



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

NÁVRH SOFTWAREOVÉHO PROSTŘEDÍ PRO ZPRACOVÁNÍ DAT Z MĚŘENÍ TELEKOMUNIKAČNÍCH SÍTÍ

DESIGN OF SOFTWARE ENVIRONMENT FOR PROCESSING DATA FROM
TELECOMMUNICATION NETWORK MEASUREMENT

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. JAN FORNŮSEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PAVEL REICHERT

BRNO 2012



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Jan Fornůsek

ID: 109648

Ročník: 2

Akademický rok: 2011/2012

NÁZEV TÉMATU:

Návrh softwarového prostředí pro zpracování dat z měření telekomunikačních sítí

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte a popište problematiku měření optických sítí. Na základě získaných poznatků navrhnete a následně zrealizujete aplikaci pro zpracování výsledků měření. Základními požadavky na aplikaci je její přenositelnost, komplexnost (typově různé měření v jediném vyhodnocení), funkčnost (vyhodnocovací, limity, export/tisk), jednoduchost a přehlednost.

DOPORUČENÁ LITERATURA:

[1] KUCHARSKI, M., DUBSKÝ, P.: Měření přenosových parametrů optických vláken, kabelů a tras, MIKROKOM, Praha 1998

[2] MARMANIS, H. BABENKO, D. Algorithms of the intelligent web, Manning Publications, 2009. ISBN 978-1-933988-66-5

[3] KOSEK, J. PHP tvorba interaktivních internetových aplikací, Grada, Praha, 1999.

Termín zadání: 6.2.2012

Termín odevzdání: 24.5.2012

Vedoucí práce: Ing. Pavel Reichert

Konzultanti diplomové práce:

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ANOTACE

Tento dokument seznamuje čtenáře s měřeními souvisejícími s optickými sítěmi. Nejedná se o výčet všech měření. Jsou zmíněny pouze vybraná měření související se vstupními soubory do vyvíjené aplikace. Praktickou část popisuje dokumentace k aplikaci.

Klíčová slova: útlum, RFC 2544, Java, grafické uživatelské rozhraní, komponenta

ABSTRACT

This document introduces the reader with measurements related to optical networks. This is not a list of all measurements. There are only mentioned selected measurements related to input files into the developed application. The practical part describes the application's documentation.

Keywords: loss, RFC 2544, Java, graphical user interface, component

Bibliografická citace díla:

FORNŮSEK, J. *Návrh softwarového prostředí pro zpracování dat z měření telekomunikačních sítí*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 38 s. Vedoucí diplomové práce Ing. Pavel Reichert.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma Návrh softwarového prostředí pro zpracování dat z měření telekomunikačních sítí jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením tohoto semestrálního projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu práce Ing. Pavlovi Reichertovi za účinnou metodickou pomoc a cenné rady při zpracování diplomové práce.

V Brně dne

.....

podpis autora

Výzkum popsany v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Obsah

Úvod	5
1 MĚŘENÍ OPTICKÝCH SÍTÍ	6
1.1 Fyzická vrstva	6
1.1.1 Přímá (transmisní) metoda	6
1.1.2 Metoda OTDR	7
1.2 Vyšší vrstvy	8
1.2.1 RFC 2544	9
1.2.2 EtherSAM	14
2 PROGRAM	21
2.1 Vývojová část	21
2.1.1 Java	21
2.1.2 NetBeans	24
2.1.3 Aplikace	25
2.2 Uživatelská část	29
2.2.1 Přímá metoda	31
2.2.2 RFC 2544 test	33
Závěr	36
Literatura	37
Seznam zkratk a symbolů	38

ÚVOD

V současné době jsou prakticky veškeré dálkové komunikační okruhy a rozlehlé sítě postaveny na optické infrastruktuře. V optických sítích nedochází k tak vysokým ztrátám jako v klasických sítích, přesto však nejsou zanedbatelné. Kromě velké přenosové rychlosti je další významnou výhodou optických vláken jejich naprostá necitlivost vůči elektromagnetickému rušení. [4]

Tato diplomová práce pojednává o problematice měření optických telekomunikačních sítí. Text práce je členěn do tří základních částí. První kapitola představuje rozbor měření a testů vykonávaných na optických sítích. Ve druhé kapitole jsou zmíněny informace o vytvořené aplikaci. Jedna část této kapitoly popisuje vývoj aplikace, druhá část obsahuje jednoduchý návod, kterým jsou zároveň popsány funkce aplikace. Závěrem je uvedeno zhodnocení práce.

1 MĚŘENÍ OPTICKÝCH SÍTÍ

Od doby vzniku technologie optických vláken uplynulo desítky let. Vzhledem ke svým nárokům se optická vlákna zprvu používala jen pro podmořské spoje a velké páteřní sítě. Dnes je již situace jiná. Vzhledem k velké přenosové kapacitě optického vlákna, která je mnohonásobně vyšší než u metalických vedení, se poskytovatelům síťových služeb vyplatí výstavba optických tras do hustě zabydlených oblastí. V poslední době se vývoj snaží kromě skleněných vláken zefektivnit přenosové vlastnosti plastových vláken, které by vzhledem k nižším nákladům našli uplatnění v koncových sítích. [1]

Vývoj optoelektroniky současně vyžaduje také vývoj měřících metod a přístrojů pro měření parametrů optických prvků.

1.1 Fyzická vrstva

Na optických vláknech se provádí mnoho měření. Pro nás je zajímavé měření nejdůležitějšího přenosového parametru, útlumu. Útlum optického vlákna mezi dvěma body určíme dle vztahu

$$A = 10 \log \frac{P_1}{P_2} \quad [\text{dB}], \quad (1.1)$$

kde P_1 a P_2 jsou optické výkony při vlnové délce λ .

V závislosti na délce trasy lze definovat měrný útlum vlákna

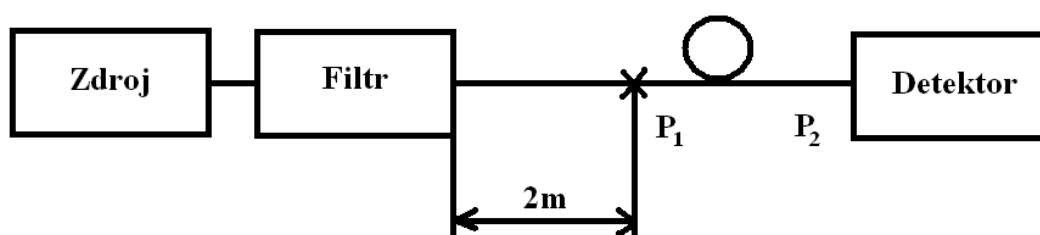
$$\alpha = \frac{A}{l} \quad [\text{dB.km}^{-1}], \quad (1.2)$$

kde l je vzdálenost mezi začátkem a koncem měřeného vlákna. Pro měření útlumu se využívá několika měřících metod, mezi nejpoužívanější patří přímá metoda a metoda OTDR. [1]

1.1.1 Přímá (transmisní) metoda

Měření přímou metodou spočívá v použití dvou samostatných přístrojů, umístěných na obou koncích měřeného úseku. První přístroj má za úkol vytvářet

optické záření, chová se tedy jako jeho zdroj. Druhý přístroj měří výkon tohoto záření. Jelikož přenos probíhá v obou směrech, je vyžadováno také obousměrné měření, což znamená, že po měření se musí přístroje vyměnit a provést měření z druhého konce. Některé měřicí soupravy jsou proto vybaveny zdrojem i detektorem na obou stranách. Metoda je vhodná pro určení celkového útlumu trasy, zjištění ohybů, nebo přerušení ve vlákne. Měřicí soustava je zapojena podle obr. 1.1.



Obr. 1.1: Přímá metoda

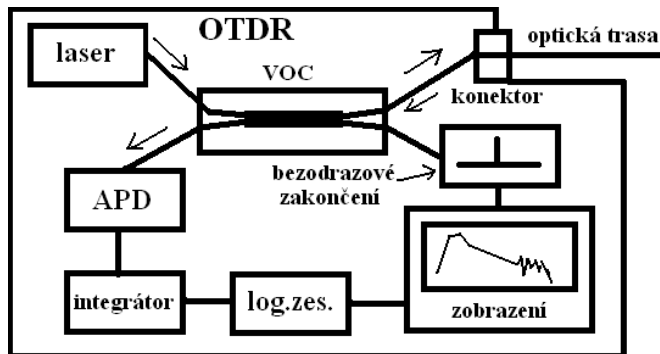
V prvním kroku je změřen výstupní výkon P_2 signálu pro průchodu vláknem o délce l . V druhém kroku se vlákno asi 2metry od zdroje zlomí a připojením detektoru změříme vstupní výkon P_1 . Potom je dle vztahu (1.1) stanoven útlum vlákna.

Existuje také modifikace transmisní metody, při které nedochází k lámání vlákna, a není tedy destruktivní. Této metodě se říká OLTS, neboli metoda vložného útlumu. Jedná se o metodu provozní a je vhodná pro vlákna a kabely opatřené konektory. Nejprve je nutno přístroj kalibrovat přímým propojením zdroje a detektoru, čímž změříme P_1 . Následně mezi zdroj a detektor zapojíme měřené vlákno a získáme P_2 . Výsledný útlum a měrný útlum se vypočítá dle (1.1) a (1.2). [5, 6]

1.1.2 Metoda OTDR

Někdy označovaná jako metoda zpětného rozptylu, nebo metoda optické reflektometrie v časové oblasti, je měřicí metodou využívající odrazu části optického signálu při průchodu světlovodem. Mezi využívané odrazy patří Rayleighův rozptyl, pomocí něhož sledujeme postupnou ztrátu energie způsobenou útlumem vlákna a Fresnelův rozptyl, pomocí kterého jsme schopni lokalizovat sváry, konektory a

přerušeni vlákna. Výhodou této metody je také, že se měřící soupravu tvoří pouze jediné zařízení. Měřicí přístroj se nazývá reflektometr.



