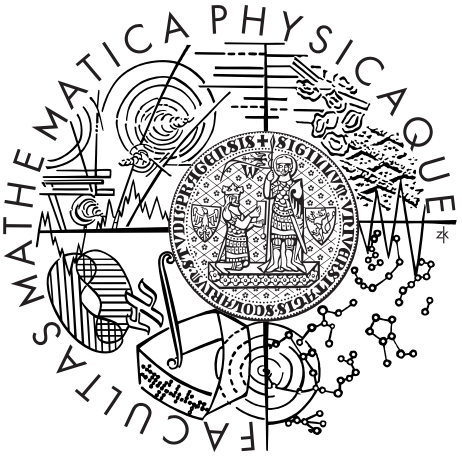
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Pavol Rohár

Archiv mailových konferencí

Katedra Aplikované Matematiky

Vedoucí bakalářské práce: Mgr. Martin Mareš, Ph.D.

Studijní program: Informatika
Studijní obor: Obecná informatika

Praha 2014

Chcem sa poďakovať môjmu vedúcemu Mgr. Martinovi Marešovi, Ph.D. za odbornú pomoc, usmernenie, cenné rady a pripomienky pri písaní mojej práce. Ďalej sa chcem poďakovať Alexanderovi Wirtovi za poskytnutie veľkých mailových archívov projektu Debian na účely testovania a ladenia môjho programu. A v neposlednom rade sa chcem poďakovať mojej rodine za jazykové korektúry a ďalšie pripomienky k mojej práci.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V ..... dne ..... podpis

Název práce: Archiv mailových konferencí

Autor: Pavol Rohár

Katedra: Katedra Aplikované Matematiky

Vedoucí bakalářské práce: Mgr. Martin Mareš, Ph.D.

Abstrakt: Tato práce se zabývá automatickým archivováním mailů z mailových konferencí. Správcem konference k tomu má poskytnout vhodný nástroj a jednotlivým účastníkům umožnit číst archivované maily pomocí webového prohlížeče. Za cíl si práce klade funkční a efektivní program, který bude podporovat maily ve formátu MIME a bude umět pracovat s mailovými přílohami. Program bude též umět zobrazovat maily v různých národních znakových sadách a bezpečně zpracovávat potenciálně nedůvěryhodná data. Kromě toho dokáže rozumně seskupovat maily do jednotlivých vláken. A v neposlední řadě i co nejlépe zpracovávat maily, které nesplňují požadované a běžně používané internetové standardy.

Klíčová slova: e-mail, konference, archiv

Title: Mailing list archives

Author: Pavol Rohár

Department: Department of Applied Mathematics

Supervisor: Mgr. Martin Mareš, Ph.D.

Abstract: This thesis deals with the concept of automated archiving of e-mails from mailing list archives. It is supposed to provide suitable tool for archiving e-mails for administrators and enable users to read archived e-mails via web browser. The main objective of the thesis is to produce an effective program which will support e-mails in MIME format and will be able to work with attachments. The program will be able to correctly display e-mails in various national character encodings and safely process potential untrustworthy data. It will also be able to reasonably group e-mails into individual threads. Lastly, the program will be able to process e-mails which don't meet required and customary Internet Standards in the best possible way.

Keywords: e-mail, mailing list, archive

Názov práce: Archív mailových konferencií

Autor: Pavol Rohár

Katedra: Katedra Aplikovanej Matematiky

Vedúci bakalárskej práce: Mgr. Martin Mareš, Ph.D.

Abstrakt: Táto práca sa zaoberá automatickým archivovaním mailov z mailových konferencií. Správcom konferencií má k tomu poskytnúť vhodný nástroj a jednotlivým účastníkom umožniť čítať archivované maily pomocou webového prehliadača. Za cieľ si práca kladie funkčný a efektívny program, ktorý bude podporovať maily vo formáte MIME a bude vedieť pracovať s mailovými prílohami. Program bude vedieť aj správne zobrazovať maily v rôznych národných znakových sadách a bezpečne spracovávať potenciálne nedôveryhodné dáta. Okrem toho dokáže aj rozumne zoskupovať maily do jednotlivých vlákien. A v neposlednom rade aj čo najlepšie spracovávať maily, ktoré nespĺňajú požadované a zaužívané internetové štandardy.

Kľúčové slová: e-mail, konferencia, archív

Obsah

Úvod

Mailová konferencia je nástroj na komunikáciu medzi veľkou skupinou ľudí. Ako komunikačný prostriedok slúžia elektronické mailové správy. Zvyčajne účastníci mailovej konferencie píšu maily iba na jednú zvolenú adresu. Mailový server, ktorý sa stará o príjem správ zo zvolenej adresy, automaticky posiela prijaté správy ďalej na mailové adresy všetkých zaregistrovaným členov konferencie.

Táto práca sa zaoberá archivovaním mailov z mailových konferencií a následným zobrazovaním jednotlivých mailov vo webovom prehliadači. Výstupom práce má byť program na archiváciu správ a možnosť prepojiť ho s webovým serverom.

Mailové konferencie môžeme rozlíšiť na dva typy: verejné a privátne. Verejná konferencia prijíma správy od ľubovoľného prispievateľa (ten nemusí byť ani jej členom) a umožňuje ktorejkoľvek osobe sa stať členom. Typický príklad použitia verejnej konferencie je pri open source projektoch. Každý človek sa môže vyjadriť ku dianiu či vývoju projektu a takisto sa môže pozrieť, čo sa dialo s projektom pred rokmi.

Druhý typ konferencie je privátny a je určený iba pre samotných členov. Nikto okrem členov nemôže do konferencie písať maily a iba správca konferencie má právo prideľovať a rušiť členstvo. Používa sa často vo firemnej sfére, alebo medzi ľuďmi v situáciach, keď nechcú alebo nemôžu zverejniť obsah svojich rozhovorov.

Mailový server, ktorý prijíma správy z adresy konferencie, nezvykne samotné správy archivovať. Ak si jednotlivý členovia konferencie nerobia vlastné zálohy mailov, tak sa zvyčajne už k starým správam nedostanú.

Pri verejných konferenciách je zvykom mať aj verejný archív na internete. V ňom si môže každý človek prezerať aktuálne dianie, históriu a prečítať ktorúkoľvek správu, ktorá sa v archíve nachádza.

