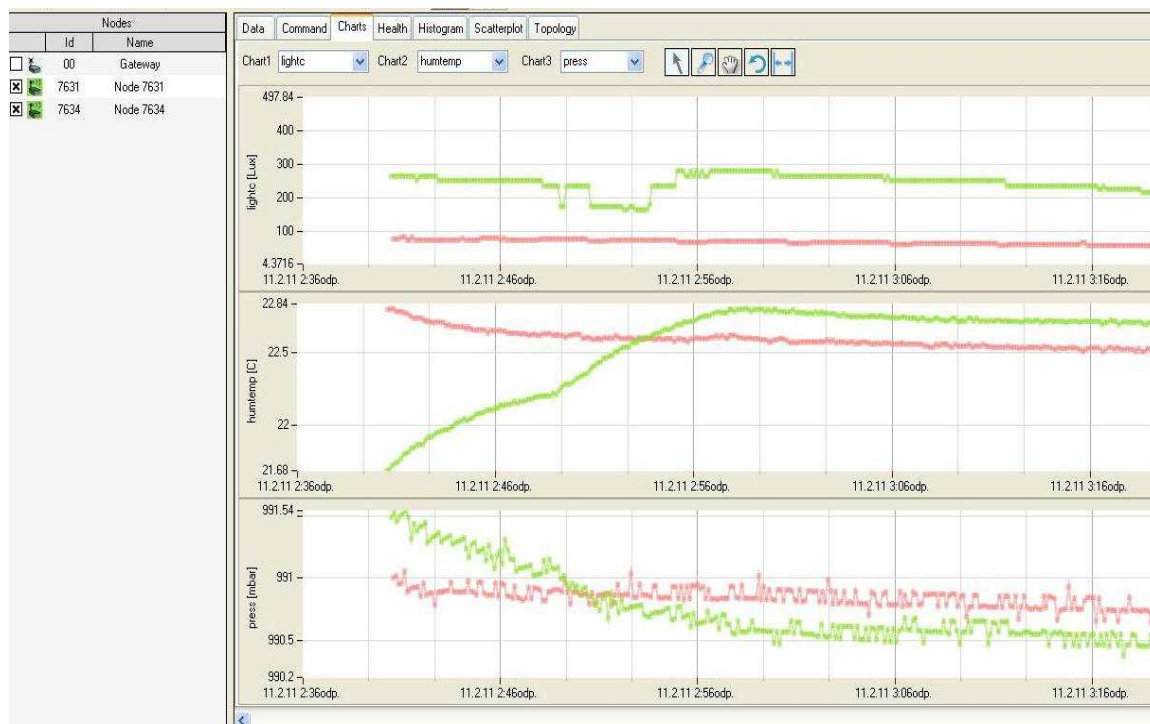
MoteView umožňuje sedm různých způsobů vizualizace sensorových dat (*Data*, *Command*, *Charts*, *Health*, *Histogram*, *Scatterplot* a *Topology*). První vizualizační záložka *Data* zobrazuje poslední data odeslaná sensorovými uzly. Sensorová data jsou automaticky uváděna v běžně používaných jednotkách. Klepnutím pravým tlačítkem myši na záhlaví sloupce lze zobrazit menu pro převody jednotek. Klepnutím levým tlačítkem na záhlaví sloupce pak lze tento sloupec seřadit. Na obrázku 21 jsou zobrazena poslední data odeslaná sensorovými uzly s Id označením 7634 a 7631. Tyto sensorové uzly mají čidla měřící napětí baterie, vlhkost, teplotu vlhkosti, teplotu tlaku, barometrický tlak, okolní světlo a vibrace. Odeslané hodnoty byly změřeny 2.11.2011 v čase okolo 15 hodin a 10 minut.

Data	Command	Charts	Health	Histogram	Scatterplot	Topology					
Node Data											
▶	Id	voltage	humid	humtemp	prtemp	press	lightc	accel_x	Δ accel_y	Time	
	7634	3.19 V	43.46 %	22.74 C	22.81 C	990.66 mba	250.01 Lux	-0.06 g	0.06 g	2.11.2011 15:10:53	
	7631	3.08 V	43.64 %	22.55 C	22.88 C	990.76 mba	66.21 Lux	0.02 g	-0.02 g	2.11.2011 15:10:55	

Obr. 21: Vizualizační záložka Data

Záložka *Command* uživateli dovoluje měnit parametry uzlu, a to minimální přenosovou rychlost dat a nastavení barvy LED (Light-Emitting Diode) diody pro určitý stav uzlu.

Charts vizualizační záložka poskytuje možnost vytvářet grafy senzorových měření. Umožňuje vykreslit tři různé grafy pro maximálně 24 uzlů vybraných ze seznamu uzlů v levé části aplikace. Uživatel si pomocí několika nástrojů může s grafy posouvat a přibližovat si či oddalovat na vybrané části grafů.



Obr. 22: Vizualizační záložka Charts

Na obrázku 22 jsou zobrazeny grafy okolního světla (Lux), teploty vlhkosti (°C) a barometrického tlaku (mbar) pro senzorové uzly s Id označením 7634 a 7631. Hodnoty senzorového uzlu 7631 jsou znázorněny růžovou barvou a hodnoty senzorového uzlu 7634 jsou znázorněny barvou zelenou. Grafy zobrazují hodnoty naměřené 2.11.2011 v čase 14:40 až 15:20.

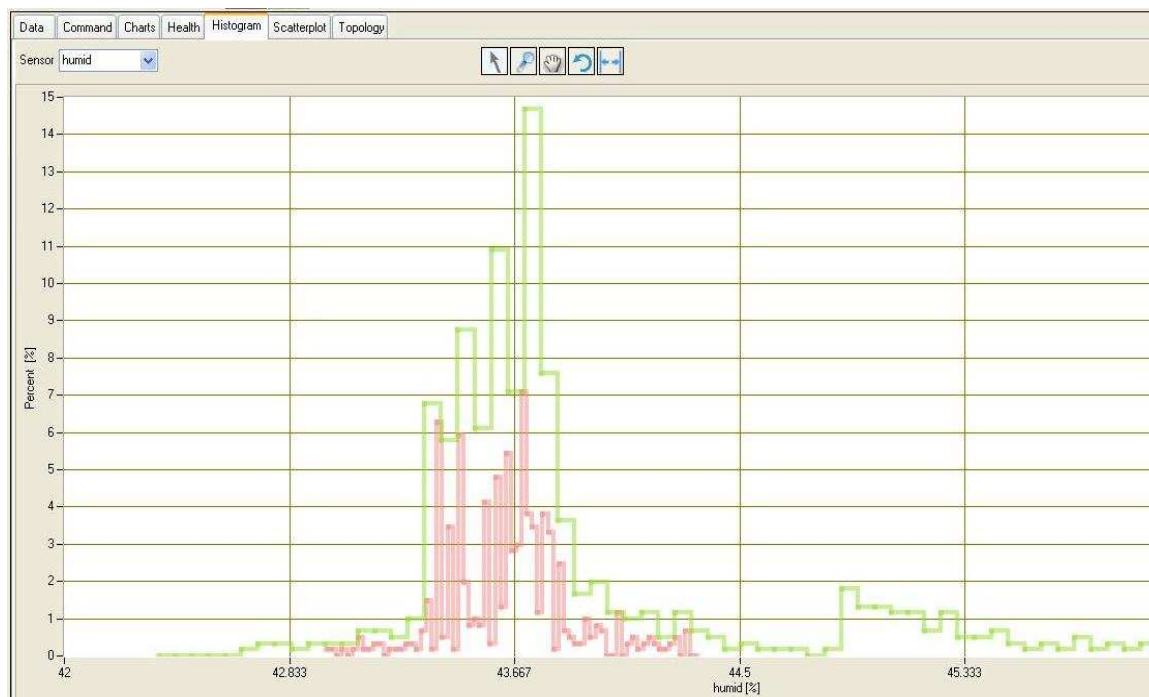
Záložka *Health* zobrazuje informace o „zdraví“ jednotlivých uzlů. Každý uzel v síti zasílá v určitém časovém intervalu tzv. pakety zdraví, které oznamují stav jednotlivých uzlů (např. napětí baterie, atd.). Klepnutím pravým tlačítkem myši na záhlaví sloupce lze opět zobrazit menu pro převody jednotek. Klepnutím levým tlačítkem na záhlaví sloupce pak lze tento sloupec seřadit.

Data	Command	Charts	Health	Histogram	Scatterplot	Topology								
Node Health														
▶	Id	health_pkts	node_pkts	forwarded	dropped	retries	battery	power_sum	board_id	quality_tx	quality_rx	path_cost	parent_rssi	Time
	7631	6.3 %	98.74 %	0 %	1.26 %	0 %	3 v	0 mAHr	133	100 %	100 %	4	33	2.11.2011 15:09:31
	7634	6.33 %	98.73 %	0 %	1.27 %	0 %	3.1 v	0 mAHr	133	100 %	93.33 %	4	47	2.11.2011 15:09:27