Obr. 1.2: Blokové schéma reflektometru

Zařízení funguje tak, že laserem je vyslán světelný paprsek do vlákna prostřednictvím směrového vazebního článku (VOČ). Zpětně odražené nebo rozptýlené světlo z vlákna je detekováno lavinovou fotodiodou APD. Detekovaný elektrický signál je veden do integrátoru, který má za úkol odstranit nadbytečný šum. Dále je signál logaritmován. Výsledné hodnoty jsou zobrazeny na obrazovce. [1,6]

1.2 Vyšší vrstvy

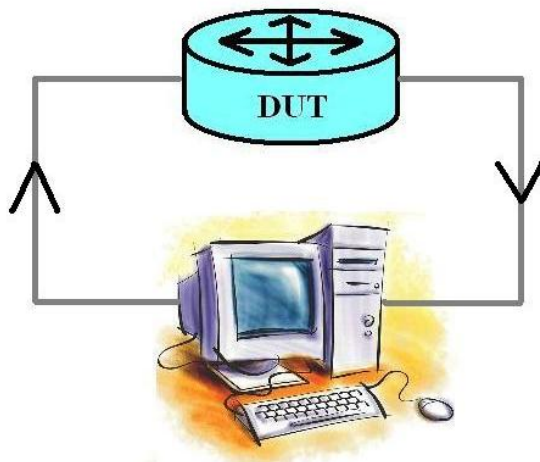
V dnešní době existuje mnoho metod pro měření síťových prvků nebo sítí jako celku. Pro tato měření se používají klientské aplikace nebo specializovaná hardwarová zařízení, která obvykle umožňují širší možnosti a větší míru přesnosti. Protože se tyto prostředky mohou ve svých možnostech lišit, byly vydány určité doporučení, které by měly testy splňovat. Tyto testy jsou v dnešní době považovány za standard v této oblasti. Následující vybrané testy patří mezi nejpoužívanější.

1.2.1 RFC 2544

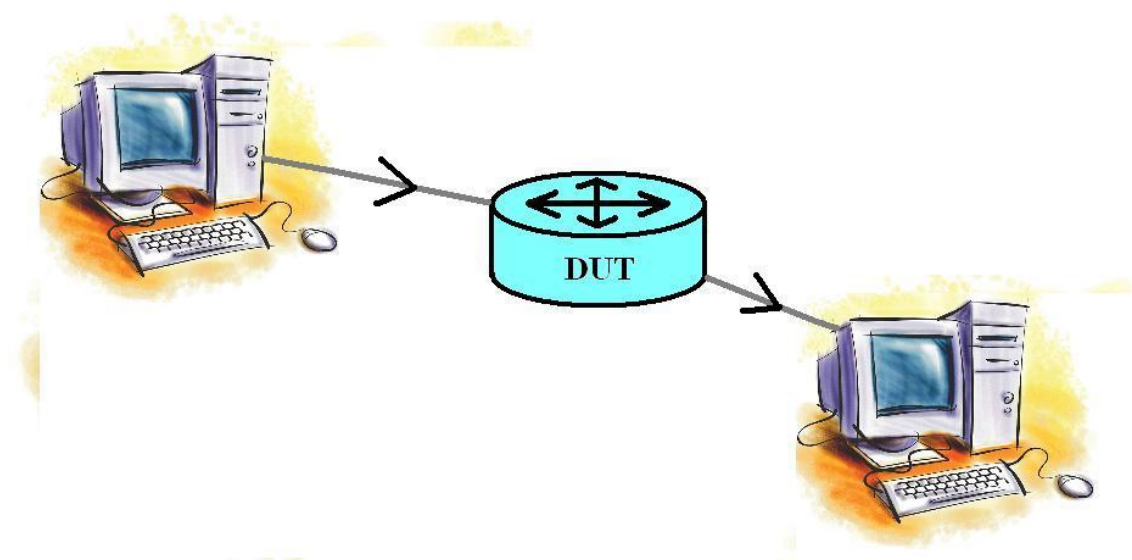
Metodologie měření pro síťově propojené zařízení, tak zní český překlad názvu dokumentu pracovní skupiny BMWG, který vydala toto doporučení v roce 1999 jako náhradu za starší RFC 1944. Obsahuje popis výkonnostních srovnávacích testů včetně jednoznačného formátu pro reprezentaci výsledků. To umožňuje jednoduše porovnávat zařízení od různých výrobců.

V dokumentu se rozlišuje mezi výrazy „musí, měl by a volitelné“. Podle míry jejich splnění pak testy úplně nebo částečně odpovídají specifikacím tohoto dokumentu.

Při zapojení měřicího zařízení a testovaného zařízení (DUT) se nám naskýtají dvě možnosti. Buď může měřicí zařízení obsahovat jak vstupní port pro vysílaná testovací data, tak i výstupní port pro přijímaná testovací data, nebo je měřicí zařízení rozděleno na vysílací a přijímací část. V tomto případě však musíme přenášet stavové informace. Na základě přijatých dat a jejich porovnáním s daty vyslanými jsme pak schopni určit jaký výkon dané zařízení má.



Obr. 1.3: Vstupní i výstupní port v jednom zařízení



Obr. 1.4: Vstupní a výstupní port jsou odděleny

Při nastavení testovaného zařízení se očekává, že budou všechny podporované protokoly nakonfigurovány a během testů se nebudou měnit.

Dalším tématem je velikost použitých rámců. Všechny testy by měly být spuštěny nejméně pro 5 různých velikostí rámců. V testu by měly být zahrnuty rámce s nejmenší a největší možnou velikostí pro dané médium. Doporučené velikosti rámců pro různá média jsou uvedeny v příloze standardu. Pro Ethernet např. jsou doporučené velikosti rámců 64, 128, 256, 512, 1024, 1280 a 1518 bajtů. Ke každé technologii (Ethernet, Token ring, FDDI) je uvedena skladba paketu a vzhledem k velikosti rámce také paketová rychlost (pps).

Je nutné kontrolovat, zdali mezi přijatými testovacími rámci nejsou také jiné, např. směrovací informace. Tyto se pak do testu nesmí počítat.

Při testování více propojených zařízení je na všechny pohlíženo jako na jediné zařízení. V tomto případě musíme brát v úvahu, že některá zařízení mohou zvyšovat zpoždění.

Každý konkrétní test se skládá z několika procesů. Každý proces vrací jeden údaj, např. ztrátovost při určité vstupní paketové rychlosti. Proces se skládá z těchto částí:

- Jestli je testovací zařízení směrovač, pošleme se směrovací informace na jeho vstupní port a počkáme 2 sekundy pro jistotu, aby směrovač stihl informace zpracovat
- Pošleme „učící rámce“ na výstupní port a čekáme 2 sekundy na zpracování. Tyto rámce naučí zařízení kam posílat přijaté testovací rámce
- Spustíme test, který by měl trvat minimálně 60 sekund
- Počkáme 2 sekundy, než jsou přijaty všechny rámce
- Dále počkáme alespoň 5 sekund na stabilizaci testovaného zařízení

Testem jsou zjišťovány následující parametry:

Propustnost

Propustnost je nejvyšší rychlost, při které nejsou žádné odeslané rámce zahozeny. RFC 2544 doporučuje test provést podle následujícího postupu. Poslat určitý počet rámců určitou rychlostí na testované zařízení a počítat rámce přijaté z testovacího zařízení. Jestliže počet vyslaných rámců je stejný jako počet přijatých rámců, rychlost vysílání je zvýšena a test se opakuje. Jestliže je přijato méně rámců, než bylo vysláno, rychlost je naopak snížena a test se opakuje.

Výsledky testu by měly být interpretovány ve formě grafu. Na ose x jsou velikosti rámců, na ose y je rychlost. V grafu by měly být vyneseny alespoň dvě křivky. Jedna bude znázorňovat teoretickou rychlost pro dané médium v závislosti na velikosti rámce. Druhá křivka by měla zobrazovat výsledky testu. Přidáním dalších průběhů do grafu je možno zobrazit výsledky jiného typu testovacích dat.

Zpoždění

Zpoždění, neboli latence, je rozdíl času přijetí první bitu rámce a času odeslání posledního bitu rámce s identifikační značkou. Poté co je zjištěna propustnost pro různé velikosti rámců, je vytvořen tok rámců dané velikosti protékající skrz testovací zařízení stanovenou propustností. Doba vysílání by měla trvat alespoň 120 sekund. Po 60 sekundách by měla být do jednoho z rámců zahrnuta identifikační značka. Čas, kdy je tento rámeček plně vyslán je zaznamenán. Přijímací logika v testovacím zařízení musí rozpoznat rámeček s identifikační značkou a zaznamenat čas kdy k tomu došlo. Tento test

musí být opakován minimálně 20 krát a výslednou hodnotu potom určíme průměrem hodnot dílčích testů. Ve výsledku testu musí být uvedeno, která definice zpoždění z [2] byla při testu použita. Výsledky testu by měly být zobrazeny ve formě tabulky, ve které každý řádek představuje velikost testovacího rámce a ve sloupcích jsou uvedeny rychlost zasilání rámců, typ použitého média a výsledná hodnota zpoždění.