Cieľom tejto práce je poskytnúť program, ktorý bude vytvárať práve takéto archívy dostupné cez internet a medzi kľúčové vlastnosti programu bude patriť:

- podpora pre maily vo formáte MIME
- správne zobrazenie mailov, ktoré sú v národných znakových sadách
- bezpečné zaobchádzanie s potenciálne nebezpečnými časťami a prílohami
- rozumné zoskupovanie mailov do vlákien a budovanie stromu odpovedí
- efektívne spracovávanie veľkého objemu mailov

Kapitola 1 tejto práce sa zaoberá formátmi, ktoré sa používajú mailovými aplikáciami na spracovávanie a uchovávanie mailových správ. Kapitola 2 popisuje najčastejšie problémy a ich riešenia, ktoré sa vyskytujú pri spracovávaní mailov. Ďalej kapitola 3 popisuje existujúce internetové služby, ktoré poskytujú archiváciu mailov z verejných konferencií. Kapitola 4 obsahuje návrh programu práce, popis použitých formátov a algoritmov. Ďalej, kapitola 5 rozoberá samotnú implementáciu a ako konkrétne program rieši niektoré problémy. A v kapitole 6 sa nachádza užívateľská dokumentácia, konkrétne ako sa program inštaluje, ovláda a ako sa s archívmi pracuje.

1. Používané formáty pre mailové správy

V tejto kapitole sú uvedené formáty, ktoré sa používajú pre mailové správy.

1.1 Formát mailovej správy

1.1.1 Jednoduchá správa

Jednotlivé programy, ktoré slúžia na vytváranie a spracovávanie mailových správ, sa majú riadiť štandardom RFC2822 [1]. Tento štandard rozširuje a nahrádza jeho staršiu verziu RFC822 [2]. Štandard presne definuje akú má mať mailová správa štruktúru, aby sa predišlo problémom so vzájomnou nekompatibilitou.

Konkrétne, mailová správa je postupnosť znakov a obsahuje dve časti: Povinnú hlavičku a nepovinné telo správy oddelené od hlavičky prázdnym riadkom.

Príklad mailovej správy:

```
From: address1@example.org
To: "Name" <address2@example.org>, address3@example.org
Date: Tue, 24 Jun 2014 12:05:42 +0200
Subject: Email subject
Message-Id: <201406241205.06518@example.org>
In-Reply-To: <201406101716.06518@example.org>
References: <201406060612.04742@example.org>
  <201406101716.06518@example.org>

This is body of message.
```

Hlavička mailu predstavuje akúsi tabuľku, ktorá obsahuje názvy položiek a k ním prislúchajúce hodnoty. Telo správy potom obsahuje samotný text mailu. V hlavičke musia byť prítomne aspoň povinné položky `From` a `Date`. Význam povinných ako aj ďalších nepovinných položiek je uvedený v nasledujúcej tabuľke.

Položky v hlavičke:

Názov položky	*Význam hodnoty položky*
`From`	Odosielateľ mailovej správy (povinný údaj)
`Date`	Dátum[1], kedy bola správa vytvorená (povinný údaj)
`To`	Prijímatelia mailovej správy
`Cc`	Prijímatelia kópie mailovej správy
`Subject`	Predmet správy
`Message-Id`	Jednoznačný identifikátor správy
`In-Reply-to`	Hodnoty položiek `Message-Id` zo správ, na ktoré sa odpovedá
`References`	Všetky hodnoty položiek `References` a `Message-Id` zo správ, na ktoré sa odpovedá

[1] Presný formát dátumu definuje RFC2822

Položky `In-Reply-To` a `References` majú dôležitý význam pri určovaní správ, ktoré patria do jednej ucelenej konverzácie – vlákna.

Položka `In-Reply-To` pre jeden konkrétny mail nám určuje všetky maily, na ktoré je daný mail odpoveďou. V grafovej terminológii si môžeme jednotlivé maily označiť ako vrcholy a orientované hrany nám potom budú tvoriť hodnoty z položiek `In-Reply-To`. V prípade, že všetky maily budú pravdivo uvádzať túto položku, potom takto vzniknutý orientovaný graf bude acyklický.

Položka `References` pre jeden konkrétny mail udáva všetky hodnoty položiek `References` a `Message-Id` z mailov, na ktoré je daný mail odpoveďou. Teda hodnoty z položiek `References` nám budú pre graf z predchádzajúceho odstavca tvoriť jeho tranzitívny uzáver.

Položky `Message-Id`, `In-Reply-To` a `References` nie sú v maili povinné a teda nie vždy dostaneme pre jedno vlákno úplný tranzitívny uzáver orientovaného acyklického grafu. Pri vynechaní položky `In-Reply-To` sa môže stať, že graf, ktorý dostaneme z položiek `References` bude dokonca obsahovať viac vrcholov ako graf vybudovaný z položiek `In-Reply-To`. Alebo pri poškodení mailových hlavičiek (napr. pri prenose mailu, vytvorení dvoch mailov s rovnakým `Message-Id` alebo pri chybnej interpretácií) sa môže stať, že graf bude obsahovať cyklus. Pre jednoduchšiu prácu budeme ďalej súvislé komponenty, ktoré sa nachádzajú v neorientovanom grafe vzniknutom zjednotením grafov z `In-Reply-To` a `References` položiek považovať za jednotlivé oddelené vlákna.

1.1.2 Správa vo formáte MIME

Štandard RFC2822 však povoľuje používať iba 7-bitové znaky ASCII a nedefinuje žiaden spôsob, ako pripájať k mailovej správe prílohy či ďalšie časti. Naviac telo správy musí byť text. Nemožnosť používať národné znakové sady, či univerzálny Unicode prakticky znemožňuje vytvoriť mailové správy v národných abecedách, či použiť inú abecedu ako latinku.

Tieto problémy rieši až MIME (Multipurpose Internet Mail Extensions), čo je rozšírenie RFC822. Definujú ho internetové štandardy RFC2045 [3], RFC2046 [4], RFC2047 [5] a rozširujú ho ďalšie dokumenty ako napr. RFC2183 [6], RFC2231 [7], RFC6532 [8] či ďalšie. Mailová správa vo formáte MIME naďalej vyhovuje štandardu RFC2822, avšak na jej spracovanie musí aplikácia podporovať tento formát. MIME definuje spôsob, akým sa 8-bitový text dá jednoznačne zakódovať na 7--bitové ASCII znaky. Štandard RFC2045 definuje niekoľko kódovaní a na rozdiel od RFC2822 povoľuje aj 8-bitové kódovanie pre telo mailu.

MIME zásadne mení štruktúru tela mailovej správy. RFC2822 hovorí, že telo správy tvorí samotný text. Telo správy vo formáte MIME je jeden MIME objekt.

MIME objekt obsahuje, podobne ako mailová správa, hlavičku a telo. Pre účely archivácie mailov sa v hlavičke nachádzajú tieto dôležité položky: `Content-Type`, `Content-Disposition` a `Content-Description`. Položka `Content-Type` hovorí o aký typ dát sa v tele objektu jedná a môže obsahovať dodatočné informácie o použitom kódovaní textu (ak sa jedná o textovú časť). `Content-Disposition` určuje, či je daný objekt určený na okamžité zobrazenie (hodnota `inline`), alebo je to príloha mailu (hodnota `attachment`) a špecifikuje ďalšie parametre ako názov súboru, či dátum. Položka `Content-Description` uvádza textový popis daného MIME objektu. Následne telo objektu tvoria samotné dáta.