Obr. 23: Vizualizační záložka Health

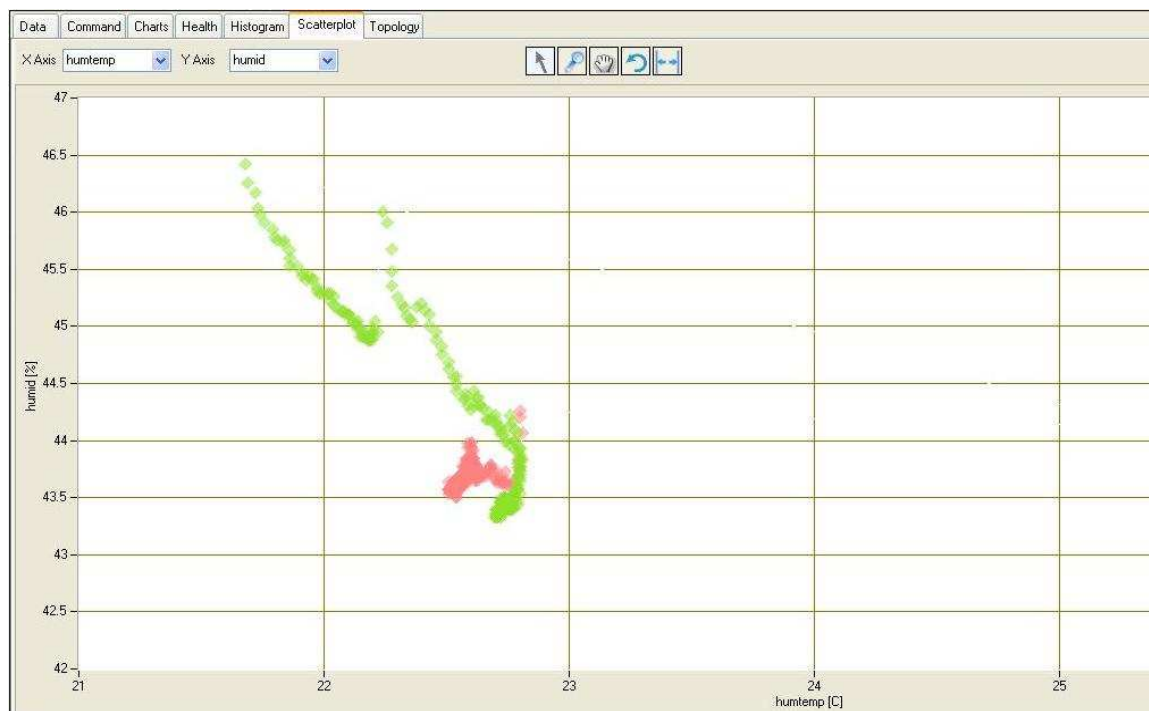
Obrázek 23 je ukázkou vizualizační záložky *Health*. Sloupec *health_pkts* udává hrubé číslo podílu paketů zdraví k celkovému počtu paketů uzlu. *Node_pkts* je celkový počet paketů, které vznikly na uzlu od posledního restartu. *Forwarded* udává podíl počtu paketů, které byly uzlu předány z jiného uzlu k celkovému počtu paketů. *Dropped* udává podíl počtu paketů, které uzlu spadly k celkovému počtu paketů. Paket je považován za spadnutý, pokud byl již osmkrát přenášén a ani jednou se nepotvrdil úspěšný přenos. *Retries* udává celkový počet opakovaných přenosů. *Battery* udává napětí baterie uzlu. *Power_sum* je vždy 0, jelikož toto měření není v současnosti podporováno. *Board_id* udává identifikátor připojené sensorové desky. *Quality_tx* a *quality_rx* jsou odhady úrovně kvality, která bere v potaz kolize paketů. *Path_cost* je odhadovaný počet přenosů potřebných k odeslání paketu z uzlu k základní stanici (bráně). *Parent_rssi* udává úroveň síly signálu mezi uzlem a bránou.

Záložka *Histogram* zobrazuje sloupcový graf, který graficky shrnuje statistické rozdělení sensorových dat. Z rozvíracího seznamu pro vykreslování lze vybrat pouze jeden senzor, který se bude zobrazovat. Na obrázku 24 je zobrazena vlhkost pro sensorové uzly s Id označením 7634 a 7631. Hodnoty sensorového uzlu 7631 jsou znázorněny růžovou barvou a hodnoty sensorového uzlu 7634 jsou znázorněny barvou zelenou. Podobně jako u vizualizační záložky *Graphs* lze vykreslit histogramy pro maximálně 24 uzlů, volitelných ze seznamu uzlů v levé části aplikace. Uživatel opět může použít stejné nástroje pro manipulaci s histogramy jako v záložce *Graphs*.



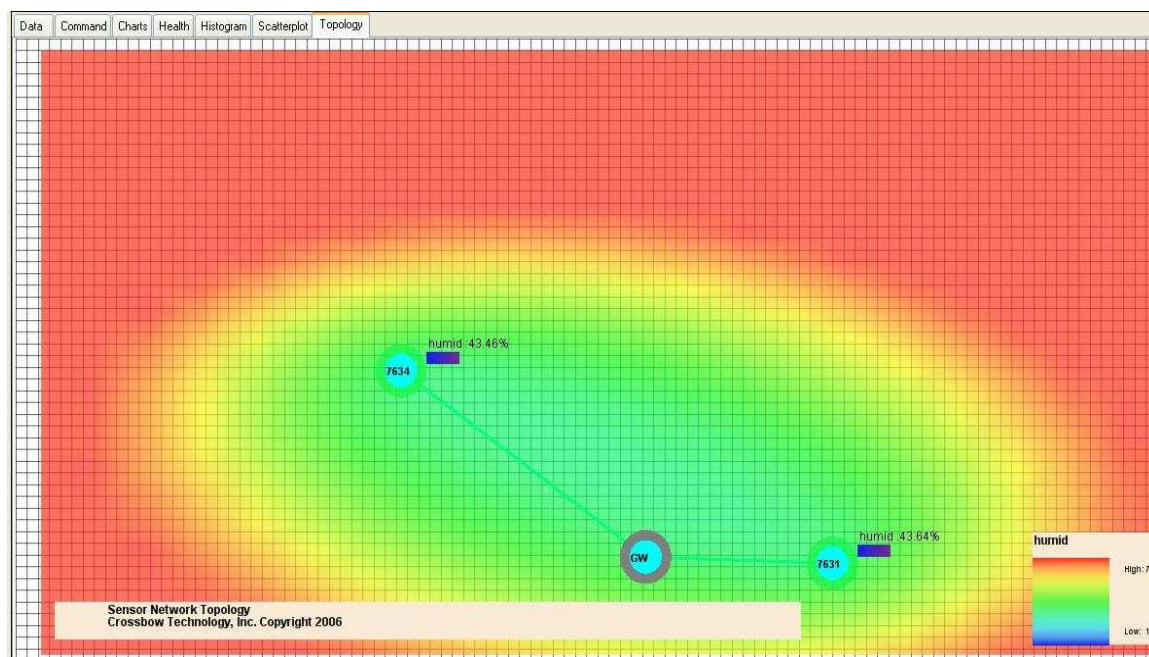
Obr. 24: Vizualizační záložka Histogram

Další vizualizační záložkou je *Scatterplot*, která umožňuje získat vizuální porovnání dvou senzorových datových sad a určuje tak jejich vztah. Opět lze použít nástroje pro manipulaci s grafem, jako tomu bylo u záložek *Graphs* a *Histogram*. Klepnutím pravým tlačítkem je možné zvolit rozsah zobrazovaných hodnot (poslední hodina, poslední den, poslední týden, poslední měsíc, poslední čtvrtletí a všechna data). Obrázek 25 zobrazuje porovnání všech naměřených hodnot teploty vlhkosti a vlhkosti. Hodnoty senzorového uzlu 7631 jsou znázorněny růžovou barvou a hodnoty senzorového uzlu 7634 jsou znázorněny barvou zelenou.



Obr. 25: Vizualizační záložka Scatterplot

Poslední záložkou je *Topology*, která zobrazuje mapu senzorové sítě s umístěním uzlů. Uživatel může jednotlivé uzly přetahovat, a měnit tak jejich umístění a vzájemné vztahy v síti. Umístění uzlů je uloženo v databázi. Klepnutím pravým tlačítkem do mapy lze vyvolat menu, ve kterém je pak možné nastavit vlastnosti vizualizace. Obrázek 26 zobrazuje umístění senzorových uzlů v síti a naměřenou vlhkost s vyjádřením možné okolní vlhkosti.



Obr. 26: Vizualizační záložka Topology

Aby MoteView zobrazovalo data z aktivní senzorové sítě, je potřeba zatrhnout políčko LIVE umístěné na nástrojové liště v horní části aplikace. V režimu LIVE je pak obnovován seznam uzlů, grafy a zobrazení topologie. Jestliže je režim LIVE neaktivní, lze využít nástroj přehrávání v dolní části záložek vizualizace. Ten umožňuje procházet shromážděná data v různých časech. Toto rozhraní podporují záložky *Data* a *Topology*.

Detailní popis tohoto softwaru je dostupný na stránkách:

<http://www.xbow.com/index.html>;

http://www.willow.co.uk/html/mote-view_client_software.html;

http://www.willow.co.uk/html/moteworks_software_platform.html

Uložení dat v databázi

Všechny nástroje pro vizualizaci v MoteView vyžadují připojení k databázi. Tato databáze může být uložena na vlastním počítači (localhost), vzdáleném serveru nebo Stargate. Databáze je omezena velikostí úložiště, které má systém k dispozici.

Aplikace MoteView využívá databáze PostgreSQL 8.0. Data se ukládají do databáze s názvem *task*. Tabulky v této databázi jsou již předpřipraveny pro jednotlivé typy senzorových desek. Každý typ senzorové desky má v databázi přiřazeny tři tabulky (např. *mts400_results*, *mts400_results_i*, *mts400_l*). První tabulka *mts400_results* obsahuje naměřené hodnoty jednotlivých sledovaných veličin, id senzoru a čas. Tabulka *mts400_results_i* obsahuje souřadnice brány a senzorových uzlů. Poslední tabulka *mts400_l* obsahuje poslední naměřené hodnoty veličin na jednotlivých uzlech. Kromě těchto tabulek a prázdných tabulek ostatních typů senzorových desek jsou v databázi ještě uloženy tabulky s výsledky stavů uzlů během měření (např. stav baterie) a tabulky s ukázkovými daty.

public.mts400_results	public.mts400_results_i	public.mts400_results_l
- result_time: timestamp - epoch: integer - nodeid: integer - parent: integer - voltage: integer - humid: integer - humtemp: integer - prtemp: integer - press: integer - taosch0: integer - taosch1: integer - accel_x: integer - accel_y: integer	- nodeid: integer - x_coord: integer - y_coord: integer - z_coord: integer - mote_name: text - calib: bytea - board_id: integer	- result_time: timestamp - epoch: integer - nodeid: integer - parent: integer - voltage: integer - humid: integer - humtemp: integer - prtemp: integer - press: integer - taosch0: integer - taosch1: integer - accel_x: integer - accel_y: integer
- mts400_nodeid_idx - mts400_time_idx		

Obr. 27: Datové typy atributů tabulek senzorové desky typu MTS400

Na obrázku 27 jsou znázorněny datové typy jednotlivých sloupců tabulek určených pro senzorovou desku typu MTS400. Mezi tabulkami neexistují žádné relace.