Ztrátovost

Ztrátovost je procentuální údaj vyjadřující množství rámců, které nebyly přijaty z důvodu nedostatku systémových požadavků. Test se provede posláním určitého počtu rámců určité velikosti skrz testované zařízení a počítají se přijaté rámce. Ztrátovost se pak vypočítá dle vzorce:

$$ztrátovost = \frac{počet_{vstup} - počet_{výstup}}{počet_{vstup}} * 100 \text{ [%]} \quad (1.3)$$

$počet_{vstup}$ – počet rámců vyslaných na vstup testovacího zařízení

$počet_{výstup}$ – počet rámců přijatých z výstupu testovacího zařízení

První test by se měl provést maximální (100%) rychlostí pro dané médium. Další testy jsou provedeny, v intervalu 10%, s nižší rychlostí. Testy provádíme do té doby, než dosáhneme ve dvou testech žádné ztracené rámce. Krok 10% mezi testy je maximální, je možnost volit krok mezi testy menší.

Výsledky testu ztrátovosti by měly být zobrazeny ve formě grafu. Na ose x musí být vyneseno procentuální vyjádření rámcové rychlosti vzhledem k teoretické rychlosti použitého média. Na ose y musí být vynesena ztrátovost v procentech. V počátku osy x a osy y musí být hodnota 0%, na konci obou os musí být hodnota 100%. V grafu může být vyneseno více křivek pro různé velikosti rámců, protokoly a typy přenášených dat.

Back-to-back

Hodnota back-to-back vyjadřuje počet rámců v nejdelším shluku, které testované zařízení dokáže zpracovat bez ztráty jakéhokoliv rámce.

Doba testu musí být nejméně 2 sekundy a měl by být 50 krát opakován, výsledná hodnota je pak dána průměrem hodnot jednotlivých opakování.

Výsledky testu by měly být zobrazeny ve formě tabulky s řádky pro každou velikost testovacího rámce. Sloupce by pak měly obsahovat výsledný průměrný počet rámců pro každý typ testovacích dat.

Obnovení systému

Cílem testu je charakterizovat rychlost, za kterou se testované zařízení zotaví z přetížení. V průběhu testu se odešlou rámce rychlostí 110% ze změřené propustnosti nebo maximální rychlosti pro dané médium, po dobu nejméně 60 sekund. Následně je uložen čas a zároveň rychlost redukována na 50% ze zadané a poté zaznamenán čas ztráty posledního rámce. Rozdílem těchto časů je určena obnova systému. Test by měl být několikrát opakován, výsledek pak bude zprůměrovaná hodnota ze všech testů.

Výsledky by měly být vyneseny do tabulky, kde řádky budou reprezentovat jednotlivé velikosti testovacích rámců a sloupce pak budou obsahovat rámce velikost rámce, propustnost pro každý typ datového toku a čas nutný pro zotavení se.

Restart

Cílem testu je zjistit rychlost s jakou se testované zařízení zotaví z hardwarového či softwarového restartu. Pošleme nepřetržitý tok rámců rychlostí, která odpovídá propustnosti pro minimální velikost testovacího rámce. Provedeme restart testovaného zařízení a sledujeme výstupní port. Zaznamenáme čas, kdy byl přijat poslední rámeček před restartem a první rámeček po restartu. Výsledkem je rozdíl těchto časů. Pokud chceme vyvolat restart přerušením napájení, měl by výpadek trvat alespoň 10 sekund.

Výsledek by měl být formou jednoduché sady údajů, jeden pro každý typ restartu. [2]

1.2.2 EtherSAM

Zatímco se Ethernet vyvíjí jako transportní technologie, sítě přesunuly svoje zaměření z prostého přesunu dat na poskytování zábavy a nových aplikací. Služby založené na Ethernetu, jako jsou mobilní páteřní sítě, obchodní a velkoobchodní služby, musí poskytovat širokou škálu aplikací jako hlas, video, e-mail, online obchodování a jiné. Tyto nové aplikace kladou nové požadavky na výkon sítě a metodik používaných k ověření výkonu těchto služeb.

Tuto část popisuje EtherSAM, neboli ITU-T Y.1564, což je nový ITU-T standard pro instalaci a odstraňování problémů Ethernetových služeb. EtherSAM je jediná standardní testovací metoda, která umožňuje kompletní ověření služeb dohodnuté úrovně (SLA) v jednom, výrazně rychlejším testu a s vyšším stupněm přesnosti.

Ethernetové sítě nyní poskytují citlivé služby a služby v reálném čase. U služeb máme na mysli různé typy provozu, které je síť schopna přenášet. Obecně lze veškerý síťový provoz klasifikovat do tří typů: datový, v reálném čase a s vysokou prioritou. Každý z těchto typů je ovlivněn rozdílně podle charakteru sítě a musí být upraven na takový tvar, aby byly splněny jeho minimální výkonnostní cíle.

Tab. 1.1: Aplikace a služby v závislosti na typu provozu

Typ provozu	Hlavní aplikace	Příklady služeb
Data	Ne v reálném čase nebo přenos dat	Data, přístup na internet, FTP down/up, server
Real-time	Real-time vysílání, které po ztrátě nemůže být obnoveno	VoIP, IPTV, online rádio, TV, internetové hry a videokonference
Vysoká priorita	Povinný provoz zajišťující stabilitu v síti	OAM rámce, přepínání/směrování kontrolních rámců

Pro zajištění kvality služeb (QoS), musí poskytovatelé správně nakonfigurovat své sítě, aby zvládali upřednostnění provozu v síti. Toho je dosaženo tím, že se přiřadí

různé stupně priority pro každý typ služby a přesně nakonfigurují algoritmy zajišťující přidělování priorit.

Smlouva o poskytování služeb (SLA) je závazná smlouva mezi poskytovatelem služeb a zákazníkem, kterému je garantováno zajištění minimální úrovně poskytovaných služeb. Tato SLA určuje klíčové charakteristiky a jejich garantované minimální výkony.

Tab. 1.2: Třídy a KPI v závislosti na typu provozu

Typ provozu	Real-time data	Vysoko-prioritní data	Best-efford data (přístup k internetu)
CIR (Mb/s) zelený provoz	5	10	2.5
EIR (Mb/s) žlutý provoz	0	5	5
Zpoždění (ms)	<5	5-15	<30
Jitter (ms)	<1	-	-
Ztrátovost (%)	<0.001	<0.05	<0.05
VLAN	100	200	300

Provoz k zákazníkovi je klasifikován do tří tříd a každému je přidělena konkrétní barva. Zelená pro dohodnutý provoz, žlutá pro nadměrný provoz a červená pro vyřazený provoz.

Dohodnutá rychlost (CIR) - odkazuje na šířku pásma, která je zaručena být vždy k dispozici pro dané služby. Pro tento provoz jsou dány minimální výkonnostní cíle, které musí být garantované.

Nadměrná rychlost (EIR) - odkazuje na větší šířku pásma než CIR, které může být použito v závislosti na načítání a použití sítě.

Vyřazený provoz - jedná se o provoz, který přesahuje CIR nebo CIR/EIR rychlost, a který nemůže pokračovat bez přerušení ostatních služeb.

Tab. 1.3: Záruky provozu a KPI

Zelený provoz	0 - CIR	Garantovaný přenos	Garantovány KPI
Žlutý provoz	CIR - EIR	Best effort	KPI nejsou garantovány
Červený provoz	>EIR nebo CIR	Zrušený provoz	Nevztahuje se

Klíčové výkonnostní indikátory (KPI)

KPI jsou specifické provozní charakteristiky, které vyjadřují minimální výkon pro určité nastavení. Je-li splněna podmínka CIR, musí být splněny minimální požadavky pro veškerý provoz. Mezi KPI indikátory patří:

ŠÍŘKA PÁSMA

Šířka pásma určuje maximální množství dat, které může být přeneseno za dobu jedné sekundy. Šířka pásma musí být kontrolována, protože více služeb obvykle sdílí stejnou přenosovou linku. Proto musí být každá služba omezena, aby nebyly narušeny ostatní služby. Generování provozu nad šířkou pásma vede obvykle ke ztrátě rámců, nebo k výpadku služby.

ZPOŽDĚNÍ

Zpoždění, neboli latence je měření časové prodlevy mezi paketem vyslaným a přijatým. Obvykle se jedná o obousměrné měření. Měření je důležité pro hlasové služby, u kterých může příliš velké zpoždění ovlivnit kvalitu hovoru, což uživatel vnímá jako ozvěny, nesouvislou konverzaci nebo dokonce ztrátu spojení.

ZTRÁTOVOST

Ke ztrátě rámce může dojít z různých důvodů, jako jsou chyby v přenosu, nebo přetížení sítě. K chybě při přenosu může dojít v síťových zařízeních, jako jsou prepínače či směrovače, které jsou založené na porovnání kontrolní sekvence. Síťová zařízení musí zahazovat rámce také, aby nedošlo k saturaci linky v přetížených podmínkách.

JITTER

Ke kolísání zpoždění, neboli jitteru, dojde, pokud se liší zpoždění paketů průchodem sítí. Jak pakety putují sítí, často čekají ve frontách, odkud jsou posílány ve shlcích k dalšímu uzlu. Může dojít k náhodnému výběru priorit, což má za následek přenos paketu náhodnou rychlostí. Pakety jsou tedy obdrženy v náhodných intervalech. Toto kolísání se promítá ve vyrovnávací paměti koncových uzlů, kde tyto paměti mohou být nadměrně, nebo nedostatečně využity tam, kde jsou velké výkyvy zpoždění.