MIME objekt môže byť špeciálneho typu keď má v položke `Content-Type` uvedenú hodnotu `multipart`. Jeho telo potom tvorí usporiadaný zoznam ďalších MIME objektov. Toto umožňuje vytvoriť stromovitú štruktúru objektov v rámci mailovej správy vo formáte MIME. Takýmto spôsobom je možné kombinovať textové časti mailovej správy s prílohami.

V prípade, že objekt je typu `multipart/alternative`, tak telo objektu obsahuje tiež zoznam MIME objektov. Na rozdiel od predchádzajúceho prípadu, tieto objekty sú alternatívy rovnakých dát, ale v rôznych formátoch. Sú medzi sebou zameniteľné a je na aplikácii, ktorú verziu z ponúkaných alternatív užívateľovi zobrazí. Väčšinou sa do objektu typu `multipart/alternative` vkladá HTML verzia mailovej správy a jej ekvivalentná alternatíva pozostávajúca len z čistého neformátovaného textu.

Ďalej nasleduje príklad mailovej správy vo formáte MIME. Správa obsahuje aj slovenské znaky, ktoré sú zakódované vhodným povoleným spôsobom. Hlavný MIME objekt, ktorý tvorí telo mailu je typu `multipart/alternative`. Obsahuje dve časti a to čistý text a formátovaný text v HTML.

```
From: =?utf-8?q?Odosielate=C4=BE?= <address1@example.org>
To: =?utf-8?q?Prij=C3=ADmate=C4=BE?= <address2@example.org>
Subject: Toto je predmet mailovej =?utf-8?q?spr=C3=A1vy?=
Date: Tue, 24 Jun 2014 12:17:34 +0200
MIME-Version: 1.0
Content-Type: multipart/alternative;
  boundary="Boundary-01=_+AVqTgWJZL2OhiZ"
Content-Transfer-Encoding: 7bit

--Boundary-01=_+AVqTgWJZL2OhiZ
Content-Type: text/plain; charset="utf-8"
Content-Transfer-Encoding: quoted-printable

Toto je samotn=C3=A9 telo mailovej spr=C3=A1vy.

--Boundary-01=_+AVqTgWJZL2OhiZ
Content-Type: text/html; charset="utf-8"
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN" "http://www.w3.o=
rg/TR/REC-html40/strict.dtd">
<html><head><meta name=3D"qrichtext" content=3D"1" /><style type=
=3D"text/css">
p, li { white-space: pre-wrap; }
</style></head><body style=3D" font-family:'DejaVu Sans Mono'; fo=
nt-size:8pt; font-weight:400; font-style:normal;">
<p style=3D" margin-top:0px; margin-bottom:0px; margin-left:0px; =
margin-right:0px; -qt-block-indent:0; text-indent:0px; -qt-user-s=
tate:0;">Toto je samotn=C3=A9 telo mailovej spr=C3=A1vy.</p></bod=
y></html>
--Boundary-01=_+AVqTgWJZL2OhiZ--
```

Znaky, ktoré nie sú súčasťou 7-bitovej ASCII tabuľky, sú kódované spôsobom *quoted-printable*. Každý jeden bajt je nahradený znakom `=` a ďalšími dvomi znakmi udávajúcimi číslo v šestnástkovej sústave, ktoré zodpovedá danému bajtu. Za povšimnutie stojí znak `=`, ktorý je na konci riadku. Udáva, že aktuálny riadok pokračuje na ďalšom riadku bez zlomu.

Obdobne sú kódované aj položky hlavičiek, v tomto prípade ide o `From`, `To` a `Subject`. Naviac je na začiatku hodnôt špecifikovaná znaková sada a tá sa nemusí zhodovať s tou, ktorá je použitá v iných častiach mailu.

Kódovanie *quoted-printable* nie je jediný spôsob ako zakódovať iné znaky ako ASCII. Napríklad na binárne prílohy sa používa *Base64*. Kódovanie spočíva v prevedení binárneho textu na postupnosť znakov iba z ASCII tabuľky[2].

Za zmienku ešte stojí hodnota `boundary`, ktorá je špecifikovaná v položke `Content-Type`. Tá určuje oddeľovač MIME objektov. Čiže miesto, kde sa jednotlivé objekty začínajú a kde končia. Na uvedenom príklade je vidno, kde presne sa začína textová časť a kde HTML časť.

1.2 Formát archívu mailových správ

Štandard RFC2822 definuje, ako má presne vyzerať jedna mailová správa. Nerieši však akým spôsobom sa majú maily ukladať a ani ako sa má pracovať so zoznamom správ. Nakoľko telo mailovej správy môže obsahovať ľubovoľný znak z ASCII tabuľky (s použitím MIME aj ďalšie 8-bitové znaky) ako aj ľubovoľný počet prázdnych riadkov, nie je možné použiť bezpečne žiadny reťazec znakov ako oddeľovač mailových správ.

Štruktúra jednej správy má striktnú a presnú špecifikáciu. Na rozdiel od toho, formát pre definovanie zoznamu správ nemá žiadnu pevne definovanú normu. Vo svete sa zaužívalo niekoľko rôznych formátov a ich variánt. Medzi najčastejšie používané formáty na uchovávanie viacerých mailových správ na disku patria *MBox* a *Maildir*.

1.2.1 Formát *Maildir*

Formát *Maildir* [9] vymyslel pán Daniel J. Bernstein pre mailový program qmail. Nakoľko sa formát ujal medzi mailovými klientmi ako aj IMAP servermi, dnes je už nejaká jeho varianta (často aj nekompatibilná s inými programami) používaná na ukladanie mailov od užívateľa. *Maildir*, ako už názov napovedá, používa používa celý adresár na ukladanie mailov. Konkrétne ukladá každý mail do samostatného súboru. Tým sa zbavuje problému oddeľovača jednotlivých mailov a rieši aj problém globálneho zamykania súborov, keď by napr. chceli dve aplikácie naraz editovať dva rôzne maily. Keďže každý mail je uložený v samostatnom súbore, nekladie sa žiadna réžia na presun mailu z jedného archívu do druhého. Formát je vďaka týmto vlastnostiam populárny medzi mailovými klientmi, ktoré používajú bežný ľudia.