Přenos dat do vlastní databáze

Nejprve je opět potřeba získat data, která budou transportována do vlastní databáze. Data z MoteView vyexportujeme nejlépe do Spreadsheet souboru (*.csv). Vytvořený soubor není třeba nijak upravovat, jako to bylo u Fiedler-Mágr. Exportovaný soubor ihned obsahuje hlavičku s názvy sloupců a jednotlivé hodnoty sloupců. Není potřeba nic odmazávat.

Před prvním přesunem dat z *.csv souboru se opět musí vytvořit vlastní databáze. V tomto případě se vytvořená databáze bude jmenovat *MOTE* a její kódování bude UTF-8.

Ukázka SQL kódu vytvoření databáze pro transportovaná data:

```
sql4: CREATE DATABASE "MOTE"
      WITH OWNER = postgres
      ENCODING = 'UTF8'
      TABLESPACE = pg_default
      CONNECTION LIMIT = -1;
```

Dalším krokem je vytvoření tabulky, jejíž atributy budou mít stejné názvy jako atributy v hlavičce *.csv souboru. Ukázka zápisu kódu vytvoření tabulky:

```
sql5: CREATE TABLE "MTS400" (
  "Id" integer,
  "Sample" integer,
  "Time" character varying,
  "parent" integer,
  "voltage" character varying,
  "humid" character varying,
  "humtemp" character varying,
  "prtemp" character varying,
  "press" character varying,
  "lightc" character varying,
  "accel_x" character varying,
  "accel_y" character varying
)
WITH (OIDS=FALSE);
```

Import dat z *.csv souboru do vytvořené vlastní databáze se provádí pomocí SQL příkazu COPY. Tento příkaz obsahuje název tabulky, do které jsou data vkládána, názvy jednotlivých sloupců tabulky, cestu k souboru a oddělovač, kterým jsou data v souboru oddělena. Dále příkaz specifikuje typ souboru, ze kterého jsou data kopírována do tabulky

(tedy *.csv) a specifikuje, že soubor obsahuje hlavičku s názvy jednotlivých sloupců. Ukázka zápisu kódu pro příkaz COPY:

```
sql6: COPY "MTS400"  
("Id", "Sample", "Time", "parent", "voltage", "humid", "humtemp", "prtemp", "pre  
ss", "lightc", "accel_x", "accel_y")  
FROM '/exportM0TEdatabase.csv'  
WITH DELIMITER ',', '  
CSV HEADER;
```

5.1.3 EasyWeather

ProWeatherStation™ je výkonná bezdrátová meteorologická stanice zaznamenávající data moderní senzorovou technologií s výsledky s dobrou přesností při nízkých nákladech. Poskytuje okamžité a historické údaje vnitřní i vnější teploty a vlhkosti, rychlost větru (průměr a v nárazech), směr větru, barometrický tlak a srážky^[35].

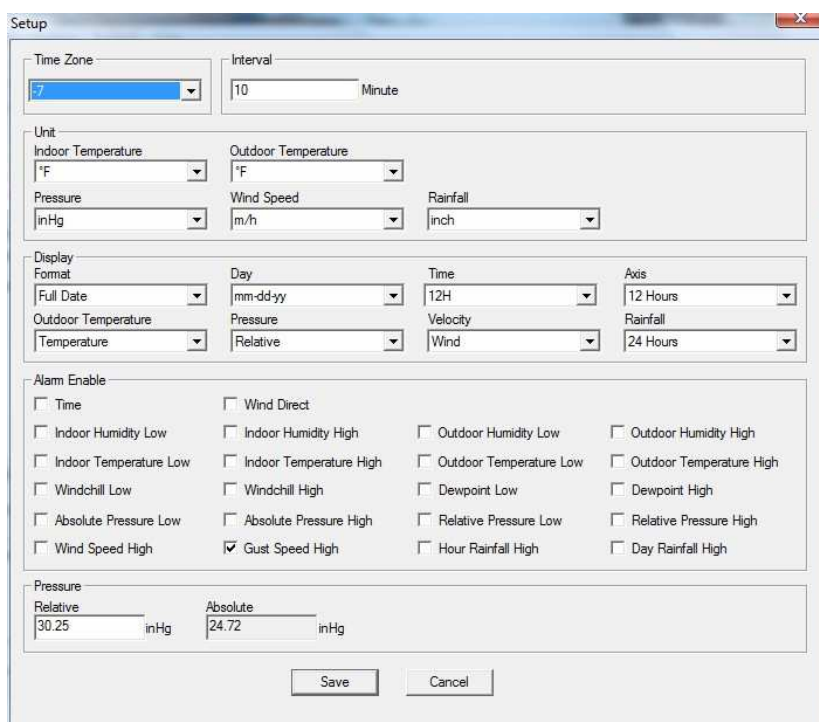
Tato stanice obsahuje vnitřní dotykový ovladač s kapacitou 4080 datových bodů s nastavitelným odstupem (minimum 5 min.) intervalu vzorkování z venkovní stanice. Displej ukazuje stav paměti. Pokud je paměť zaplněná, data lze stáhnout pomocí USB portu a archivovat na počítači. Ovladač funguje na baterie AA a může být ve vzdálenosti do 100 metrů od senzorů. Vysílací frekvence je 915 MHz. Senzorové pole vydrží na dvě AA baterie až dva roky, takže neexistují žádné starosti s montáží v horní části budovy nebo věže.



Obr. 28: Venkovní stanice a vnitřní ovladač ProWeatherStation^[35]

EasyWeather software pro operační systém Windows umožňuje vzdálené pozorování ProWeatherStation. Hlavní okno aplikace ukazuje změřené hodnoty spolu s maximálními a minimálními hodnotami pro relativní a absolutní tlak, rosný bod, pocitovou teplotu, vnitřní a vnější teplotu, vlhkost. Dále ukazuje srážky za poslední hodinu, posledních 24h, poslední týden, poslední měsíc a celkem a rychlost a směr větru. Další okna jsou pak pro nastavení parametrů stanice a alarmů, prohlížení a export zaznamenaných dat a vizualizaci v grafech.

EasyWeather software exportuje data ve dvou formátech (XLS a TXT). V rámci nastavení exportu lze vytvořit hlavičku, určit pořadí naměřených prvků v seznamu a vybrat typ oddělovače (tabulátor, mezera, |, “;“, “.:“, “;“, “:“). Grafy mohou být exportovány do formátu JPEG.



Obr. 29: EasyWeather - okno pro nastavení parametrů a alarmů ^[35]

Přenos exportovaných dat do vlastní databáze

Vyexportovaný soubor *.txt je potřeba upravit tak, aby jej bylo možné přenést do databáze PostgreSQL. Je potřeba nastavit kódování na UTF-8 a odmazat hlavičku v souboru, jinak by se po přenosu zobrazovala na prvním řádku v tabulce. Opět je potřeba vytvořit databázi (CREATE DATABASE), ale hlavně tabulku, do které budou data transportována (CREATE TABLE). Pomocí příkazu COPY pak zkopírovat údaje z textového souboru do tabulky *meteostanice* v databázi *EASYWEATHER*. Následující kód popisuje postup:

sql7: CREATE DATABASE "EASYWEATHER"

```
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1
```

sql8: CREATE TABLE "meteostanice" (
 "No" character varying,
 "Time" character varying,
 "Interval(mi)" character varying,
 "Indoor Humidity(%)" character varying,


```

"Indoor Temperature(°C)" character varying,
"Outdoor Humidity(%)" character varying,
"Outdoor Temperature(°C)" character varying,
"Absolute Pressure(Hpa)" character varying,
"Wind(m/s)" character varying,
"Gust(m/s)" character varying,
"Direction" character varying,
"Relative Pressure(Hpa)" character varying,
"Dewpoint(°C)" character varying,
"Windchill(°C)" character varying,
"Hour Rainfall(mm)" character varying,
"24 hour Rainfall(mm)" character varying,
"Week Rainfall(mm)" character varying,
"Month Rainfall(mm)" character varying,
"Total Rainfall(mm)" character varying,
"Wind Level(bft)" character varying,
"Gust Level(bft)" character varying
)
WITH (OIDS=FALSE);

```

```

sql9: COPY "meteostanice"("No","Time","Interval(mi)","Indoor Humidity(%)",
"Indoor Temperature(°C)","Outdoor Humidity(%)",
"Outdoor Temperature(°C)","Absolute Pressure(Hpa)","Wind(m/s)",
"Gust(m/s)","Direction","Relative Pressure(Hpa)","Dewpoint(°C)",
"Windchill(°C)","Hour Rainfall(mm)","24 hour Rainfall(mm)",
"Week Rainfall(mm)","Month Rainfall(mm)","Total Rainfall(mm)",
"Wind Level(bft)","Gust Level(bft)")
FROM '/meteostanice.txt'
WITH DELIMITER ' ';

```

5.1.4 HOBO Event Data Logger

Datový box značky HOBO slouží ke stahování naměřených dat z čidel stejné značky, které jsou umístěny v terénu. Dále se využívá k překontrolování stavu baterie čidel a k nastavení jejich základních parametrů pro měření. HOBO Event Data Logger je primárně využíván pro překlápěcí člunkové srážkoměry a průmyslové počítačové aplikace^[14]. Dokáže zaznamenat až 8000 událostí a pro každou z nich uložit čas a datum měření. Záznamy z datového boxu jsou přeneseny do počítače pomocí programu BoxCar Pro. Z tohoto programu jsou naměřená data exportována do textového dokumentu ve formátu TXT.

Přenos dat do vlastní databáze

Vyexportovaný textový soubor je nutné pro přenos do databáze PostgreSQL upravit. Je potřeba odstranit první řádek a označení počátku a konce měření (Start, End). Výsledný upravený soubor je uložen opět jako textový dokument (*.txt) s kódováním UTF-8.