Aplikace v reálném čase, jako hlas a video, jsou obzvláště citlivé na paketový jitter. Buffery jsou navrženy tak, aby zachovávaly určité množství hlasových a obrazových paketů, které jsou poté zpracovány v pravidelných intervalech, aby poskytl plynulý a bezchybný přenos pro koncového uživatele.

Chceme-li vyřešit problémy se stávajícími metodikami, ITU-T představil nový standard: ITU-T Y.1564, který je v souladu s dnešními požadavky na Ethernetové služby. EtherSAM umožňuje kompletní ověření všech SLA parametrů v jediném testu, aby bylo zajištěno optimální QoS. Ve skutečnosti EtherSAM simuluje všechny typy služeb, které běží v síti a současně splňují všechny klíčové SLA parametry pro každou z těchto služeb. Nabízí také obousměrné měření. EtherSAM je založen na principu, že většina služeb se vyskytují ve dvou kategoriích

- a) v konfiguraci síťových prvků, které nesou služby
- b) při výkonu sítě za vysoké zátěže, kdy více služeb způsobí přetížení

Nastavení služeb

Zařízení jako jsou přepínače, směrovače, mosty a jednotky síťového rozhraní jsou základem každé sítě propojující více segmentů. Tato zařízení musí být správně nastavena, aby bylo zajištěno, že provoz bude odpovídat úrovni svých služeb. Pokud není služba správně nastavena, může být výkon sítě ovlivněn a služba nemusí být správně prováděna. To může vést k výpadku linky a celé řadě problémů, jako jsou zpoždění a selhání linky. Proto je velmi důležitou součástí testování, snaha zajistit, aby bylo zařízení správně nastaveno pro zpracování síťového provozu, jak bylo zamýšleno.

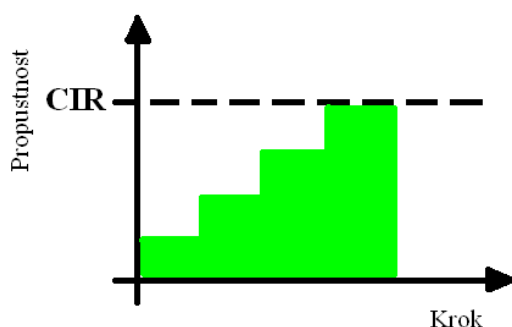
Testy a podtesty

Test nastavení služeb

Tento test ověřuje šířku pásma a požadavky na speciální služby definované uživatelem. Tento proces zahrnuje následující tři klíčové fáze a sleduje všechny ukazatele, aby se zajistilo, že jsou současně splněny.

Fáze 1 - Minimální přenosová rychlost až CIR

V této fázi je šířka pásma konkrétní služby zvýšena z minimální přenosové rychlosti na dohodnutou rychlost (CIR). To zajišťuje, že síť je schopna podporovat konkrétní službu různými rychlostmi přenosu dat při zachování úrovně výkonu. Vzhledem k tomu, že služba přechází do CIR postupně, systém automaticky měří KPI v každém kroku, aby byly minimální výkonové požadavky vždy splněny.



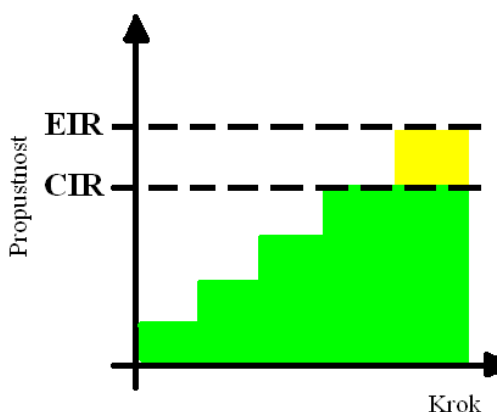
Obr. 1.5: Fáze 1

Tab. 1.4: Fáze 1

Schváleno	KPI v SLA
Neschváleno	Jakékoliv chybné KPI

Fáze 2 - CIR - EIR

V této fázi běží služba v pásmu CIR až k nadměrné rychlosti EIR. Rychlosti sice může být dosaženo, avšak podle zásad není zaručen výkon, proto není provedeno hodnocení KPI. V této fázi systém sleduje pouze přijatou propustnost. Vzhledem k tomu, že EIR není zaručeno, nemusí být šířka pásma dostupná pro veškerý provoz nad hodnotou CIR. CIR je považováno jako minimum, jakýkoliv příjem pod nižší je CIR je považován jako neúspěšný.



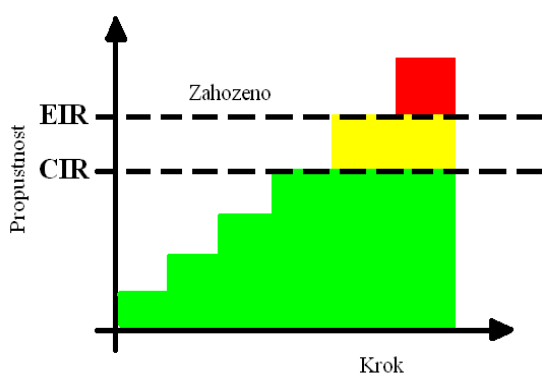
Tab. 1.5: Fáze 2

Schváleno	$CIR \leq Rx \text{ rate} \leq EIR$
Neschváleno	$Rx \text{ rate} < CIR$

Obr. 1.6: Fáze 2

Fáze 3 - Test v přetížení

Jednou z vlastností při přepravě paketů je schopnost zvládnout nárazový provoz. Za těchto podmínek může dojít k překročení šířky pásma EIR, což obvykle vede ke ztrátě provozu. V této fázi testu je zaslaný provoz nad EIR a získaná rychlost kontrolována. Přinejmenším by měl být přenesen provoz CIR. EIR by měl být přenesen v závislosti na dostupnosti zdrojů. Jakýkoliv provoz nad toto maximum je potřeba zlikvidovat, aby nebyla síť zahlcena. Pokud přijatý provoz překročí EIR, znamená to, že zařízení není správně nakonfigurováno a je indikován stav selhání.



Tab. 1.6: Fáze 3

Schváleno	$CIR \leq Rx \text{ rate} \leq EIR$
Neschváleno	$Rx \text{ rate} < CIR$ nebo $Rx \text{ rate} > EIR$

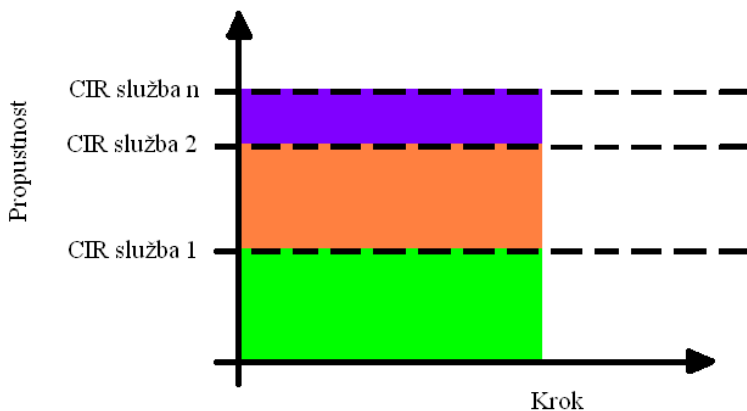
Obr. 1.7: Fáze 3

Tyto tři fáze jsou prováděny za provozu, takže jestli v síti existuje více služeb, musí být zkoušeny jednotlivé služby postupně. Na konci toho testu má uživatel jasnou

představu o tom, které síťové prvky a cesty byly správně nakonfigurovány, aby byly schopny přenést služby při minimální KPI.

Výkonnostní test služeb

Zatímco test nastavení služeb se zaměřuje na správné nastavení jednotlivých služeb v síťových prvcích, test výkonnosti služeb se zaměřuje na dodržování parametrů QoS. V tomto testu jsou nastavené služby vytvořené ve stejnou dobu pro stejné CIR po dobu od několika minut až několika dní. Během této doby je výkonnost služeb jednotlivě sledována. Pokud služba nesplní své výkonnostní parametry, je prohlášeno selhání stavu.



Obr. 1.8: Výkonnostní test

Tab. 1.7: Výkonnostní test

splněno	KPI s minimálními povolenými hodnotami
nesplněno	Jakékoliv KPI nedosahující dané hodnoty

Kombinace těchto dvou testů poskytuje všechny důležité výsledky v jednoduché kompletní testovací metodice. Tímto se rychle zjistí chyby v konfiguraci pomocí nástroje pro konfiguraci služeb se zaměřením na jednotlivé služby. Tento test se pak zaměří na to, jak v síti zajistit a garantovat všechny služby současně. Jakmile je také toto zajištěno, obvod je připraven k aktivaci a uveden do provozu. [3]

2 PROGRAM

Ke tvorbě zadaného software bylo přistoupeno dle následujících částí:

1. Upřesnění zadání
2. Návrh prostředí
3. Realizace aplikace

Po upřesnění zadání bylo zřejmé, že program by měl ze zadaného textového souboru, jehož obsahem jsou náměry z optických telekomunikačních sítí, zpracovat tyto data. Po zpracování jsou data, jakožto výsledky měření, zobrazeny požadovaným způsobem v grafickém uživatelském prostředí (GUI).

Dle možného zájmu čtenáře se tato kapitola dále člení na dvě části. V první části je zmíněna vnitřní struktura práce. To neznamená rozbor několikatisíciřádkového kódu programu, který je sice členěn do tříd, metod a místy komentován, ale zaměření se spíše na funkční bloky programu, které jsou sdruženy z jednotlivých řádků kódu.