[2]Konkrétne sa používa iba množina 64 znakov: malé a veľké písmená anglickej abecedy, číslice a znaky plus a lomka

Maildir má ale jednu obrovskú nevýhodu. Každý mail je uložený v samostatnom súbore a naviac v jednom adresári. Pri veľkom množstve mailov, ktoré sa v archíve mailových konferencii dokážu za dlhšie obdobie nazbierať, začne dochádzať k zvýšeniu prístupovej doby k jednotlivým súborom na disku. Výhody, ktoré *Maildir* ponúka, sa pri programe na archiváciu mailových konferencií veľmi nepoužijú a veľké množstvo mailov, ktoré tvoria zásadnú nevýhodu prevážia v neprospech *Maildir*-u. Z tohto dôvodu programy na archiváciu mailov z mailových konferencií formát *Maildir* vôbec nepoužívajú a ostávajú pri niektorej variante formátu *MBox*.

1.2.2 Formát *MBox*

Formát *MBox* umožňuje uložiť zoznam mailových správ do jedného súboru a jednotlivé správy ukladá za seba. Pred každou správou sa ešte nachádza špeciálny *úvodný* riadok. Tento riadok začína slovom `From`, ďalej nasleduje ␣ (medzera) a za ňou ďalší text. Niekedy sa tento riadok označuje aj ako *From_* riadok.

Z historických dôvodov existuje viac vzájomne nekompatibilných variánt formátu *MBox*. V rokoch 1995–1996 páni Daniel J. Bernstein, Rahul Dhesi a ďalší [10] zaviedli mennú konvenciu na rozlíšenie týchto rôznych variánt. V praxi bežne funguje, že novo vytvorené programy, ktoré ukladajú maily na disk, si vytvoria novú a s ostatnými programami nekompatibilnú variantu *MBox*-u.

Túto anarchiu rôznych variánt *MBox*-ov však zatiaľ nevyriešil žiadny RFC dokument. Existuje síce dokument RFC4155 [11], ktorý popisuje základný formát *MBox*. Avšak ten je iba informačného charakteru a nešpecifikuje presne, ako má *MBox* vyzerať. Ako jednu z možností uvádza odkaz na *MBox* formát, používaný programom qmail (konkrétne varianta *mboxrd*).

Nižšie spomenuté najbežnejšie varianty *mboxo*, *mboxrd*, *mboxcl* a *mboxcl2* majú spoločný formát úvodného riadku. Obsahujú za `From␣` mailovú adresu odosielateľa, ďalej medzerou oddelený dátum prijatia mailu a za ním môže nasledovať ešte ďalší text. Dátum je vo formáte, ktorý vracia funkcia `asctime()` v jazyku C (Príklad: `Wed Jun 30 21:49:08 1993`).

1.2.3 Varianta *mboxo*

Varianta *mboxo* používa ako oddeľovač jednotlivých správ prázdny riadok. Nasledujúca správa sa identifikuje podľa úvodného riadku, ktorý nasleduje za prázdnym riadkom. Aby sa predišlo omylu, keď súčasťou tela mailu bude prázdny riadok a za ním na novom riadku `From␣`, tak pred každý riadok začínajúci na `From␣` sa pridáva znak `>`.

1.2.4 Varianta *mboxrd*

Varianta *mboxrd* je modifikácia *mboxo* a rieši problém jednoznačnosti spätného rozkladu *MBox* archívu na jednotlivé správy. V prípade, ak sa v tele mailu nachádza riadok začínajúci na `>From␣`, nie je možné rozlíšiť či znak `>` bol v pôvodnom maili, alebo či bol pridaný až pri ukladaní do *MBox*-u. Varianta *mboxrd* tento problém rieši tak, že pred každý riadok, ktorý začína postupnosťou (aj prázdnou)

znakov `>` a za tým `From␣`, pridá na začiatok riadku znak `>`. Tým sa docieli jednoznačnosť spätného dekódovania. Znak `>` sa na začiatku riadku odstráni, ak je nasledovaný postupnosťou (hoci aj prázdnou) znakov `>` a za tým reťazcom `From␣`.

1.2.5 Varianty *mboxcl* a *mboxcl2*

Ďalšia varianta *mboxcl* je rovnaká ako *mboxrd*. Naviac ale pridáva do každej hlavičky mailu novú položku `Content-Length` s hodnotou veľkosť tela správy. Na nájdenie nasledujúcej správy sa teda dá použiť rovnaká metóda ako pri *mboxo* a *mboxrd* variantách. Je ale možné použiť aj práve novú položku `Content-Length` z hlavičky mailu. Niektoré programy okrem `Content-Length` pridávajú aj položku `Lines` s hodnotou počtu riadkov v tele správy. Táto informácia sa dá taktiež použiť na nájdenie nasledujúcej správy.

Varianta *mboxcl2* vychádza z *mboxcl*. Tá ale už nepridáva znak `>` pred riadky tela mailu začínajúce `From␣`. To znamená, že na nájdenie nasledujúcej správy v archíve sa dá použiť iba hodnota položky `Content-Length` alebo `Lines`. Program, ktorý rozumie iba *mboxrd* variante sa na archívy typu *mboxcl2* použiť už nedá.

1.2.6 Ďalšie varianty formátu *MBox*

Ďalšia z používaných variánt *MBox*-u je *MMDF*. Tá na rozdiel od iných nepridáva úvodný riadok pred každú správu a podobne ako *mboxcl2* nepridáva znak `>`. Namiesto prázdneho riadku, ako oddeľovač mailov, používa štyri znaky Ctrl+A (v ASCII tabuľke je to `0x01`).

Varianta *Eudora MBox* je modifikácia *mboxo*. Nepridáva ale prázdny riadok na koniec každého mailu. Okrem toho namiesto mailovej adresy odosielateľa v úvodnom riadku používa reťazec `???@???` a odstraňuje prílohy z MIME mailov. Prílohy ukladá oddelene od *MBox* súborov.

Často vyskytujúca sa varianta je aj *Netscape MBox*. Je to *mboxrd* s modifikovaným obsahom v úvodnom riadku.

2. Problémy pri spracovávaní mailov

V tejto kapitole sú rozobraté jednotlivé problémy, ktoré nastávajú pri spracovávaní mailov.

2.1 Problémy v mailových správach

Pri strojovom spracovávaní mailov dochádza k viacerým problémom. Najväčší problém je ten, že väčšina mailov nedodržuje štandardy. V tomto prípade sú na vine programy, ktoré maily generujú. Maily z takýchto programov sa vyskytujú v konferenciách dosť často a teda je potrebné, aby sa program na spracovávanie a archivovanie mailov s týmito mailami popasoval a čo najlepšie ich interpretoval.