Před prvním přenosem dat ze souboru do databáze se musí vytvořit prázdná databáze, do které budou naměřené hodnoty uloženy. V tomto případě se vytvořená databáze bude jmenovat *HOBO* a její kódování bude UTF-8. Ukázka SQL kódu vytvoření databáze pro transportovaná data:

```
sql10: CREATE DATABASE "HOBO"
      WITH OWNER = postgres
      ENCODING = 'UTF8'
      TABLESPACE = pg_default
      CONNECTION LIMIT = -1
```

Dalším krokem je vytvoření tabulky. Ukázka zápisu kódu vytvoření tabulky:

```
sql11: CREATE TABLE "hobo" (
  "DateTime" character varying,
  "Value" character varying
)
WITH (OIDS=FALSE);
```

Import dat do vytvořené vlastní databáze se opět provádí pomocí SQL příkazu COPY, který obsahuje název tabulky, do které jsou data vkládána, názvy jednotlivých sloupců tabulky, cestu k souboru *.txt a oddělovač, kterým jsou data v souboru oddělena (v tomto případě se jedná o tabulátor). Ukázka zápisu kódu pro příkaz COPY:

```
sql12: COPY "hobo"("DateTime", "Value")
FROM '/hobo1.txt'
WITH DELIMITER ' ';
```

5.1.5 MiniCube

Datalogger MiniCube od firmy Environmental Measuring System Brno je zařízení pro automatické zaznamenávání údajů v náročných terénních aplikacích. Je určeno pro připojení široké škály různých senzorů. Datalogger nemá žádný displej ani klávesnici, proto jeho nastavení a stažení dat vyžaduje připojení k počítači ^[8]. Počítač lze připojit pomocí kabelu nebo vzdáleného připojení (sítě GSM). K modelu MiniCube VV/VX je možné připojit dvanáct různých napěťových vstupů a jeho paměťová kapacita dosahuje až 220000 uložených hodnot. Stažení dat z dataloggeru se provádí pomocí programu Mini32, který je volně stažitelný z webových stránek společnosti Environmental Measuring System Brno. Z tohoto programu jsou naměřená data exportována do textového dokumentu ve formátu DAT.

Přenos dat do vlastní databáze

Vyexportovaný textový soubor je nutné pro přenos do databáze PostgreSQL upravit, a to nejvíce ze všech předchozích. Jelikož se jedná o textový soubor formátu DAT, nejsou jednotlivé sloupce od sebe odděleny žádným separátorem, a proto když je tento soubor otevřen v nějakém tabulkovém editoru (např. Excel), zobrazují se sloupce se záznamy všechny v jednom sloupci. Je potřeba, aby sloupce byly oddělené. To lze provést v Excelu funkcí *Text do sloupců*, která se nachází pod záložkou *Data*.

V dalším kroku se musí odstranit první řádek, který obsahuje názvy jednotlivých sloupců, tak aby výsledný soubor obsahoval pouze hodnoty vkládané do databáze. Výsledný upravený soubor je uložen jako *.csv (oddělený středníkem).

Opět je potřeba vytvořit databázi (CREATE DATABASE) a tabulku, do které budou data přenesena (CREATE TABLE). Příkazem COPY budou údaje ze souboru zkopírovány do tabulky *minicube* v databázi *MINICUBE*. Následující kód popisuje postup:

```
sql13: CREATE DATABASE "MINICUBE"
      WITH OWNER = postgres
      ENCODING = 'UTF8'
      TABLESPACE = pg_default
      CONNECTION LIMIT = -1
```

```
sql14: CREATE TABLE "minicube"(
  "Datum" character varying,
  "Čas" character varying,
  "Teplota vzdu" character varying,
  "Vlhkost vzdu" character varying,
  "Pudni teplot" character varying,
  "Pudni teplot2" character varying,
  "Pudni vlhkos" character varying,
  "Pudni vlhkos2" character varying,
  "Tint" character varying,
  "SWP [hPa]" character varying,
  "SWP 2 [hPa]" character varying,
  "Voltage" character varying,
  "Global radi" character varying,
  "Voltage 2" character varying,
  "Napeti bate" character varying,
  "Srazky [mm]" character varying
)
WITH (OIDS=FALSE);
```

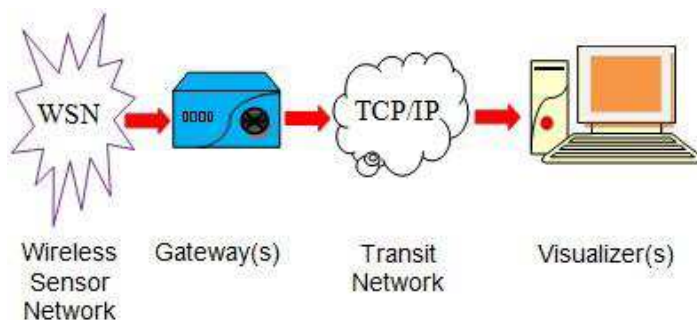
```
sql15: COPY "minicube"("Datum", "Čas","Teplota vzdu","Vlhkost vzdu","Pudni
teplot","Pudni teplot2","Pudni vlhkos","Pudni vlhkos2","Tint","SWP
[hPa]","SWP 2 [hPa]","Voltage","Global radi","Voltage 2","Napeti
bate","Srazky [mm]")
FROM '/minicube1.csv'
WITH DELIMITER ';;';
```

5.2 Další nástroje

5.2.1 SpyGlass

SpyGlass je modulární a rozšiřitelný vizualizační rámec pro sensorové sítě. Aplikace je primárně určena k vizualizaci sensorových sítí na obrazovce. Zobrazuje soubor sensorových uzlů a jejich umístění na mapě, stejně jako jakákoli data posílaná uzly, např. stav propojení mezi uzly nebo sesbírané údaje shromážděné čidly ^[32].

Dokumentace k tomuto programu je dostupná z <http://www.itm.uni-luebeck.de/projects/spyglass-trac>



Obr. 30: Hlavní komponenty SpyGlass ^[32]

5.2.2 TinyViz

TinyViz poskytuje rozšiřitelné uživatelské grafické rozhraní pro ladění, vizualizace, simulace a interakci s TOSSIM aplikací TinyOS. Použitím TinyViz lze snadno sledovat výkon aplikací TinyOs, nastavit zářezky při zajímavých událostech, vizualizovat rádiové zprávy a manipulovat s virtuální pozicí a rádiovou konektivitou částí. Kromě toho podporuje TinyViz jednoduchý plugin API, který umožní napsat vlastní TinyViz moduly pro vizualizaci dat v aplikaci – specifickým způsobem nebo v interakci s běžící simulací ^[32].

5.2.3 Surge Network Viewer

Surge Network Viewer je Java aplikace, která je standardně v distribuci TinyOS Tools. Je užitečná pro monitorování sensorové sítě a analýzu výkonu sítě. Mezi její vlastnosti patří: automatické zjišťování a konfigurace sítě, prohlížení topologie sensorové

sítě, zápis dat a zobrazení statistických údajů o síti včetně výnosu, předpokládaného výnosu a kvality spojení pro každý uzel, grafický nástroj pro prohlížení uložených dat ^[32].

Potrobnější informace a dokumentace jsou dostupné na: http://www.cmt-gmbh.de/surge_network_viewer.htm; <http://www.willow.co.uk/html/software.html>

5.2.4 MonSense

MonSense je aplikace k navrhování, monitorování a řízení bezdrátových sensorových sítí, které slouží různým cílům. Je velmi modulární a má různá rozšíření. MonSense zobrazí stávající propojení jako neorientované grafy, jehož uzly jsou čidla přístroje a hrany jsou aktuální propojení. Tato aplikace má sloužit dvěma typům uživatelů: zákazníci WSN a výzkumní pracovníci WSN. Aplikace musí být schopna připojit různé sensorové sítě současně, což umožňuje existenci více než jedné základní stanice pro stejnou sensorovou síť. Získaná data musí být srozumitelná konečným uživatelům a volitelně mohou být zveřejněna na internetu jako HTML, prostý text nebo obrázky, což umožní přístup k informacím aniž by bylo potřeba instalace softwaru ^[32].

Podrobnější informace a dokumentace jsou dostupné na: <http://whale.fe.up.pt/wsnwiki/index.php/MonSense>;
<http://whale.fe.up.pt/wsnwiki/images/7/7b/MonSense.doc>

5.2.5 NetTopo

NetTopo je rozšiřitelný integrovaný rámec pro simulaci virtuální WSN, vizualizaci reálných testbed a interakci mezi simulovanou WSN a testbed na pomoc zkoumání algoritmů ve WSN. Většina stávajících vizualizačních nástrojů podporuje pouze jediný druh sensorových sítí a jsou velmi často vázané na TinyOS. Nicméně, NetTopo se zaměřuje na vizualizaci a řízení testbedů WSN, kde jsou použity heterogenní prostředky, např. bezdrátová kamera, sensorové zařízení založené na Bluetooth, a tyto zařízení obecně nepoužívají TinyOS ^[32]. Dokumentace k tomuto programu je dostupná z <http://sites.google.com/site/leishuderi/nettopo>

5.2.6 Octopus

Octopus je open-source panel pro vizualizaci a ovládání sensorové sítě v TinyOS prostředí. Poskytuje uživatelům grafické uživatelské rozhraní (GUI) pro zobrazení topologie sensorové sítě. To umožňuje dynamicky řídit chování sensorových uzlů, jako např. spotřebu energie, dobu odběru vzorků, pracovní cyklus a formulování dotazů na uzly ^[32].

Dokumentace k tomuto programu je dostupná z <http://www.csi.ucd.ie/content/octopus-dashboard-sensor-networks-visual-control>

5.2.7 TOSGUI

TOSGUI je projekt skládající se z modulárních komponent, které mohou být použity k vytvoření vlastní aplikace. Bohužel, komponent architektury je úzce spojen s TinyOS a Mote hardware platformou ^[32]. Dokumentace k tomuto programu je dostupná z <http://deneb.cs.kent.edu/~mikhail/classes/es.u02/>

5.2.8 MSR Sense

MSR Sense projekt je schopen sbírat data ze sensorové sítě a vizualizovat je, ale vizualizaci nelze provést v reálném čase a software není závislý na platformě. Jde o sbírku softwarových nástrojů, které umožňují uživatelům shromažďovat, zpracovávat, archivovat a vizualizovat data ze sensorové sítě. Aktuální verze obsahuje: rekonfigurovatelné Microserver prostředí pro spouštění (mSEE), malou knihovnu provádějící zpracování signálu a algoritmů pro detekci událostí, rozšíření Excel 2003 (Senscel) pro import, vizualizaci a zpracování sensorických dat a rozhraní pro SQL server pro načtení dat do archivu a zpět, microserver interakce konzoly (mSIC) pro uživatele, aby mohli konfigurovat a kontrolovat microservery. Veškerý software je implementován v jazyce C# v rámci Visual Studio 2005 a .NET framework 2.0 ^[32].