Druhá část podrobně popisuje uživateli práci s aplikací od spuštění přes všechny možné pozice v programu až po jeho ukončení.

2.1 Vývojová část

Aby program splňoval další požadavek, kterým je přenositelnost, byl vybrán multiplatformní jazyk Java. Aplikace napsané v tomto jazyce jsou jednoduše spustitelné na známých používaných operačních systémech pomocí běhového prostředí (JRE). Předpokládá se, že tato podmínka bude již před spuštěním požadované aplikace splněna, protože JRE je využíván množstvím internetových služeb.

2.1.1 Java

Java je objektově orientovaný jazyk, rozsáhlá počítačová technologie a počítačová platforma. Technologie Java je základem pro téměř všechny typy síťových aplikací a je světovým standardem pro vývoj a poskytování mobilních aplikací, her, webového obsahu a podnikového softwaru. Jazyk Java vyvinula firma Sun Microsystems, nyní jej vlastní firma Oracle. Syntaxe je podobná jako má jazyk C++,

avšak neobsahuje složité konstrukce jako např. ukazatele. Díky automatické správě paměti (garbage collector) se programátor nemusí starat o manuální čištění paměti po použitých objektech. Je zde implementován mechanismus vláken a lze tudíž spouštět více úloh v rámci jednoho programu. Velkým ulehčením pro programátory je obsáhlost standardně dodávaných knihoven a dokumentace, se kterými se nemůže srovnávat asi žádný běžně používaný jazyk. Java také podporuje tvorbu dynamicky rozšiřitelných aplikací (plug-inů). Java také klade důraz na bezpečnost. Díky tomu, že je napsaný kód přeložen do bajtkódu a použitím bezpečnostních mechanismů lze zajistit, že program v Javě, který si uživatel stáhne ze sítě, mu nezformátuje disk, nebude komunikovat s žádným jiným počítačem, než ze kterého pochází. Jednou z největších výhod Javy je přenositelnost programu na libovolný operační systém, aniž by bylo třeba jej v tomto systému překládat. Přenositelnosti se dosahuje pomocí bajtkódu, jehož interpretace je pak úkolem speciálních programů označovaných jako Java platforma, které jsou předem na tento systém připraveny. [7]

Existují dva základní produkty, které Javu implementují:

1. Java SE Development Kit (JDK)
2. Java SE Runtime Environment (JRE)

JDK (Java vývojový kit) je nadstavba JRE (Java běhové prostředí) a obsahuje vše, co je zahrnuto v JRE plus nástroje jako jsou kompilátory, debuggery, které jsou nezbytné pro vývoj aplikací a apletů. JRE poskytuje knihovny, JVM (Java virtuální stroj) a ostatní komponenty, potřebné ke spuštění aplikací a apletů napsaných v jazyce Java. K běhu programů napsaných v Javě je (s výjimkou případů, kdy se Java programy zkompilují do nativního strojového kódu) bezpodmínečně nutné běhové prostředí nejméně v té verzi, pro kterou je daný program určen. Podle použití cílové aplikace je k jejímu vývoji použita edice JDK určená pro mobilní aplikace (Java ME), pro standardní počítačové aplikace (Java SE) nebo rozsáhlé podnikové aplikace (Java EE). [8]



Obr. 2.1: Java koncepční diagram

Aplikační programové rozhraní (API) obsahuje velké množství (několik tisíc) knihovných tříd, které jsou považovány za standardní, to znamená, že se musí vyskytovat v každém prostředí, kde se Java používá. Pokud program používá metody z API, není jejich kód součástí programu, protože je součástí API, takže program obsahuje pouze kód, který jsme napsali a soubory, ve kterých je kód uložen mají tedy poměrně malou velikost. Všechny třídy, metody a jejich proměnné jsou velmi dobře zdokumentovány a dokumentace je přístupná pomocí web prohlížečů. [7]

Java virtuální stroj je základním prvkem platformy Java. JVM je odpovědný za hardwarovou a systémovou nezávislost, malou velikost zkompilevaného kódu a schopností chránit uživatele před nebezpečnými programy. JVM je abstraktní virtuální stroj. Jako skutečný výpočetní stroj má instrukční sadu a manipuluje za běhu s různými částmi paměti. Je poměrně běžné implementovat programovací jazyk pomocí virtuálního stroje. JVM neví nic o programovacím jazyku Java, zná jenom formát binárního kódu (bajtkódu), který je uložen v souboru s příponou class. Tento soubor obsahuje instrukce pro JVM, tabulku symbolů a další doplňkové informace. [9]

Swing zavádí sadu komponent pro tvorbu grafického uživatelského rozhraní a přidává grafickou funkčnost a interaktivitu do Javy aplikací. Swing komponenty jsou realizovány výhradně v programovacím jazyce Java. Obsahuje také metody, které

umožňují vytvářet grafické uživatelské rozhraní mající stejný vzhled jako platforma, na níž je aplikace spuštěna. [10]

2.1.2 NetBeans

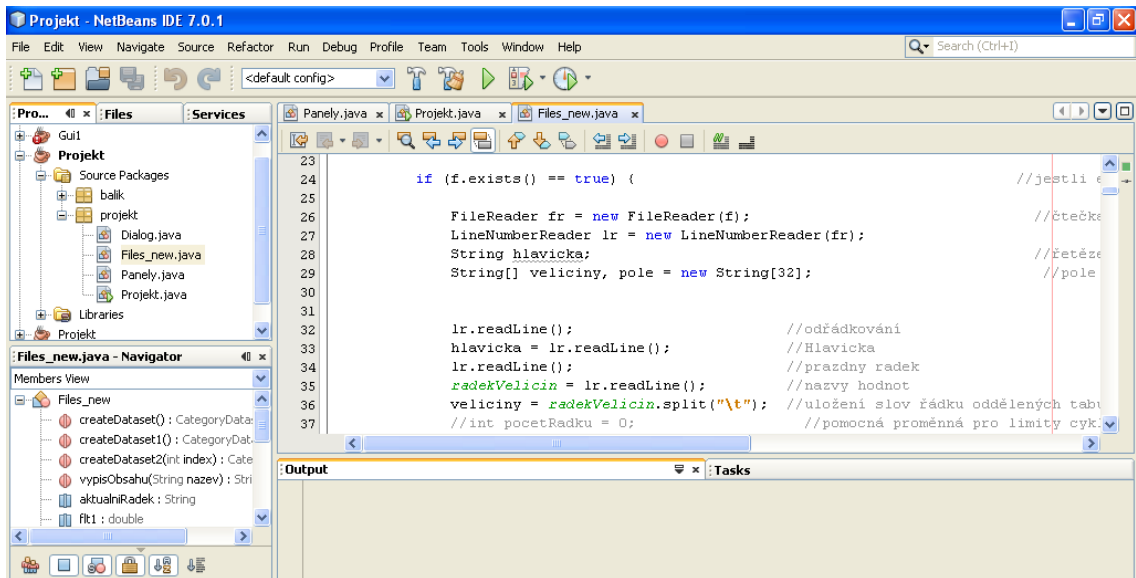
Pro vývoj software bylo použito návrhové prostředí NetBeans IDE, které poskytuje při tvorbě spoustu užitečných zjednodušení. NetBeans je vývojový Open Source projekt, který založila firma Sun Microsystems v červnu 2000. NetBeans nabízí programátorovi vše, co potřebuje pro pohodlné multiplatformní programování pro desktop, web i mobilní aplikace. Obsahuje například komponentu pro snadný návrh grafického uživatelského rozhraní, podporu pro Subversion (správa verzí software) a spoustu jiných věcí. Pokud některá z funkcí není přímo zakomponována, bude nejspíše dostupná jako zásuvný modul. V tuto dobu existují dva produkty, které jsou vyvíjeny pod licencí Open Source a je možné je bezplatně používat v komerčním i nekomerčním prostředí.[12]

Jedná se o tyto dva produkty:

1. Vývojové prostředí NetBeans (NetBeans IDE)
2. Vývojová platforma NetBeans (The NetBeans Platform)

Vývojové prostředí NetBeans je nástroj, pomocí kterého mohou programátoři psát, ladit, překládat a distribuovat své aplikace. Samotné vývojové prostředí je vytvořeno pro jazyk Java, ale podporuje i jakékoliv jiné programovací jazyky. Také existuje velké množství modulů, které toto vývojové prostředí rozšiřují. Vývojové prostředí NetBeans IDE je bezplatně šiřitelný produkt a jeho užívání není nijak omezeno.