2.1.1 Problémy s dátumom

Medzi častý problém patrí spracovávanie dátumu. Štandard RFC2822 [1] síce presne definuje v akom formáte má byť dátum uvedený, veľakrát sa ale v položke `Date` uvádza dátum aj v iných formátoch. Napríklad sú v položke vložené medzery navyše, pridané komentáre alebo uvedený slovný popis časovej zóny a to štandard nepripúšťa. Ako vhodné riešenie na spracovanie položky `Date` sa naskytuje použiť nejakú knižnicu, ktorá dokáže rozpoznať rôzne formáty dátumu a nie len formát definovaný v RFC2822.

Pri dátumoch sa vyskytuje aj ďalší problém. Dátum a čas do mailu vypĺňa program, ktorý zvyčajne nebeží na počítači, ktorý mail zároveň aj prijíma. Systémový čas sa môže o málinko líšiť a pri zle nakonfigurovanom stroji aj o interval v ráde niekoľko dní. A preto sa môže stať, že dátum, kedy mail prišiel nastal skôr ako dátum, ktorý má v položke `Date`. Našťastie, ak mail prechádza viacerými SMTP servermi, tak každý väčšinou pridá do hlavičky položku `Received`. V nej sa nachádza dátum prijatia mailu daným serverom. Čiže, ak v položke `Date` je uvedený zlý dátum, tak je možné použiť informáciu z položky `Received`.

2.1.2 Problémy s identifikátormi

Ďalší problém sa vyskytuje v položkách `Message-Id`, `In-Reply-To` a `References`. Tieto položky obsahujú identifikátory správ. Štandard RFC2822 presne definuje, ako majú byť hodnoty v týchto položkách uvedené, skutočnosť je ale často iná. V konferenciách sa vyskytujú maily, ktoré majú v týchto položkách naviac rôzne biele znaky, zbytočné zalamovanie riadkov, alebo nadbytočné komentáre. Ako oddeľovače identifikátorov sa používajú medzery, čiarky alebo bodkočiarky. Stane sa, že identifikátor nie je vôbec ani uzavretý medzi zátvorkami `<` a `>`. Stane sa, že špecifikovaný identifikátor je príliš krátky (3 znaky), čo je nevhodné na použitie ako jednoznačný identifikátor mailu.

Ako riešenie tohto problému sa naskytuje možnosť z položky najprv odstrániť všetky biele znaky a potom vybrať najväčšiu množinu disjunktných podreťazcov, pre ktoré platí, že začínajú znakom `<`, končia znakom `>`, neobsahujú vo vnútri

žiadny znak <, > a sú dlhé aspoň 5 znakov. Tým sa zaručí, že identifikátor je dostatočne dlhý a vyrieši sa problém s bielymi znakmi a rôznymi oddeľovačmi či komentármi. V prípade, že množina týchto reťazcov pre položku `Message-Id` je prázdna, je tu možnosť použiť samotnú hodnotu položky bez bielych znakov ako identifikátor. Dôvod je práve ten, že sa vyskytujú maily, ktoré nemajú identifikátor správne vnorený medzi zátvorkami < a >.

2.1.3 Problémy s mailovými adresami

Problém sa vyskytuje aj pri spracovávaní položiek `From`, `To` a `Cc`, ktoré obsahujú zoznam mailových adries s menami. Pri použití MIME je správne spracovanie takejto položky trochu zložité a viacero mailových programov, keď odpovedajú na mail a kopírujú obsah hlavičky, tak spravia rôzne chyby. Štandard umožňuje do týchto položiek pridávať vnorené komentáre, čo značne zkomplikuje správne rozdelenie mailových adries a im prislúchajúcich mien. Často sa preto stáva, že po správnom rozpoznaní mien z položiek, sú samotné mená zabalené viacnásobnými jednoduchými (') alebo dvojitými (") úvodzovkami. Preto je vhodné nadbytočné úvodzovky na začiatku a konci reťazca udávajúce meno odstrániť.

2.1.4 Problémy v tele mailových správ

Ďalší problém sa nachádza v texte mailu. Rozšírenie MIME umožňuje použiť rôzne znakové sady a viacero spôsobov kódovania týchto znakov do ASCII. Nie ale každý program, čo maily vygeneruje, nastaví správnu hodnotu použitej znakovej sady do položky `Content-Type`, atribútu `charset`. Stane sa aj, že informácia o použitej znakovej sade úplne chýba. Preto program, ktorý spracováva texty, musí vedieť znakovú sadu nejak detegovať, aby dokázal obsah čo najlepšie zobraziť.

Položka `Content-Type` slúži primárne na informáciu o tom akého typu je daná časť mailu. Pre mailové prílohy to má význam, rýchlo identifikovať či ide o PDF dokument, HTML stránku alebo zvukovú nahrávku. Často sa však vyskytujú maily, ktoré majú túto položku nastavenú zle.

Napríklad niekedy je pre PDF dokument položka `Content-Type` nastavená na `text/plain`. To ale udáva, že ide o obyčajný textový dokument a nie PDF – čo je zle. Stane sa aj, že položka `Content-Type` neprináša žiadnu hodnotu. Napríklad hodnota `application/octet-stream` nehovorí o obsahu absolútne nič.

V tom najhoršom prípade sa môže stať, že pre obyčajný textový mail, ktorý neobsahuje žiadne vnorené MIME objekty, je položka `Content-Type` nastavená na `multipart/mixed`.[1] To hovorí, že daná časť mailu je MIME objekt, ktorý obsahuje ďalšie objekty. Nakoľko programy na spracovávanie obsahu mailu interpretujú hodnoty začínajúce na `multipart/` špeciálne (nakoľko by malo ísť o MIME objekt), tak takéto zle vygenerované maily nedokážu poriadne ani zobraziť.

Ako riešenie sa naskytuje použitie nejakej systémovej knižnice na rozpoznávanie typu súboru podľa jeho obsahu (`shared-mime-info` alebo `libmagic1`). A potom ignorovať hodnotu z `Content-Type`, ak je zjavne nastavená zle.

Položky `Content-Type` a `Content-Disposition` obsahujú zoznam hodnôt. Štandardy definujú presne predpísaný spôsob, akým sa majú hodnoty kódovať a oddeľovať od seba. Programy, ktoré generujú maily sa nie vždy riadia týmito

[1] Takto zle vytvorené a nečitateľné maily rozposielajú Koleje a Menzy Univerzity Karlovy

pravidlami, čo spôsobuje opäť problémy s interpretáciu hodnôt. Najčastejšie sa jedná o chybu, že za poslednou hodnotou nasleduje ešte oddeľovač, ktorý štandard zakazuje. Ďalšia chyba spočíva v tom, že sa používa nesprávna hodnota na uloženie názvu prílohy, alebo že názov prílohy obsahuje nepovolené znaky (alebo znaky sú nesprávne zakódované). Tieto problémy sa dajú riešiť tak, že nadbytočné oddeľovače a nepovolené znaky sa buď budú ignorovať alebo sa nejak spracujú.