Dokumentace k tomuto programu je dostupná z <http://research.microsoft.com/en-us/projects/msrsense/default.aspx>

5.2.9 MeshNetics WSN Monitor

MeshNetics WSN Monitor zobrazuje topologii sítě, data snímačů a kvalitu signálu mezi uzly. Automaticky generuje schémata topologie sítě, jakmile jsou přidány a detekovány uzly do systému. Tyto uzly jsou pak pravidelně sledovány a sensorová data jsou automaticky zobrazena v tabulkách a grafech. Je vybaven rámcem založeným na XML pro rychlé přizpůsobení uživatelského rozhraní a naměřených parametrů snímače ^[32].

5.2.10 GSN

GSN middleware software pro různé sensorové sítě. Usnadňuje prohlížení velkého množství dat, která jsou získávána z WSN a uložena ve formě číselných údajů v centrální základní stanici ^[32]. Dokumentace k tomuto programu je dostupná z <http://sourceforge.net/apps/trac/gsn/>

6 DOHLEDOVÉ CENTRUM – SNĚHOMĚRNÝ VAROVNÝ SYSTÉM

Dohledové centrum bylo vytvořeno jako webová aplikace pro uživatele využívající naměřená data z čidel senzorové sítě, která jsou uložena v databázi. Aplikace umožňuje několik možností prohlížení naměřených dat a následný export senzorových dat do formátů CSV a XML. Zároveň je aplikace určena k výpočtu vodní hodnoty sněhu na základě empirické rovnice. Výsledné vodní hodnoty sněhu je možné exportovat do CSV. Vytvořená aplikace dohledového centra – sněhoměrný varovný systém běží lokálně na katedře geoinformatiky.

6.1 Softwarové požadavky aplikace

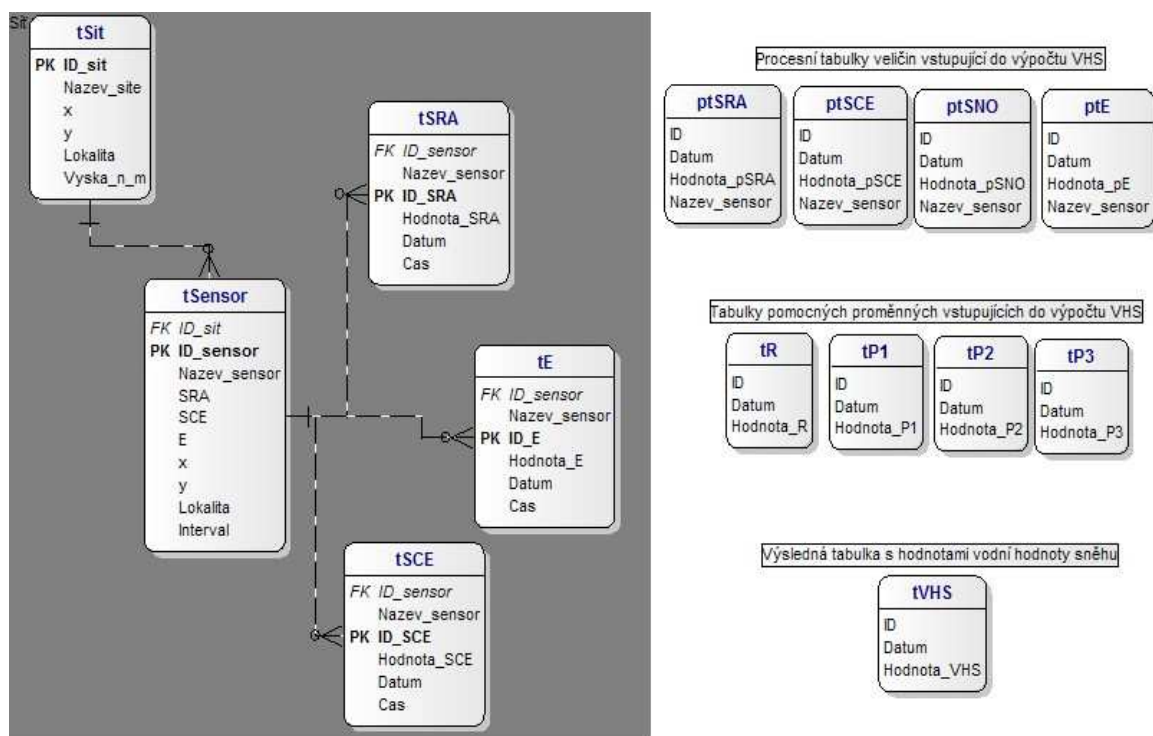
Vytvořená webová aplikace dohledového centra zpracovává data uložená v databázi PostgreSQL s prostorovou nadstavbou PostGIS. Data z databáze jsou volána pomocí PHP skriptů, které zajišťují funkčnost celé aplikace. K vytvoření aplikace byly využity tyto softwary:

- databáze PostgreSQL 9.0 s nadstavbou PostGIS 1.5 – úložiště dat
- pgAdminIII – administrační platforma pro PostgreSQL k vytvoření databázové struktury a správě dat
- skriptovací jazyk PHP verze 5.3.6
- Apache HTTP Server 2.2
- PSPad Editor – aplikace pro vytváření skriptů PHP a CSS
- DeZign for Databases 7.0.1 – aplikace pro navrhování databázové struktury

6.2 Návrh databázové struktury

Vytvoření databázové struktury tvořilo několik procesních kroků od prvotního návrhu na papír až po vytvoření struktury v administrační platformě pgAdminIII, která byla postupně testována do konečné podoby. Bylo potřeba vhodně navrhnout atributy jednotlivých tabulek včetně jejich datových typů.

Výsledná databázová struktura by se dala rozdělit do dvou částí. První část tvoří tabulky popisující senzorovou síť a v ní obsažené senzory včetně jejich naměřených hodnot. Druhá část je pak tvořena procesními tabulkami a tabulkami pomocných proměnných, které ukládají data vstupující do výpočtu vodní hodnoty sněhu, včetně tabulky s výslednými hodnotami. Konečný počet tabulek obsažených v databázi *vodni_hodnota_snehu* je čtrnáct.



Obr. 31: Databázová struktura aplikace dohledového centra

6.3 Tvorba a naplnění databázové struktury

Jak již bylo zmíněno výše, tvorba databázové struktury probíhala v PostgreSQL administrační platformě pgAdminIII. Prvním krokem bylo založení prázdné prostorové databáze s názvem *vodni_hodnota_snehu* a kódováním UTF-8. Ukázka SQL kódu vytvoření databáze:

```
sql16: CREATE DATABASE vodni_hodnota_snehu
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1;
```

Dalším krokem již bylo vytváření tabulek. Jako první byla vytvořena tabulka *tSit*, ve které jsou uloženy záznamy jednotlivých sensorových sítí. Tato tabulka je tvořena atributem *ID_sit*, jenž je zároveň primárním klíčem, dále pak atributy *Nazev_site*, *x*, *y*, *Lokalita* a *Vyska_n_m*. Atributy *x* a *y* obsahují prostorové souřadnice umístění sensorové sítě. Datové typy použité pro atributy tabulky jsou uvedeny v ukázce SQL kódu.

Ukázka SQL kódu vytvoření tabulky *tSit*:

```
sql17: CREATE TABLE "tSit" (
  "ID_sit" integer NOT NULL,
```

```

"Nazev_site" character(20),
x double precision,
y double precision,
"Lokalita" character(50),
"Vyska_n_m" integer,
CONSTRAINT "primarni klic1" PRIMARY KEY ("ID_sit")
)
WITH (OIDS=FALSE);

```

Jako druhá byla vytvořena tabulka *tSensor*, jenž obsahuje záznamy o senzorech. Tato tabulka je propojena s tabulkou *tSit* pomocí atributu *ID_sit*, jenž je v tabulce *tSensor* cizím klíčem. Dále pak tabulka obsahuje atributy *ID_sensor* (primární klíč), *Nazev_sensor*, *SRA*, *SCE*, *E*, *x*, *y*, *Lokalita* a *Interval*. Atributy *SRA*, *SCE* a *E* mají datový typ boolean a určují zda senzor měří srážky, sněh, nebo tlak. Atributy *x* a *y* obsahují prostorové souřadnice umístění senzoru a atribut *Interval* určuje časovou hodnotu, ve které senzor měří danou charakteristiku.