Vývojová platforma je modulární a rozšiřitelný základ pro vytváření rozsáhlých desktopových aplikací. Nezávislí dodavatelé softwaru nabízejí dodatečné moduly, které lze snadno integrovat a které mohou být použity k vývoji jejich vlastních nástrojů a řešení. Platforma NetBeans poskytuje instalace funkcí, které by si musel programátor napsat sám (ukládání, připojení akcí k položkám menu, panely nástrojů, klávesové zkratky, správa oken aj.) [11]



Obr. 2.2: Vývojové prostředí Netbeans

2.1.3 Aplikace

Samotný kód aplikace je členěn do tříd. Třída definuje vlastnosti objektu. Každá třída má na starosti jiný problém. Jako první byla vytvářena třída pro načtení naměřených hodnot ze souboru do dvourozměrného pole programových proměnných. Toho je docíleno pomocí metod z nejdůležitější programové knihovny API (aplikační programové rozhraní). Jako první je zde obsažena metoda pro zjištění hlavičky, kterou představuje první nenulový řádek v souboru. Podle ní je ve spustitelné třídě (třída, která obsahuje metodu main) rozhodnuto, který soubor je načten a které objekty mají být vytvořeny, aby došlo k odpovídajícímu zobrazení. Dále je v této třídě definována metoda, která vrací pole řetězců, kterým je přímo naplněna tabulka. Obsahem této metody je čtečka souboru, která v jednom kroku přečte řádek v čteném souboru. Řádek je uložen do řetězcové proměnné. S těmito daty je dále pracováno pomocí API metod pro práci s řetězci, jako je rozdělení řetězce podle určitého obsaženého znaku, odstranění prázdných znaků ohraničující danou posloupnost nebo zkrácení řetězce o určitý počet znaků.

```

public static String[][] vypisObsahu(String nazev) throws IOException {
    File f = new File(nazev); //nova instance soubor
    if (f.exists() == true) { //jestli existuje tak se pro
        FileReader fr = new FileReader(f); //čtečka souborů
        LineNumberReader lr = new LineNumberReader(fr);

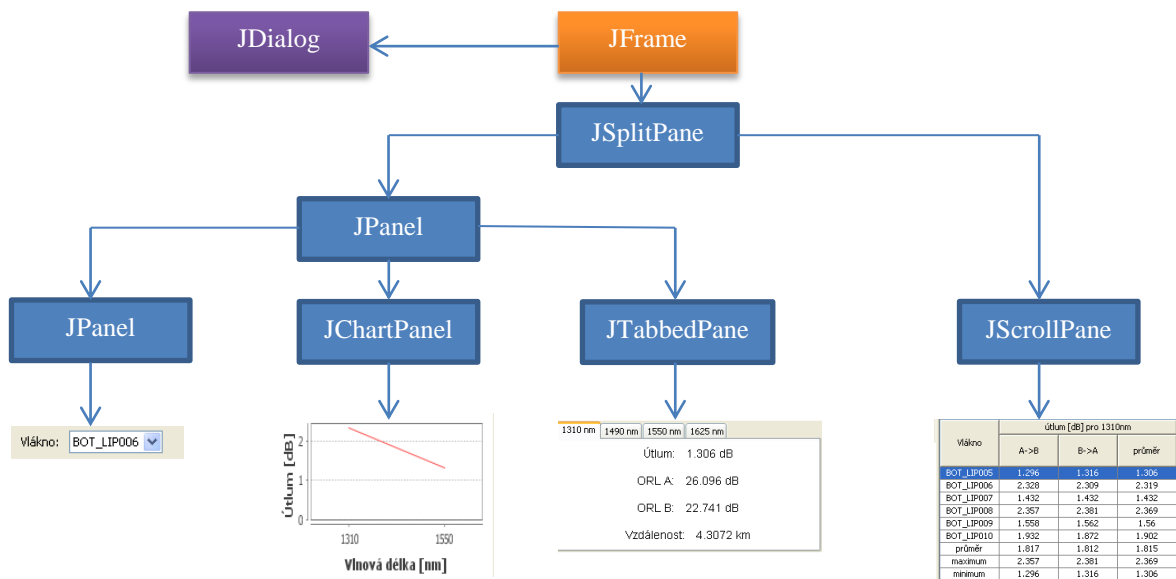
        String[] veliciny, pole = new String[32]; //řetězec pro načtení hlavič
        //pole řetězců pro uložení r

        lr.readLine(); //odřádkování
        hlavicka = lr.readLine(); //Hlavicka
    }
}

```

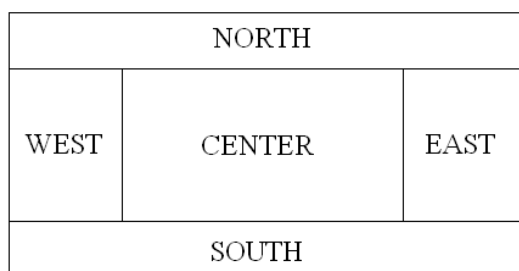
Obr. 2.3: Ukázka části kódu obsaženého v metodě pro čtení ze souboru

Ostatní třídy jsou použity pro potřeby grafického uživatelského rozhraní. Existuje zde hlavní spustitelná třída, která slučuje objekty vytvořené z ostatních tříd a obsahuje spustitelnou metodu main. Hlavní komponentou grafického uživatelského rozhraní je JFrame, což je hlavní okno programu, do kterého jsou poté vkládány další komponenty. Rozvržení komponent, jako jsou tabulka, přepínací panel a graf, je dáno přidělením správce rozvržení hlavní kontejnerové komponentě (JFrame).



Obr. 2.4: Rozvržení kontejnerových komponent – přímá metoda

Na hlavní kontejnerovou komponentu je položen rozdělovací panel (JSplitPane) pomocí BorderLayout správce rozložení, který je nastaven implicitně. Správce rozložení (layout) je definován u každé kontejnerové komponenty, na které lze pokládat další. Definuje velikost, pozici nebo pořadí umístěných bloků. Správce BorderLayout určuje, že mezi pokládanými bloky nejsou žádné mezery a bloky jsou plošně roztaženy ve své pozici, která je daná konstantou.



Obr. 2.5: Rozložení správce BorderLayout

Rozdělovací panel je nastaven tak, že je vertikálně rozdělen na pravou a levou část. Do pravé části je vložen JPanel pomocí stejného správce rozložení. Tento panel sdružuje tři další panely, které už náleží jednotlivým komponentám, nebo jejich seskupením. Více panelů (komponent) se do jednoho panelu na konkrétní pozici uloží pomocí konstant konkrétního správce rozložení, např. BorderLayout.NORTH a BorderLayout.SOUTH jsou konstanty, které jsou použity jako druhý parametr metody add. Tuto metodu používá kontejner (JFrame, JPanel, aj.) pro přidání komponenty na svoji plochu.

V případě vyhodnocení, že čtený soubor obsahuje náměry testu RFC 2544 je rozložení kontejnerů dáno blokovým schématem podle Obr. 2.2.

varování, chyb a statistik. Na těchto dílčích panelech jsou ještě umístěny přepínací panely a další panely, které už obsahují objekty typu JLabel a ty obsahují samotný text.

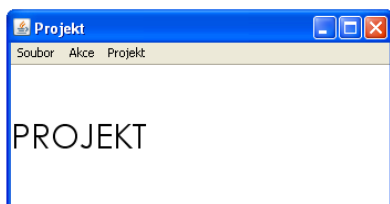
Dynamičnost programu je dána událostmi. Program je ovládán pomocí myši, případně pomocí číselných kláves při zadávání parametrů trasy. Uvnitř metod volaných při interakci s komponentami je převážná část kódu aplikace. Některé objekty (přepínací panel a sloupce tabulky) jsou vytvořeny dynamicky podle vektoru vlnových délek ze zdrojového textového souboru.

Naměřené hodnoty, i když jsou na pohled číselného formátu, jsou paměťově zpracovány ve formě řetězce znaků. Výjimkou je datový vstup pro vytvoření instance grafu, výpočet průměru, maxima a minima v tabulce, zaokrouhlení hodnot, výpočet a vyhodnocení limitu v dialogovém okně (JDialog). V těchto případech se řetězce přetypovávají na číselnou hodnotu primitivního datového typu Double, jedná-li se o reálné číslo, resp. Integer, má-li být zpracováno celé číslo. Po této konverzi se provede daná matematická operace, nebo sled operací a výsledek je zpět převeden do podoby posloupnosti znaků, čili řetězce typu String. Tento převod je proveden z toho důvodu, že s číslem v podobě řetězcového datového typu nelze provádět početní operace.

2.2 Uživatelská část

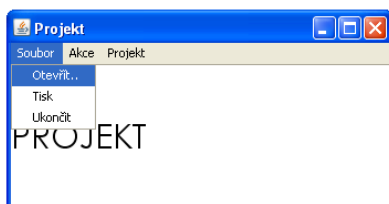
Z pohledu uživatele, který nepotřebuje znát vývojové podrobnosti programu, je důležité především porozumět ovládání programu. To je velmi jednoduché, protože program nenabízí mnoho přebytečných funkcí, ale poskytuje jen ty potřebné.

Při spouštění aplikace je nutno dbát na to, aby ve stejném adresáři, jako je spustitelný JAR soubor, byl také adresář s externími knihovnamí. Při oddělení knihoven a spuštění aplikace, operační systém zobrazí výjimku a aplikace nebude spuštěna. Řešením je vytvoření zástupce ke spustitelnému souboru na požadované místo, odkud chce uživatel aplikaci spouštět.

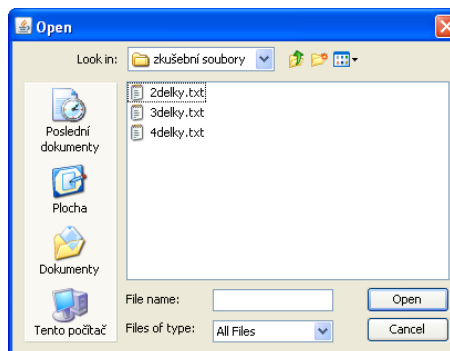


Obr. 2.7: Úvodní obrazovka

Po spuštění aplikace, jak je vidět na obr. 2.3, je zobrazena úvodní obrazovka. Uživatel má v tuto chvíli několik možností. Jako u jiných aplikací jsou v pravém horním rohu umístěny tlačítka pro minimalizaci, maximalizaci a zavření hlavního okna aplikace. Kromě toho má uživatel dvě možnosti jak pokračovat. Buď může načíst data ze souboru pomocí volby z menu Soubor → Otevřít, nebo může ukončit aplikaci pomocí volby z menu Soubor → Ukončit.