Pri spracovávaní obsahu mailu sa môže ešte vyskytnúť problém s oddeľovaním koncov riadkov. Nájdu sa maily, ktoré mixujú dokopy `CRLF` a `LF`. Niektoré programy majú fixne nastavený znak na koniec riadku, takže je dobré, aby sa konce riadkov pri spracovávaní mailov zmenili na jednotné (napr. `LF`).

Ďalší problém sa týka HTML častí. HTML sa v mailoch používa na pokročilé formátovanie textu. Na vygenerovanie HTML mailu sa vo veľkej väčšine používa niektorá z pokročilých HTML knižníc. A ich súčasťou je aj knižnica na vykonávanie kódu v jazyku JavaScript, čo má za následok, že maily sa stali nástrojom na šírenie rôzneho škodlivého softwaru, ktorý zneužíva chyby vo vykresľovaní HTML dokumentov. Za posledné roky bolo v rôznych vykresľovacích knižniciach nájdených veľké množstvo chýb, ktoré viedli k pádu knižnice alebo umožnili spustiť kód útočníka. Z týchto dôvodov je potrebné, aby program, ktorý maily spracováva, mal možnosť HTML časti nezobrazovať, alebo aby bezpečne previedol HTML dokument na čistý text, ktorý sa už nebude zobrazovať pomocou pokročilej HTML knižnice s možnosťou spúšťať JavaScript.

2.2 Problémy s *MBox* archívmi

Hlavný problém s *MBox* archívmi je ten, že existuje niekoľko medzi sebou nekompatibilných variánt, ktoré sa bežne používajú. Veľakrát sa dokonca jednotlivé varianty v rámci jedného súboru aj mixujú. Nastane to napr. vtedy, ak sa ručne zlepia za seba dva *MBox* súbory, ktoré sú rôznych variánt. Potom je problém s interpretáciou dát a program sa musí rozhodnúť či sa má riadiť položkou `Content-Length` alebo hľadať ďalší úvodný riadok. Ďalší problém je ten, ako sa má program zachovať, ak prečíta riadok, ktorý začína na `>From␣`. Musí sa rozhodnúť, či znak `>` bol pôvodnou súčasťou mailu, alebo či bol vložený umelo pri pridávaní mailu do *MBox* súboru.

2.2.1 Problém nájdenia nasledujúceho mailu

V prípade, že je dopredu jasné, o ktorú variantu *MBox* formátu sa jedná, tak čítanie jednotlivých mailov je jednoduché. Stačí sa riadiť pravidlami, ktoré definujú ako sa jednotlivé maily oddeľujú v tej *MBox* variante. Problém nastáva, ak nie je presne určený typ *MBox* varianty.

Najbežnejšie sa používajú varianty *mboxro*, *mboxrd*, *mboxcl* a *mboxcl2*. A preto sa táto práca bude ďalej zaoberať iba týmito variantami. Nakoľko varianty *mboxro*, *mboxrd* a *mboxcl* používajú na identifikáciu ďalšieho mailu úvodný riadok a zabezpečujú, že v tele mailu sa nebude žiaden riadok začínať rovnako ako úvodný, tak úvodný riadok jednoznačne určuje kde začínajú maily. V *MBox* súbore, ktorý je varianty *mboxcl2* sa môžu vyskytnúť aj riadky, ktoré začínajú rovnako ako

úvodný riadok. Je to preto, že varianta *mboxcl2* neupravuje telo mailu žiadnym spôsobom.

Na odlíšenie *mboxcl2* od zvyšných variánt sa dá použiť nasledujúci spôsob. Program bude čítať *MBox* archív postupne po riadkoch až kým nenarazí na prázdny riadok, ktorý oddeľuje hlavičku mailu od samotného tela. Ak v priebehu čítania hlavičky narazil na `Content-Length` alebo `Lines`, tak využije túto informáciu na nájdenie pozície ďalšieho mailu ako pri variante *mboxcl* a *mboxcl2*.

V prípade, že ďalší mail bude začínať úvodným riadkom, znamená to, že informácia v použitej položke bola správna a daný mail je pravdepodobne formátu *mboxcl* alebo *mboxcl2*. Môže sa stať, že položka nebola nájdená, alebo informácia bola chybná a na danej pozícii z použitej položky sa nezačína úvodný riadok.

V tomto prípade sa buď jedná o poškodený *mboxcl* či *mboxcl2* archív, alebo sa jedná o *mboxro* či *mboxrd* archív, ktorý má v mailoch naviac položku `Content-Length` s neznámou informáciou. V prípade, že je to poškodený *mboxcl2* archív, tak s tým sa veľa urobiť nedá. Dá sa skúsiť nájsť ďalší úvodný riadok a dúfať, že bude znamenať začiatok ďalšieho mailu.

V prípade, že sa jedná o *mboxro* alebo *mboxrd* (alebo aj *mboxcl*) archív s položkami `Content-Length`, ktoré sú navyše, tak v takomto archíve sa môže bez problémov začať hľadať ďalší úvodný riadok, ktorý bude udávať jednoznačný začiatok mailu.

Čiže v skratke, program bude interpretovať archív ako *mboxcl2*. V prípade, že špecifické položky nebudú dostupné alebo nebudú vykazovať správne informácie, program začne interpretovať daný mail ako *mboxrd*. Po prečítaní celého mailu sa opäť prepne do režimu *mboxcl2* a začne čítať ďalší mail.

Týmto spôsobom sa dá pomerne presne nájsť začiatok každého mailu, za predpokladu, že archív nie je veľmi poškodený. A nebude vadiť ani keď sa zmiešajú rôzne varianty, nakoľko sa každý jeden mail číta a interpretuje osobitne.

2.2.2 Problémy s riadkami začínajúcimi na `>From␣`

Druhý problém, ktorý treba vyriešiť pri čítaní *MBox* archívu, je zistiť či úvodný znak `>`, ktorý sa nachádza na riadku začínajúcim postupnosťou `>` a reťazcom `From␣` je súčasťou pôvodného mailu alebo nie. Žiaľ pri variante *mboxro* to nie je možné, nakoľko oba riadky `From␣` aj `>From␣` sa zmenia na `>From␣` a teda nie je medzi nimi možné rozlíšiť. Všeobecne nie je možné rozlíšiť ani medzi variantami *mboxcl* a *mboxcl2*, keďže obe sa líšia iba tým či znak `>` pridajú alebo nie. Takže všeobecne bez dodatočnej informácie to rozlíšiť nejde. Je asi viac pravdepodobné, že v originálnych mailoch sa bude nachádzať skôr riadok začínajúci na `From␣` ako na `>From␣`. Nech už sa zvolí možnosť odstrániť úvodný znak alebo nie, treba si uvedomiť jednu vec. Ak sa takýto riadok nachádza v textovej časti mailu, tak jeden znak `>`, ktorý sa nachádza na začiatku riadku toho veľa nepokazí. Väčšina mailov bude čitateľných aj keď na začiatku riadku bude chýbať znak `>` a aj keď bude navyše.