Ukázka SQL kódu vytvoření tabulky *tSensor*:

```

sql18: CREATE TABLE "tSensor" (
  "ID_sit" integer NOT NULL,
  "ID_sensor" integer NOT NULL,
  "Nazev_sensor" character(30),
  "SRA" boolean,
  "SCE" boolean,
  "E" boolean,
  x double precision,
  y double precision,
  "Lokalita" character(50),
  "Interval" interval,
  CONSTRAINT "primarni klic2" PRIMARY KEY ("ID_sensor"),
  CONSTRAINT "cizi klic1" FOREIGN KEY ("ID_sit")
    REFERENCES "tSit" ("ID_sit") MATCH FULL
    ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (OIDS=FALSE);

```

Do tabulek *tSit* a *tSensor* byla data vložena pomocí SQL příkazu *INSERT*. Ukázka SQL kódu vložení záznamů do tabulky *tSit*:


```

sql19: INSERT INTO "tSit"("ID_sit", "Nazev_site", x, y, "Lokalita",
"Vyska_n_m")
VALUES (1, 'SitVVHS1', 49.9999, 18.2222, 'Vysoké Pole', 422),
(2, 'SitVVHS2', 49.9999, 18.3322, 'Vysoké Pole', 432),
(3, 'REAL', 49.9999, 18.333, 'Vysoké Pole', 433);

```

Ukázka SQL kódu vložení záznamů do tabulky *tSensor*:

```

sql20: INSERT INTO "tSensor"("ID_sit", "ID_sensor", "Nazev_sensor", "SRA",
"SCE", "E", x, y, "Lokalita", "Interval")
VALUES (1, 11, 'SRA11', true, false, false, 49.9999,
18.7777, 'Vysoké Pole', '01:00:00'),
(1, 12, 'SCE12', false, true, false, 49.9999,
18.7887, 'Vysoké Pole', '01:00:00'),
(1, 13, 'E13', false, false, true, 49.9999,
18.8877, 'Vysoké Pole', '01:00:00'),
(2, 21, 'SRA21', true, false, false, 49.9999,
18.8888, 'Vysoké Pole', '01:00:00'),
(2, 22, 'SCE22', false, true, false, 49.9999,
18.8778, 'Vysoké Pole', '01:00:00'),
(2, 23, 'E23', false, false, true, 49.9999,
18.7788, 'Vysoké Pole', '01:00:00'),
(3, 31, 'FIEDLER', true, false, false, 49.9999,
18.9999, 'Vysoké Pole', '24:00:00'),
(3, 32, 'SNOW', false, true, false, 49.9999,
18.9988, 'Vysoké Pole', '24:00:00'),
(3, 33, 'METEO', false, false, true, 49.9999,
18.9977, 'Vysoké Pole', '00:10:00');

```

Jako třetí byly vytvořeny tabulky *tSRA*, *tSCE* a *tE*, které obsahují záznamy o naměřených hodnotách. Tyto tabulky jsou propojeny s tabulkou *tSensor* pomocí atributu *ID_sensor*, jenž je v těchto tabulkách cizím klíčem. Dalšími atributy jsou *Nazev_sensor*, *ID_SRA*(*SCE* nebo *E*), *Hodnota_SRA*(*SCE* nebo *E*), *Datum* a *Cas*. Atributy *ID_SRA*, *ID_SCE* a *ID_E* jsou ve svých tabulkách primárním klíčem. Naplnění tabulek bylo provedeno pomocí SQL příkazu *COPY*, který zkopíruje hodnoty z *.csv souboru do vytvořené tabulky. Následující ukázky SQL kódu byly použity k vytvoření a naplnění tabulky *tSRA*. Vytvoření a naplnění tabulek *tSCE* a *tE* používá stejný SQL kód, s tím rozdílem, že *SRA* je nahrazeno *SCE* nebo *E*. Ukázka SQL kódu vytvoření tabulky *tSRA*:

```

sql21: CREATE TABLE "tSRA" (
    "ID_sensor" integer NOT NULL,
    "Nazev_sensor" character(30),
    "ID_SRA" integer NOT NULL,
    "Hodnota_SRA" double precision,
    "Datum" date,
    "Cas" time without time zone,
    CONSTRAINT "primarni klic3" PRIMARY KEY ("ID_SRA"),
    CONSTRAINT "cizi klic2" FOREIGN KEY ("ID_sensor")
        REFERENCES "tSensor" ("ID_sensor") MATCH FULL
        ON UPDATE NO ACTION ON DELETE NO ACTION
)
WITH (OIDS=FALSE);

```

Ukázka SQL kódu naplnění tabulky *tSRA*:

```

sql22: COPY "tSRA"("ID_sensor", "Nazev_sensor", "ID_SRA", "Hodnota_SRA",
"Datum", "Cas")
FROM '/SRA.csv'
WITH DELIMITER ';'
CSV HEADER;

```

Jak bylo zmíněno výše v návrhu, těchto pět tabulek tvoří první část databázové struktury. Druhá část databázové struktury je tvořena procesními tabulkami, tabulkami pomocných proměnných a tabulkou s vypočítanými vodními hodnotami sněhu.

Procesní tabulky mají za úkol ukládat data, která vznikla zpracováním naměřených hodnot uložených v tabulkách *tSRA*, *tSCE* a *tE*, pro potřeby výpočtu vodní hodnoty sněhu. Jelikož některé senzory měří srážky v hodinových intervalech a pro potřeby výpočtu je nutná denní suma, ukládá procesní tabulka *ptSRA* hodnoty denních sum srážek. Procesní tabulka *ptSCE* ukládá poslední naměřenou hodnotu celkové výšky sněhu pro daný den, procesní tabulka *ptE* ukládá průměrnou hodnotu tlaku pro daný den a procesní tabulka *ptSNO* ukládá výšku nového sněhu pro daný den, tedy rozdíl celkové výšky daného dne a předchozího dne. Všechny tyto hodnoty jsou vázány vždy k určitému senzoru. Procesní tabulky obsahují atributy ID, Datum, Nazev_sensor a Hodnota_pSRA(SCE nebo E nebo SNO). Naplnění tabulek je prováděno během výpočtu vodní hodnoty sněhu. Následující ukázka SQL kódu byla použita k vytvoření procesní tabulky *ptSRA*. Vytvoření procesních tabulek *ptSCE*, *ptE* a *ptSNO* používá stejný SQL kód, s tím rozdílem, že SRA je nahrazeno SCE nebo E nebo SNO. Ukázka SQL kódu vytvoření tabulky *ptSRA*:

```

sql23: CREATE TABLE "ptSRA"(
    "ID" serial NOT NULL,

```

```

    "Datum" date,
    "Hodnota_pSRA" double precision,
    "Nazev_sensor" character(30)
)
WITH (OIDS=FALSE);

```

Jelikož ve výpočtu vodní hodnoty sněhu pro daný den hrají důležitou roli i některé pomocné proměnné s datem předchozího dne, bylo potřeba vytvořit tabulky pomocných proměnných, které uchovávají jejich hodnoty. Byly vytvořeny tabulky *tR*, *tP1*, *tP2* a *tP3*. Jednotlivé tabulky obsahují atributy ID, Datum a Hodnota_R(P1 nebo P2 nebo P3). Tabulky jsou opět naplňovány během výpočtu vodní hodnoty sněhu. Následující ukázka SQL kódu byla použita k vytvoření tabulky *tR*. Vytvoření tabulek *tP1*, *tP2* a *tP3* používá stejný SQL kód, s tím rozdílem, že R je nahrazeno P1 nebo P2 nebo P3. Ukázka SQL kódu vytvoření tabulky *tR*:

```

sql24: CREATE TABLE "tR" (
    "ID" serial NOT NULL,
    "Datum" date,
    "Hodnota_R" double precision
)
WITH (OIDS=FALSE);

```

Poslední tabulkou druhé části databázové struktury je tabulka *tVHS* uchovávající výsledné vypočítané vodní hodnoty sněhu. Tuto tabulku tvoří atributy ID, Datum a Hodnota_VHS. Ukázka SQL kódu vytvoření tabulky *tVHS*:

```

sql25: CREATE TABLE "tVHS" (
    "ID" serial NOT NULL,
    "Datum" date,
    "Hodnota_VHS" double precision
)
WITH (OIDS=FALSE);

```

6.4 Rovnice pro výpočet vodní hodnoty sněhu

Pro výpočet vodní hodnoty sněhu (SVHV) je v aplikaci využita empirická rovnice ^[2]. Tato rovnice je sestavena z několika dílčích rovnic tvořených z vstupních veličin a pomocných proměnných. Vstupní veličiny jsou:

- SRA - denní úhrn srážek [mm]

- SNO - výška nového sněhu [mm]
- SCE - celková výška sněhu [mm]
- E - průměrný denní tlak vodní páry [hPa]

Pomocné proměnné jsou:

- R - hustota sněhu ($SVHV_{-1}/SCE$) [%]
- RP - hustota sněhu [%]
- FR - funkce hustoty
- P1 - přírůstek ze sněhových srážek [mm]
- P2 - přírůstek z vodních srážek [mm]
- P3 - přírůstek z vlhkosti vzduchu [mm]

Výpočet je tvořen těmito rovnicemi:

$$SVHV = (SCE - SNO_{-1}) \cdot RP + P1_{-1} \quad \text{když } SCE > 0, \text{ jinak } SVHV = 0$$

$$RP = [R_{-1} + (P2_{-1} + P3_{-1}) / SCE_{-1}] \cdot [1 + (SCE_{-1} - SCE + SNO_{-1}) \cdot FR/SCE_{-1}]$$

když $SCE_{-1} > 0$ a $SCE - SNO_{-1} > 0$, jinak $RP = 0$

$$R = SVHV_{-1} / SCE \quad \text{když } SCE > 0, \text{ jinak } R = 0$$

$$FR = \exp^3 [-R / (90 - R)] \quad \text{když } R > 0, \text{ jinak } FR = 0$$

$$P1 = SRA \quad \text{když } SNO / SRA > 0,05$$

$$P1 = 20 \cdot SNO \text{ když } SNO / SRA \leq 0,05$$

$$P1 = 0 \quad \text{když } SRA = 0$$

$$P2 = \min (SRA - P1; SCE \cdot FR \cdot 10)$$

$$P3 = E / 2 \cdot FR \cdot SCE$$

Pozn.: Veličiny s indexem $_{-1}$ nesou hodnotu vztahující se k předchozímu dni.

6.5 PHP skripty a funkčnost aplikace

Webová aplikace dohledového centra obsahuje menu, jenž je tvořeno čtyřmi hlavními záložkami. První záložka a zároveň domovská stránka aplikace se nazývá *Rovnice VHS*. Zde je popsána empirická rovnice pro výpočet vodní hodnoty sněhu. Popsány jsou vstupní veličiny a pomocné proměnné, včetně jejich jednotek, a dílčí rovnice výpočtu.