Obr. 2.8: Menu



Obr. 2.9: Otevření souboru

Po volbě položky Otevřít je zobrazeno okno pro výběr souboru ze souborového systému. Je nutné zvolit soubor správného vnitřního formátu, stejného jako mají soubory testovací. Na toto je aplikace velmi citlivá. Jakákoliv úprava může způsobit uložení nesprávných dat do pracovních proměnných, přetečení rozsahu paměťových polí, vyvolání výjimek nebo jinak ovlivnit správný chod aplikace. Proto je doporučeno soubor vygenerovaný měřicím přístrojem ponechat v nezměněné podobě. Z vybraného souboru jsou načtena data a zobrazena v tabulce, grafu a výpisovém panelu.

Otevřením souboru se z úvodního okna stane pracovní okno. Pracovní okno se liší nejen svou velikostí, která je pro přehlednost zvětšena, ale také i svým obsahem. Podle identifikace jednoho ze dvou kompatibilních formátů vstupního souboru, které jsou v aplikaci programově implementovány, je vybrán typ zobrazení.

2.2.1 Přímá metoda

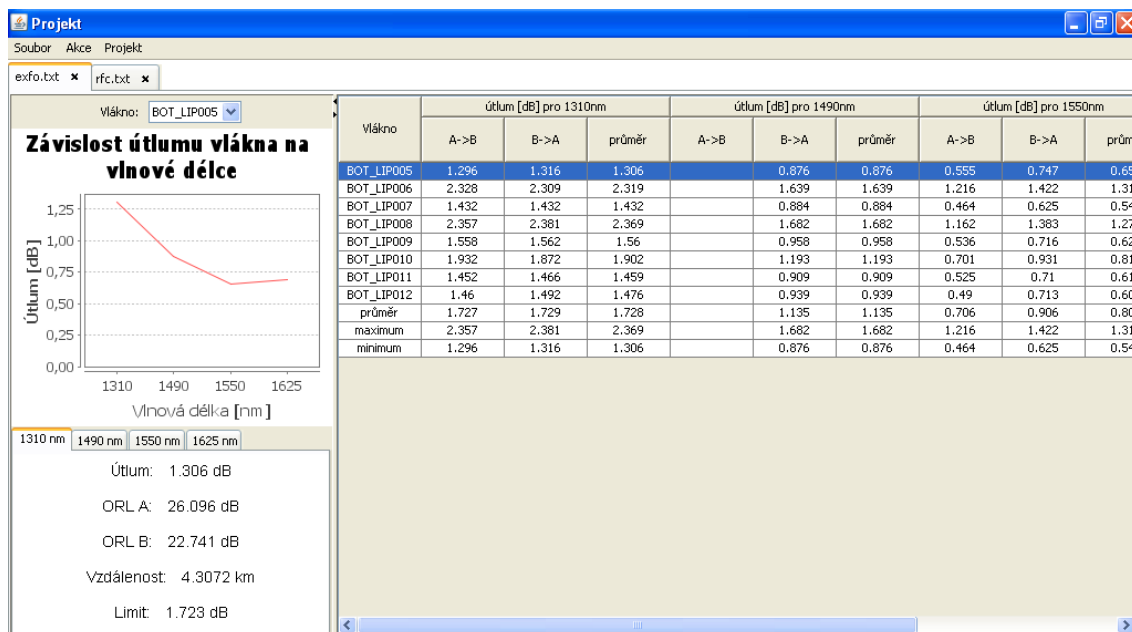
Je-li jako vstupní použit soubor s náměry týkající se fyzické vrstvy referenčního modelu, jsou v pracovním okně zobrazeny následující komponenty:

- tabulka všech naměřených hodnot (vpravo)
- graf závislosti útlumu na vlnové délce (vlevo uprostřed)
- přehledný výpis nejdůležitějších náměrů (vlevo dole)
- výběrová komponenta měřeného vlákna (vlevo nahoře)

V tabulce každý řádek obsahuje změřené hodnoty vztažené k jednomu vláknu. Popsané sloupce identifikují typ hodnoty, o jaký se jedná. Poslední tři řádky tabulky jsou vždy vyhrazeny pro matematické funkce, které jsou vykonány pro daný sloupec. Jedná se o maximální, minimální a průměrnou hodnotu z daného sloupce. Počet sloupců závisí na tom, při jakém počtu vlnových délek měření probíhalo. Každé vlnové délce náleží tři sloupce v tabulce (útlum jedním směrem, útlum opačným směrem a aritmetický průměr předchozích dvou hodnot). V případě velké rozlehlosti nejsou ve stejnou chvíli zobrazeny všechny hodnoty, proto je orientace v tabulce zajištěna podle potřeby pomocí vertikálního či horizontálního posuvníku. Tabulka si navzájem s výběrovou komponentou poskytuje jednu dynamickou vlastnost. Označením řádku tabulky dojde ve stejnou chvíli k výběru odpovídajícího položky ve výběrové komponentě a naopak. Na základě tohoto výběru jsou z pole hodnot vybrány správné hodnoty pro vykreslení grafu a vyplnění výpisového panelu.

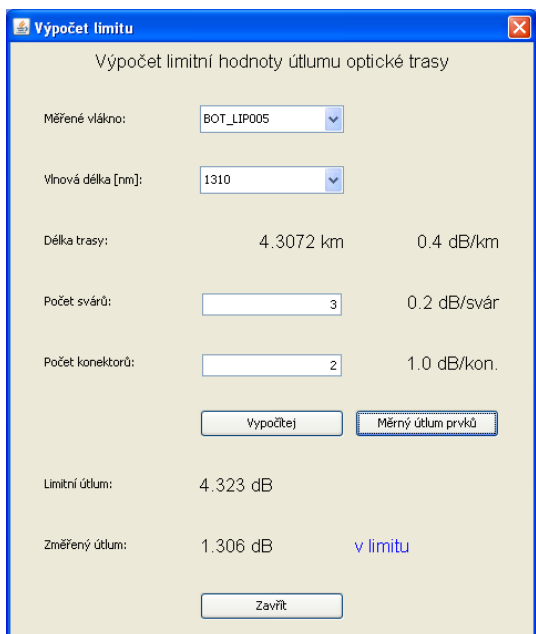
Graf je vykreslen vždy pro jedno optické vlákno. To, které vlákno je právě vybráno, můžeme ovlivnit pomocí výběrové komponenty, která obsahuje stejná vlákna jako první sloupec v tabulce. Po výběru jiného vlákna, než je právě zobrazeno, dojde k překreslení předchozího grafu a zobrazení aktuálního. Druhou možností, jakým způsobem dojde k výběru vlákna, je kliknutí myší na požadovaný řádek v tabulce.

Výpisový (přepínací) panel obsahuje záložky, jejichž počet je závislý na počtu měřených vlnových délek, a každá z nich sdružuje hodnoty právě pro určitou vlnovou délku. Vlnová délka je zobrazena v popisku záložky. Uvnitř záložek jsou zobrazeny důležité hodnoty, jakými jsou útlum, útlum odrazu (ORL) a vzdálenost, která odpovídá délce optické trasy. Tento panel je, stejně jako graf, překreslován výběrem jiného vlákna.



Obr. 2.10: Načtený soubor s náměry pro 4 vlnové délky

Chce-li se uživatel ujistit, zdali naměřené hodnoty útlumu nepřekračují doporučenou mez, potom zvolí z menu Akce → Limit. V zobrazeném dialogovém okně se výběrem z daných možností zvolí vlákno a vlnovou délku. Touto volbou je nejen přesně identifikována naměřená hodnota útlumu použita pro srovnání, ale také i vzdálenost měřeného vlákna použitá při výpočtu a také zobrazena. Na uživateli je už jen vyplnění dvou textových polí, do kterých je povoleno zadávat jen celé číslo vypovídající o počtu spojovacích prvků. Pokud je zadáno cokoliv jiného, než celé číslo, je uživatel upozorněn, že má zadat celé číslo. Pokud zůstane alespoň jedno pole nevyplněné, je uživatel upozorněn k zadání počtu svárů nebo konektorů, podle toho, které pole zůstane prázdné. Po stisku tlačítka vypočítej, jsou poté níže zobrazeny obě hodnoty útlumu (změřená i limitní) a srovnány. Srovnání proběhne jednoduchou textovou hláškou, informující o pozitivním nebo negativním výsledku. V případě pozitivního výsledku je písmo zvýrazněno modrou barvou, v opačném případě je text zvýrazněn červenou barvou.



Obr. 2.11: Dialogové okno pro výpočet limitní hodnoty

2.2.2 RFC 2544 test

Je-li jako vstupní použit soubor s náměry týkající se testu podle internetového standardu RFC 2544, jsou v pracovním okně zobrazeny následující bloky:

- Nadpis (nahore)
- Konfigurace linky a koncových zařízení (vlevo uprostřed)
- Výsledky testu (vpravo uprostřed)
- Alarmy (vlevo dole)
- Chyby (uprostřed dole)
- Statistiky (vpravo dole)

Nadpis obsahuje text „Test linky“ a je neměnný. Blok konfigurace obsahuje informace o nastavení síťového rozhraní měřicí stanice a použité linky. Informuje o typu rozhraní, názvu testu, topologii, vysílacím módu, rychlosti linky, typu duplexu, automatickém nastavení přenosové rychlosti a typu duplexu (Auto-negotiation), VLAN adrese, IP adrese, MAC adrese, využití přenosové kapacity linky, míře ztrátovosti a o době trvání testů.