2.2.3 Problémy v úvodnom riadku

Pri problémoch so spracovaním jedného mailu podľa štandardu RFC2822 sa vyskytuje problém s dátumom. Problém je buď spôsobený programom na genero-

vanie mailov, ktorý použil nesprávny formát dátumu alebo chyba je v systéme a ten oznamuje zlý čas. Ako ďalšie riešenie tohto problému je možné použiť čas prijatia mailu, ktorý je uchovaný v úvodnom riadku, ktorý je súčasťou *MBox* súboru. Tento dátum je pridaný až keď sa mail doručí a v prípade, že používateľ dôveruje času vedenému na serveri, ktorý mailovú správu zaradil do archívu, tak môže výhradne použiť tento časový údaj. A vyhne sa tým problému, či odosielateľ mailu náhodou nemal problém s nastavením času. A ak už aj nie, tak je možné informácie z `Date` a `Received` položiek skombinovať spolu s dátumom z úvodného riadku a zistiť či sa nelíšia o príliš veľký interval.

Úvodný riadok má ale tiež problémy. Rôzne programy ho generujú rôzne a preto sa v ňom vyskytujú rôzne informácie. Takže prítomnosť mailovej adresy odosielateľa ako aj dátumu prijatia mailu treba brať s rezervou a nespoliehať sa na to. Taktiež v *MBox* archívoch sa nevyskytuje formát dátumu výhradne len podľa `asctime()`. Vhodné riešenie na získanie dátumu z úvodného riadku je opäť použiť nejakú knižnicu na rozpoznávanie rôznych druhov formátu dátumov v reťazci.

3. Existujúce implementácie

Problém archivácie mailov a sprístupnenie archívu cez internet pomocou webového prehliadača sa už snažili vyriešiť viacerí. V tejto kapitole spomeniem najznámejšie internetové služby, ktoré ponúkajú archiváciu mailov z verejných mailových konferencií.

3.1 MARC

MARC, skratka za Mailing list ARChives, je jedna z najznámejších a najstarších internetových služieb ktorá ponúka archivovanie mailov z verejných mailových konferencií. Existuje od roku 1996 a používa ju viacero projektov ako oficiálny archív. Patrí medzi ne užívateľské prostredie KDE. Podľa informácií, ktoré MARC zverejnil ku dňu 02. 04. 2014, majú v archíve 70 miliónov mailov z viac než 3 500 konferencií od siedmich miliónov rôznych ľudí. Každý mesiac pribudne do archívu približne 350 tisíc nových mailov. Oficiálna stránka služby je `http://marc.info/`.

MARC vytvára pre každú mailovú konferenciu samostatný archív. Umožňuje ale maily vyhľadávať naprieč všetkými archívmi a to podľa predmetu, alebo autora mailu. Jednotlivé maily je možné prezerať podľa mesiaca, v ktorom boli prijaté. MARC pre každý mesiac zoskupuje maily podľa predmetu a vytvára tak lineárne usporiadanie mailov, ktoré tvoria celistvé vlákno.


Search: Search Reset

[ ] List [ ] Subjects [ ] Authors [•] Bodies for list 'linux-kernel'

Set Page Width: [80] [90] [100] [120]

linux-kernel Linux-Kernel List @vger.kernel.org

2014-07-01 - 2014-08-01 (20663 messages)

2014-06-01 - 2014-07-01 (20546 messages)

Next Last

1.	2014-07-27	[11]	[RFC PATCH 1/1] rcu: Use rcu_gp_kthread_wake() to wak	linux-ker	Paul E. McKen
2.	2014-07-27	[1]	Re: questions regarding drivers/staging/iio/accel/sca	linux-ker	Jonathan Came
3.	2014-07-27	[3]	[RFC PATCH 1/1] ethtool: adding support for multiple_	linux-ker	John Fastaben
4.	2014-07-27	[2]	[PATCH] staging: vt6655: coding style: Fixed commenti	linux-ker	Greg KH
5.	2014-07-27	[20]	[RFC][PATCH] irq: Rework IRQF_NO_SUSPENDED	linux-ker	Rafael J. Wys
6.	2014-07-27	[1]	[PATCH] scsi: bfa: bfa_fcs_lport.c: Cleaning up miss	linux-ker	Rickard Stran
7.	2014-07-27	[3]	[BUG] nfs3_list_one_acl oops	linux-ker	Trond Myklebu
8.	2014-07-27	[3]	[PATCH] nfs3_list_one_acl(): check get_acl() result w	linux-ker	Trond Myklebu
9.	2014-07-27	[1]	[PATCH] scsi: bfa: bfa_fcbuild.c: Cleaning up missin	linux-ker	Rickard Stran
10.	2014-07-27	[1]	[PATCH] scsi: 3w-9xxx.c: Cleaning up missing null-te	linux-ker	Rickard Stran
11.	2014-07-27	[5]	[PATCH 3.15 000/109] 3.15.7-stable review	linux-ker	Greg Kroah-Ha
12.	2014-07-27	[6]	[PATCH 4/5] [SCSI] Do not use platform_bus as a paren	linux-ker	Greg Kroah-Ha
13.	2014-07-27	[2]	[PATCH 3.14 00/87] 3.14.14-stable review	linux-ker	Guenter Roeck
14.	2014-07-27	[2]	[PATCH 3.10 00/56] 3.10.50-stable review	linux-ker	Guenter Roeck
15.	2014-07-27	[2]	[PATCH 3.4 00/23] 3.4.100-stable review	linux-ker	Guenter Roeck
16.	2014-07-27	[1]	[PATCH] scsi: qla4xxx: ql4_mbx.c: Cleaning up missin	linux-ker	Rickard Stran
17.	2014-07-27	[10]	[PATCH v4 3/3] crypto: Add Allwinner Security System_	linux-ker	Herbert Xu
18.	2014-07-27	[3]	[PATCH 00/10] of: Core devicetree changeset support	linux-ker	Grant Likely
19.	2014-07-27	[1]	[PATCH] Documentation: laptops: freefall.c: Cleaning	linux-ker	Rickard Stran
20.	2014-07-27	[1]	RE: [PATCH v2] mmc: dw_mmc: Make sure we don't get st	linux-ker	Seungwon Jeon
21.	2014-07-27	[1]	Re: [PATCH v3] iio: add support of the max5821	linux-ker	Jonathan Came
22.	2014-07-27	[2]	[PATCH v2] iio: hid-sensor-magn-3d: Fix build warning	linux-ker	Jonathan Came
23.	2014-07-27	[4]	[PATCH v2 15/25] amdkfd: Add kernel queue module	linux-ker	christian.koe
24.	2014-07-27	[9]	General flags to turn things off (getrandom, pid look	linux-ker	David Drysdal
25.	2014-07-27	[2]	[PATCH 0/3] ARM: EXYNOS: Fix Exynos5410 boot	linux-ker	Andreas FÃ¤rb
26.	2014-07-27	[1]	[PATCH 3/3] ARM: EXYNOS: Add support for Exynos5410 P	linux-ker	Andreas FÃ¤rb
27.	2014-07-27	[1]	[PATCH 2/3] ARM: dts: exynos: Add PMU to Exynos5410	linux-ker	Andreas FÃ¤rb
28.	2014-07-27	[1]	[PATCH 1/3] Documentation: devicetree: Document exyno	linux-ker	Andreas FÃ¤rb
29.	2014-07-27	[1]	Re: [PATCH] crypto: rng.c: Cleaning up missing null-t	linux-ker	Rickard Stran
30.	2014-07-27	[9]	[PATCH 11/11] seccomp: Add tgid and tid into seccomp_	linux-ker	David Drysdal