Další záložka je nazvána *Prohlížení*. Na této záložce si lze vybrat ze tří způsobů. První způsob umožňuje prohlížení naměřených dat dle zvoleného senzoru a časového období. Potvrzením výběru se zobrazí tabulka obsahující atributy ID (identifikátor záznamu), ID_senzor, Nazev_senzor, Hodnota, Datum a Cas. Následně lze zobrazenou tabulku exportovat do souboru ve formátu CSV. Tento způsob je tvořen PHP skripty *prohlizeni1.php* a *export1.php*. Druhý způsob umožňuje vybrat z tabulek naměřených hodnot (*tSRA*, *tSCE*, *tE*) a následně vybranou tabulku exportovat do souboru ve formátu CSV nebo XML. Pro tento způsob byly vytvořeny skripty *prohlizeni2.php*, *exportCSV.php* a *exportXML.php*. Poslední způsob prohlížení umožňuje vybrat z formuláře senzorovou síť a následně pro tuto síť zobrazit tabulku se senzory obsaženými v této síti. Tuto tabulku lze exportovat do souboru ve formátu XML. Pro tento způsob prohlížení byly vytvořeny skripty *prohlizeni3.php* a *export3.php*.

Třetí záložka aplikace se nazývá *Výpočet*. Zde je možné vybrat ze dvou způsobů výpočtu vodní hodnoty sněhu. První způsob zobrazí čtyři formuláře pro výběr senzorů (senzor pro denní úhrn srážek, senzor pro celkovou výšku sněhu, senzor pro průměrný denní tlak vodní páry a senzor pro výšku nového sněhu) a formulář pro výběr data. Potvrzením výběru se spustí výpočet a zobrazí se hodnoty pro vstupní veličiny, pomocné proměnné a výsledná vodní hodnota sněhu. Druhý způsob výpočtu zobrazí již jen dva formuláře, a to pro výběr sítě vstupující do výpočtu a výběr data, pro které má být výsledná hodnota spočítána. Potvrzením výběru se opět spustí výpočet a zobrazí se hodnoty jako v předchozím způsobu. Nakonec lze exportovat tabulku s vypočítanými vodními hodnotami sněhu do souboru ve formátu CSV. Toto lze provést v obou případech. Pro tuto záložku byly vytvořeny skripty *vypocet1.php*, *vypocet2.php* a *exportVHS.php*.

Všechny PHP skripty vyžadující připojení k databázi využívají skript *pripojeni.php*, který je volán pomocí funkce `include`. Poslední záložka dohledového centra se nazývá *O aplikaci*. Zde lze najít popis aplikace.

Pozn.: Celá aplikace, včetně PHP skriptů a SQL kódů jsou obsaženy na CD přiloženém k magisterské práci.

7 DISKUZE

Vzhledem k stále častějšímu využívání senzorů a sensorových sítí a častým požadavkům uživatelů na zisk dat v reálném čase vzniká potřeba standardizace tohoto odvětví. Existuje rozličné množství sensorových systémů a senzorů, které komunikují většinou pomocí vlastních standardů vytvořených jejich výrobcem. Tím vznikají skupiny, které jsou využitelné pouze v určitých odvětvích.

Používání standardů iniciativy Sensor Web Enablement není dosud příliš rozšířeno, protože standardy obsažené v této iniciativě jsou charakteristické svou univerzálností pro různá data z různých sensorových systémů. Tím vzniká obrovská komplexnost a komplikovanost těchto standardů, a proto není jednoduché je v konečném důsledku v aplikacích implementovat. Na druhou stranu organizace OGC vytvořila několik služeb, které jsou dnes již běžně (standardně) využívány širokým spektrem uživatelů (např. GML, WMS, atd.) a tyto služby jsou kompatibilní se standardy iniciativy SWE. Do budoucna lze tedy očekávat, že standardy iniciativy SWE se s vývojem senzorů a sensorových systémů začnou hojněji využívat.

Klíčovou činností této magisterské práce bylo vytvoření návrhu a následná tvorba databázové struktury na platformě PostgreSQL pro uložení dat z katedrální sensorové sítě zaměřené na S-O monitoring a současně vytvoření dílčího prototypu kontrolního panelu. Největší problémy nastaly při vytváření dohledového centra pomocí skriptovacího jazyka PHP, který si sice velmi dobře rozumí s databázovým systémem PostgreSQL, ale dostupné dokumentace k jednotlivým funkcím jazyka ve spojení s PostgreSQL jsou velmi slabé. To je dáno malým rozšířením tohoto databázového systému v uživatelské rovině, jelikož většina běžných uživatelů databáze častěji používá systém MySQL, ke kterému existuje velké množství dokumentace a diskuzních fór. Jednotlivé PHP skripty se liší pouze v několika drobnostech jako jsou např. typ uvozovek nebo jiné, ale tyto drobnosti dokážou zabrat při vývoji bez potřebné dokumentace i několik dní.

Výsledné dohledové centrum mělo dle zadání obsahovat prostředí pro zjišťování nastavení a vzdálenou konfiguraci jednotlivých prvků katedrální sensorové sítě a pro import dat ze senzoru/primárního úložiště. Tato alternativa byla po domluvě s vedoucím práce vzhledem k náročnosti naprogramování a tím i časové náročnosti zrušena. V práci jsou však popsány alternativy, které mohou toto prostředí nahradit. Zjišťování nastavení a vzdálená konfigurace senzorů je proveditelná například pomocí webového rozhraní společnosti Fiedler-Mágr a import dat lze provést manuálním kopírováním souboru ve formátu CSV do předpřipravené databáze pomocí SQL příkazu COPY.

Uživatel vytvořeného dohledového centra, který si vyexportuje *.csv soubor se může při otevírání tohoto souboru v programu Microsoft Excel setkat s hlášením, že se jedná o soubor SYLK, který může být poškozen. Po stisknutí tlačítka OK se však data v programu zobrazí. K tomuto problému dochází, jelikož první dva znaky tabulky jsou velká písmena I a D.

8 ZÁVĚR

Tato magisterská práce obsahuje přehled a rozbor standardů využívaných pro práci se senzory a sensorovými systémy. Obsahuje obecný popis standardů, popis jejich rozhraní, parametrů a operací. V přehledu jsou zahrnuty standardy iniciativy Sensor Web Enablement, která je tvořena třemi standardy kódování pro popis senzorů a sensorových pozorování (SensorML, O&M a TML) a čtyřmi standardy definic rozhraní pro webové služby (SOS, SPS, SAS a WNS). Součástí přehledu je i interakce mezi SWE službami a SWE kódováními.

V přehledu jsou zahrnuty také další technologie, standardy a protokoly pro vytváření sensorových sítí, které nebyly vytvořeny mezinárodní standardizační organizací OGC, ale patří mezi významné a hojně využívané technologie. Do této skupiny patří 6LoWPAN, IEEE 802.15.4, ZigBee, JenNet, WirelessHART, ISA100.11a, Z-Wave, ANT, ONE-NET, DASH7, Wibree a Extended Environments Markup Language.

Další součástí přehledu jsou standardy pro přenos, uložení a filtrování dat, které byly také vytvořeny organizací OGC. Zde je popsán standard GML, jenž je určen pro přenos a uložení dat, standard Simple Features určený k uložení prostorových dat a standard Filter Encoding určený k filtrování dat. V kapitole zpracování proudů dat jsou popsány systémy řízení proudů dat a speciální architektury k jejich zpracování.

Součástí magisterské práce bylo testování několika aplikací určených pro senzory a sensorové sítě. V popisu těchto aplikací je podán přehled o jejich fungování, práci s daty, možnostech vizualizace a nastavení a následném převodu dat do vlastní databáze. Kromě testovaných aplikací jsou zde uvedeny i další existující softwary pro práci se sensorovými daty určené zejména k vizualizaci dat.

V závěru práce je pak popsáno vytvoření vlastního dohledového centra, které zahrnuje seznam softwarů použitých k jeho vytvoření, návrh, tvorbu a naplnění databázové struktury, popis rovnice pro výpočet vodní hodnoty sněhu a popis funkčnosti aplikace.

SUMMARY

This diploma thesis provides an overview and analysis of standards used for working with sensors and sensor systems. It contains a general description of standards, a description of their interfaces, parameters and operations. The overview included standards of Sensor Web Enablement initiative, that consists of three coding standards for describing sensors and sensor observations (SensorML, O & M and TML) and four standards for interface definitions of web services (SOS, SPS, SAS and WNS). In the overview is also interaction between SWE services and SWE encodings.

The overview also included other technologies, standards and protocols to create sensor networks that were not created by international standardization organization OGC, but they are important and widely used technologies. This group includes 6LoWPAN, IEEE 802.15.4, ZigBee, JenNet, WirelessHART, ISA100.11a, Z-Wave, ANT, ONE-NET, DASH7, Wibree and Extended Environments Markup Language.

Another part of the overview are standards for transmission, storage and filtering data, that were also created by OGC organization. There are described the standard GML, that is used for data transfer and storage, the standard Simple Features designed to store spatial data and the Filter Encoding designed to filter the data. In chapter streams of data processing are described streams of data management systems and special architectures to their processing.

Part of the diploma thesis was to test several applications for sensors and sensor networks. The description of these applications is an overview of their operation, working with data, visualization capabilities and settings and also about transfer data to own database. In addition to the tested applications other existing softwares to work with sensor data are listed here, mainly designed to visualize data.

In the end is described creating of own monitoring center, that includes a list of softwares used in its creation, design, making and fulfillment of the database structure, a description of the equation to calculate the water equivalent of snow and description of application functionality.