Výsledky testů jsou zobrazeny v záložkách. Každá záložka obsahuje kromě názvu testu v popisku záložky, také informace o souvisejících vrstvách referenčního

modelu, zobrazení výsledků, kroku zobrazení nebo módu zpoždění (jen v případě testu zpoždění). Výsledky jsou interpretovány tabulkou, která obsahuje počet řádků stejný jako je množství různých velikostí testovacích paketů. Pro každý řádek je uvedena hodnota daného testu jedním směrem linky a hodnota stejného testu stejné velikosti paketu opačným směrem linky. Výjimkou je test zpoždění, které je definováno časem, který urazí paket tam i zpět. V tomto případě je k velikosti paketu přiřazena pouze jedna hodnota.

Alarmy informují s jakou frekvencí nebo jak dlouhou dobu docházelo při testu k varováním. V chybovém panelu jsou zobrazeny ve formě tabulkového formátu možné příčiny chyb a v odpovídajících sloupcích je pak kvantifikováno jak dlouho, v jakém počtu a v jaké míře k těmto chybám docházelo. Statistiky zobrazující celkový počet vyslaných rámců a celkový počet přijatých rámců.

The screenshot shows the 'Test linky' application window with the following sections:

- Konfigurace:**
 - Název testu: TEST
 - Typ rozhraní: Ethernet 10/100/1000
 - Typ aplikace: RFC 2544
 - Topologie: Jednoportová
 - Port: RFC2544
 - Vysílací mód: Elektrický
 - Rychlost: 1Gbps
 - Duplexní mód: Plný
 - Auto-Negotiation: Vypnuto
 - VLAN: Žádné
 - IP adresa: 192.168.1.100
 - MAC adresa: 00:03:01:08:3A:AD
- Výsledky:**
 - Propustnost, Zatížitelnost, Ztrátovost, Zpoždění
 - Vrstva: Layer 1-2-3
 - Zobrazení výsledků: Aktuální

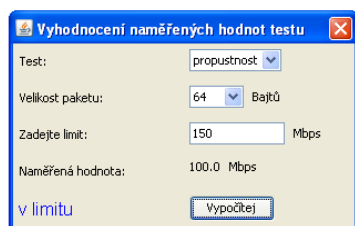
Velikost rámce(B)	Místní až vzdálená(Mbps)	Vzdálená až místní(Mbps)
64	299,999467	300
128	300,20284	300,20284
256	299,999467	299,999148
512	300,056402	300,055549
1024	299,999467	300
1280	300,023079	300,022227
1518	299,980495	299,980495
- Alarmy:**
 - Port: Ethernet
 - Frekvence[s]: --
- Chyby:**

Ethernet	Vyšší vrstvy	Sekundy	Počet	Poměr
Symbol	0	0	0	0,00E00
Nečinnost	1	255	255	3,69E-07
Chyb. přenos	1	46087	46087	1,05E-02
FCS	0	0	0	0,00E00
Jabber/Giant	0	0	0	0,00E00
Runt	0	0	0	0,00E00
Nedostatečný	0	0	0	0,00E00
Zarovnání	--	--	--	--
- Statistiky:**
 - Počet platných rámců
 - Celkový počet vyslaných rámců: 4324806
 - Celkový počet přijatých rámců: 4325845

Obr. 2.12: Pracovní okno – RFC 2544 test

Chce-li se uživatel ujistit, zdali naměřené hodnoty útlumu nepřekračují stanovenou mez, potom zvolí z menu Akce → Limit. V zobrazeném dialogovém okně se výběrem z daných možností zvolí jako první název testu (propustnost, zatížitelnost, ztrátovost a zpoždění), pak vybere velikost testovacího paketu z daných možností (64,

128, 256, 512, 1024, 1280 nebo 1518 bajtů) a zadá mezní hodnotu (limit) v určených jednotkách. Kliknutím na tlačítko „Vypočítej“ dojde k vyhodnocení. Vyhodnocení proběhne tak, že je zobrazena naměřená hodnota a dojde k jejímu porovnání se zadanou. Je-li naměřená hodnota menší, než zadaná mezní, je vypsána modrou barvou hláška „v limitu“, v opačném případě je vypsána červenou barvou hláška „nad limitem“. Zůstane-li kolonka nevyplněná, vypíše se černou barvou „zadejte limit“.



Obr. 2.13: Dialogové okno pro vyhodnocení náměrů

ZÁVĚR

V první části práce bylo dosaženo realizace základního grafického uživatelského prostředí pro zobrazení náměrů z měření optických telekomunikačních sítí přímou metodou. Tato aplikace byla napsána pomocí objektově orientovaného jazyka Java a její vývoj probíhal v NetBeans IDE, které obsahuje funkce k ulehčení práce programátorovi. Výsledkem je spustitelný jar archiv s přibalenými externími knihovny. Knihovny jsou potřebné ke správnému zobrazení grafu, ostatní prvky grafického uživatelského rozhraní jsou uloženy v interních knihovnách běhového prostředí. Program jednoduše a přehledně zobrazuje, pomocí různých grafických komponent, data z uspořádaného, ale nepřehledného textového souboru, který je vygenerován měřicím přístrojem.

Proto, aby se nejednalo o pouhé statické zobrazení, jsou komponenty programově propojeny takovým způsobem, že při změně jedné z nich, dojde k odpovídající změně také ostatních. Navíc formou dialogového okna, po zvolení určitých parametrů týkající se měřené trasy, upozorňuje uživatele na hodnoty, které jsou z výkonového hlediska přípustné, a které nikoliv.

Ve druhé části práce byla zvýšena komplexnost aplikace tím, že je možno načíst další soubor s jiným vnitřním formátem, ale hlavně s jinými náměry. Jedná se o výsledky testu RFC 2544. Změřené hodnoty propustnosti, zatížitelnosti, ztrátovosti a zpoždění jsou zobrazeny v tabulce. Kromě výsledků testu obsahuje okno také informace o nastavení měřicí stanice, parametry linky, varování, chyby a statistiky proběhlého měření.

Uživatel pomocí dialogového okna, ve kterém zadá limitní hodnotu zvoleného testu pro konkrétní velikost paketu, zjistí, zdali naměřená hodnota nepřesahuje stanovenou mez a dojde tak k rychlému vyhodnocení.

LITERATURA

- [1] FILKA, M. *Optoelektronika: Pro telekomunikace a informatiku*. Vyd. 1. Brno: Centra, spol.s.r.o., 2009. 369 s. ISBN 978-80-86785-14-1.
- [2] BRANDNER, S.; MCQUAID, J. *Benchmarking Methodology for Network Interconnect Devices* [online]. Březen 1999. Dostupné z: <<http://www.ietf.org/rfc/rfc2544.txt>>
- [3] THIerno, D. *EtherSAM: The new standard in Ethernet service testing* [online]. Březen 2011. Dostupné z: <[http:// documents.exfo.com/appnotes/anote230-ang.pdf](http://documents.exfo.com/appnotes/anote230-ang.pdf)>
- [4] ŠERÝ, R. *Optické síť* [online]. 18.12.2006. Dostupné z: <<http://programujte.com/clanek/2006120301-opticke-site/>>
- [5] *Měření útlumu optického vlákna metodou zpětného rozptylu (OTDR) a přímou metodou* [online]. 9.11.2009. Dostupné z: <http://www.comtel.cz/cz/?s=vyuka&l=pred_vyber&select=materialy&kod=20>
- [6] MARŠÁLEK, L. *Optická vlákna* [online]. 15. 4. 2006. Dostupné z: <<http://goro.czweb.org/download/interest/vlakna.pdf>>
- [7] HEROUT, P. *Učebnice jazyka Java*. Vyd. 5. České Budějovice: Kopp nakladatelství, 2010. 386 s. ISBN: 978-80-7232-398-2.
- [8] JELÍNEK, L. *JDK, vývojová prostředí* [online]. 5. 5. 2005. Dostupné z: <http://www.linuxsoft.cz/article.php?id_article=797>
- [9] *Chapter 1. Introduction* [online]. Dostupné z: <<http://docs.oracle.com/javase/specs/jvms/se7/html/jvms-1.html#jvms-1.2>>
- [10] *Swing* [online]. Dostupné z: <<http://docs.oracle.com/javase/7/docs/technotes/guides/swing/index.html>>
- [11] *The NetBeans Platform* [online]. c2012. Dostupné z: <<http://netbeans.org>>
- [12] *NetBeans* [online]. 7. 8. 2006. Dostupné z: <<http://www.abclinuxu.cz/software/programovani/ide/netbeans>>

SEZNAM ZKRATEK A SYMBOLŮ

Symbol	Význam
A	útlum optického vlákna
l	délka měřené optické trasy
α	měrný útlum optického vlákna
λ	vlnová délka optického signálu
Zkratka	Význam
APD	lavinová fotodioda
BMWG	srovnávací metodika
CIR	dohodnutá rychlost
DUT	testované zařízení
EIR	nadměrná rychlost
FTP	protokol pro přenos souborů
GUI	grafické uživatelské rozhraní
IP	internetový protokol
IPTV	televizní signál přes internet
JDK	Java vývojový kit
JRE	Java běhové prostředí
JVM	Java virtuální stroj
KPI	klíčové výkonnostní identifikátory
MAC	fyzická adresa
OAM	operace, administrace a údržba
OLTS	metoda vložného útlumu
OTDR	metoda zpětného rozptylu
QoS	kvalita služeb
RFC	standard pro testování síťových parametrů
SLA	smlouva o poskytování služeb
VLAN	virtuální síť
VoIP	přenos hlasu přes internet
VOČ	vazební optický článek