KEY WORDS: sensor, sensor network, standard, service, database, data, measuring, observations, transmission, storage

POUŽITÁ LITERATURA A INFORMAČNÍ ZDROJE

- [1] KRESSE, Wolfgang a Kian FADAIE. *ISO Standards for Geographic Information*. Germany: Springer-Verlag Berlin Heidelberg, 2004, 322 s. ISBN 3-540-20130-0.
- [2] NĚMEC, L. *Příprava map charakteristik zatížení sněhem*. In Sborník XI. konference Spolehlivost konstrukcí. Plzeň: DT Ostrava, 2010, ISBN 978-80-02-02234-3

Internetové zdroje:

- [3] BRÖRING, Arne, Johannes ECHTERHOFF, Simon JIRKA, Ingo SIMONIS, Thomas EVERDING, Christoph STASCH, Steve LIANG a Rob LEMMENS. *New Generation Sensor Web Enablement*. 2011. Dostupné z: <http://www.mdpi.com/1424-8220/11/3/2652/pdf>
- [4] CANNATA, M. a M. ANTONOVIC. WEBMGS 2010. *ISTSOS: INVESTIGATION OF THE SENSOR OBSERVATION SERVICE* [online]. 2010[cit. 2012-02-02]. Dostupné z: <http://www.isprs.org/proceedings/XXXVIII/4-W13/proceedings.html>
- [5] CGDI. *Specification of the Filter Encoding*. 2004. Dostupné z: http://www.geoconnections.org/architecture/technical/specifications/filter_encoding/filter_encoding.doc
- [6] Common Query Language. *CQL* [online]. 12.1.2004 [cit. 2012-02-02]. Dostupné z: <http://srw.cheshire3.org/cql/index.html>
- [7] DASH7. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 24.1.2012 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/DASH7>
- [8] ENVIRONMENTAL MEASURING SYSTEMS. *Datalogger MiniCube: Models VC, VV/VX, CC, VF User's Manual*. 2002. Dostupné z: http://www.emsbrno.cz/r.axd/pdf_v_MiniCube__userman_u_pdf.jpg?ver=
- [9] Extended Environments Markup Language. *EEML* [online]. 2007 [cit. 2012-02-02]. Dostupné z: <http://www.eeml.org/>
- [10] FIEDLER-MÁGR ELEKTRONIKA PRO EKOLOGII. *Webový prohlížeč měřených dat: uživatelská příručka*. 2008. Dostupné z: https://stanice.fiedler-magr.cz/pdf/manual_WebProhlizec.pdf
- [11] Geoinformatika - Geovědní faktografie. *Geoinformatika* [online]. 2006 [cit. 2012-02-02]. Dostupné z: <http://geologie.vsb.cz/geoinformatika/kap05.htm>
- [12] Geography Markup Language. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 27.1.2012 [cit. 2012-02-02]. Dostupné z: http://en.wikipedia.org/wiki/Geography_Markup_Language

- [13] GML Application Schemas. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 12.10.2011 [cit. 2012-02-02]. Dostupné z: http://en.wikipedia.org/wiki/GML_Application_Schemas
- [14] HOBO Event Data Logger for Tipping Rain Gauges and Industrial Counting. *MikroDaq.com* [online]. 2012 [cit. 2012-03-11]. Dostupné z: <http://www.microdaq.com/occ/other/event.php>
- [15] IEEE 802.15.4. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 23.11.2011 [cit. 2012-02-02]. Dostupné z: http://en.wikipedia.org/wiki/IEEE_802.15.4
- [16] Isa 100.11a. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 9.12.2011 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/Isa100.11a>
- [17] JenNet Protocol Stack. *Jennic Wireless Microcontrollers* [online]. 2012 [cit. 2012-02-02]. Dostupné z: http://www.jennic.com/products/protocol_stacks/jennet
- [18] KIMPL, Libor. *Prostorové nadstavby nekomerčních databází - vstup a správa geoobjektů*. Olomouc, 2010. Dostupné z: <http://www.geoinformatics.upol.cz/dprace/bakalarske/kimpl10/>. Bakalářská práce. Univerzita Palackého v Olomouci.
- [19] O&M. *Geostandards* [online]. 27.9.2009 [cit. 2012-02-02]. Dostupné z: http://geostandards.geonovum.nl/index.php/5.3.2_O%26M
- [20] OGC. *OpenGIS® Catalogue Services Specification* [online]. Douglas Nebert, Arliss Whiteside, Panagiotis (Peter) Vretanos. 2007 [cit. 2012-02-02]. 07-006r1. Dostupné z: <http://www.opengeospatial.org/standards/specifications/catalog>
- [21] OGC. *OpenGIS Filter Encoding 2.0 Encoding Standard* [online]. Peter Vretanos. 2010 [cit. 2012-02-02]. 09-026r1. Dostupné z: <http://www.opengeospatial.org/standards/filter>
- [22] OGC. *OpenGIS® Geography Markup Language (GML) Encoding Standard* [online]. Clemens Portele. 2007 [cit. 2012-02-02]. 07-036. Dostupné z: <http://www.opengeospatial.org/standards/gml>
- [23] OGC. *Sensor Alert Service: Best Practices* [online]. Ingo Simonis. 2006 [cit. 2012-02-02]. 06-028r3. Dostupné z: <http://www.opengeospatial.org/standards/bp>
- [24] OGC. *Sensor Model Language (SensorML) Implementation Specification* [online]. Mike Botts. 2007 [cit. 2012-02-02]. OGC 07-000. Dostupné z: <http://www.opengeospatial.org/standards/sensorml>
- [25] OGC. *Sensor Observation Service: Implementation Standard* [online]. Arthur Na, Mark Priest. 2007 [cit. 2012-02-02]. OGC 06-009r6. Dostupné z: <http://www.opengeospatial.org/standards/sos>

- [26] OGC. *Sensor Planning Service: Implementation Standard* [online]. Ingo Simonis, Johannes Echterhoff. 2007 [cit. 2012-02-02]. OGC 09-000. Dostupné z: <http://www.opengeospatial.org/standards/sps>
- [27] OGC. *Sensor Web Enablement Architecture: Best Practices* [online]. Ingo Simonis. 2008 [cit. 2012-02-02]. 06-021r4. Dostupné z: <http://www.opengeospatial.org/standards/bp>
- [28] OGC. *Sensor Web Enablement: Overview And High Level Architecture: White Papers* [online]. Mike Botts, George Percivall, Carl Reed, John Davidson. 2007 [cit. 2012-02-02]. OGC 07-165. Dostupné z: <http://www.opengeospatial.org/pressroom/papers>
- [29] OGC. *Transducer Markup Language (TML) Implementation Specification* [online]. Steve Havens. 2007 [cit. 2012-02-02]. OGC 06-010r6. Dostupné z: <http://www.opengeospatial.org/standards/tml>
- [30] OGC. *Web Notification Service: Discussion Paper* [online]. Ingo Simonis, Andreas Wytzisk. 2003 [cit. 2012-02-02]. 03-008r2. Dostupné z: <http://www.opengeospatial.org/projects/groups/sensorwebdwg>
- [31] ONE-NET. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 21.6.2011 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/ONE-NET>
- [32] PARBAT, Bhawana, A.K. DWIVEDI a O.P. VYAS. *Data Visualization Tools for WSNs: A Glimpse*. 2010. Dostupné z: <http://www.ijcaonline.org/volume2/number1/pxc387868.pdf>
- [33] POKORNÝ, Jaroslav. *Databázové architektury: současné trendy a jejich vztah k novým požadavkům praxe*. 2006. Dostupné z: <http://www.ksi.mff.cuni.cz/~pokorny/papers/MD06.pdf>
- [34] POKORNÝ, Jaroslav a Václav SNÁŠEL. *Zpracování proudů dat*. 2006. Dostupné z: <http://www.ksi.mff.cuni.cz/~pokorny/papers/DATAKON06.pdf>
- [35] ProWeatherStation Professional Data Logging Weather Station. *ProWeatherStation™* [online]. [cit. 2012-02-02]. Dostupné z: <http://proweatherstation.com/Products/products.htm>
- [36] SensorML. *Geostandards* [online]. 27.9.2009 [cit. 2012-02-02]. Dostupné z: http://geostandards.geonovum.nl/index.php/5.3.3_SensorML
- [37] Sensor Alert Service. *Geostandards* [online]. 27.9.2009 [cit. 2012-02-02]. Dostupné z: http://geostandards.geonovum.nl/index.php/5.4.3_Sensor_Alert_Service
- [38] Sensor Observation Service. *Geostandards* [online]. 27.9.2009 [cit. 2012-02-02]. Dostupné z: http://geostandards.geonovum.nl/index.php/5.4.1_Sensor_Observation_Service
- [39] Sensor Planning Service. *Geostandards* [online]. 27.9.2009 [cit. 2012-02-02]. Dostupné z: http://geostandards.geonovum.nl/index.php/5.4.2_Sensor_Planning_Service

- [40] Sensor Web. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 29.10.2011 [cit. 2012-02-02]. Dostupné z: http://en.wikipedia.org/wiki/Sensor_web
- [41] Standardy OGC: Observation and Measurements. *CAGI Český rozcestník k informacím o GIS a možnostech vzdělávání* [online]. 2011 [cit. 2012-02-02]. Dostupné z: http://gislib.upol.cz/educagi/index.php?option=com_content&view=article&id=88:standa&catid=49:normy-zakony&Itemid=74
- [42] Standardy OGC: Sensor Planning Service. *CAGI Český rozcestník k informacím o GIS a možnostech vzdělávání* [online]. 2011 [cit. 2012-02-02]. Dostupné z: http://gislib.upol.cz/educagi/index.php?option=com_content&view=article&id=88:standa&catid=49:normy-zakony&Itemid=74
- [43] What is ant. *ANT the power of less* [online]. 2011 [cit. 2012-02-02]. Dostupné z: <http://www.thisisant.com/pages/technology/what-is-ant>
- [44] What is ant plus. *ANT the power of less* [online]. 2011 [cit. 2012-02-02]. Dostupné z: <http://www.thisisant.com/pages/technology/what-is-ant-plus>
- [45] Wibree - Ultra Low Power (ULP) Bluetooth. *Electronics Weekly* [online]. 17.7.2007 [cit. 2012-02-02]. Dostupné z: <http://www.electronicsworld.com/Articles/11/07/2008/41819/wibree-ultra-low-power-ulp-bluetooth.htm>
- [46] WirelessHART. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 26.10.2011 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/WirelessHART>
- [47] Wireless Sensor Network Development Kits: Starter Kit. *MEMSIC* [online]. 2010 [cit. 2012-02-02]. Dostupné z: <http://www.memsic.com/products/wireless-sensor-networks/development-kits.html>
- [48] ZigBee. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 9.7.2011 [cit. 2012-02-02]. Dostupné z: <http://cs.wikipedia.org/wiki/ZigBee>
- [49] Z-Wave. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 16.1.2012 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/Z-wave>
- [50] 6LoWPAN. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 22.1.2012 [cit. 2012-02-02]. Dostupné z: <http://en.wikipedia.org/wiki/6LoWPAN>

PŘÍLOHY

SEZNAM PŘÍLOH

Příloha 1.: Dohledové centrum – sněhoměrný varovný systém (aplikace na CD)