Z lineárneho usporiadania mailov vo vlákne sa už ale čitateľ nedozvie, ktorý mail je odpoveď na ktorý a nezistí ani koľko odpovedí prišlo na zvolený mail.

Najväčší problém, ktorým MARC trpí je samotné zobrazovanie mailov. Nedokáže správne zobraziť UTF-8 znaky a ani *quoted-printable* kódovanie v hlavičke

mailu. Prakticky to spôsobí nemožnosť vyhľadávať maily v archíve podľa predmetu či odosielateľa.

MARC nerozpozná *quoted-printable*:

[prev in list] [next in list] [prev in thread] [next in thread]

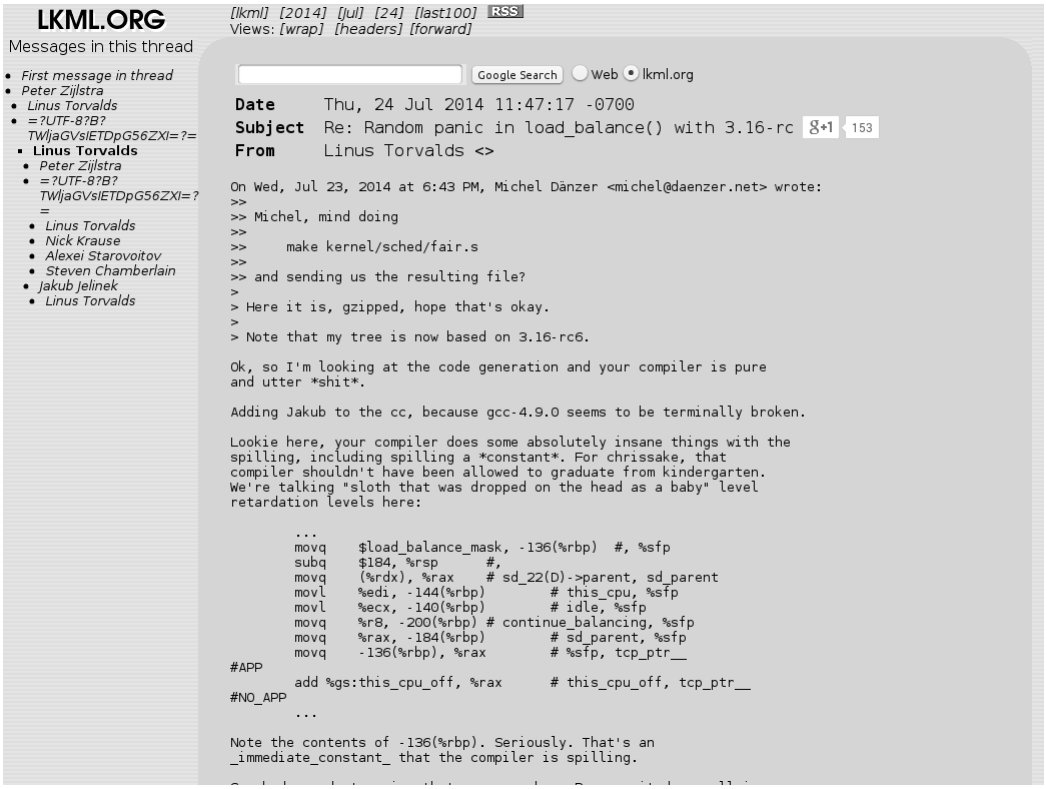
List: kopete-devel
Subject: kopete and gcc4.7
From: Pali =?iso-8859-1?q?Roh=E1r?= <pali.rohar () gmail ! com>
Date: 2013-01-16 14:08:19
Message-ID: 201301161508.19548 () pali
[Download message RAW]

[Attachment #2 (multipart/signed)]

Oproti tejto veľkej nevýhode, ktorá znemožňuje rozumne vyhľadávať a čítať maily s inými znakmi ako z ASCII tabuľky, MARC má pomerne dobrú podporu mailových príloh a čitateľovi umožňuje stiahnuť prílohy konkrétneho mailu na disk.

3.2 LKML.ORG

Ako ďalšiu službu na prezeranie mailov, spomenieme LKML.ORG. Táto služba na rozdiel od iných je výhradne určená iba pre mailovú konferenciu LKML (Linux Kernel MailingList) projektu linuxového jadra. Nearchivuje maily zo žiadnych iných konferencií a aj z toho dôvodu obsahuje niektoré vlastnosti špecifické pre účely vývoja Linuxu. Služba sídli na adrese `http://lkml.org/`.



Konferencia LKML je verejná, ale platia pre ňu určité podmienky. Je zakázané posielať maily, ktoré sú vo formáte MIME. Teda akékoľvek maily, ktoré sú komprimované, obsahujú prílohy, či iné časti ako čistý text [12]. Dôvod je ten,

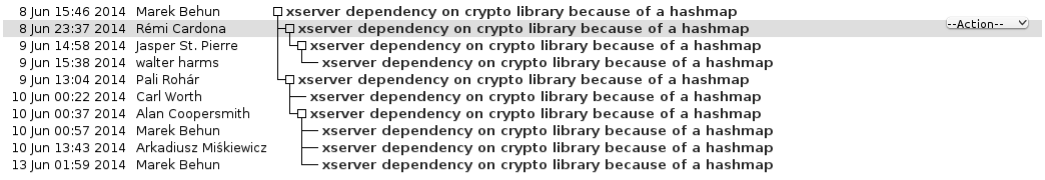
že mailové prílohy, alebo inak komprimovaný obsah, ktorý nie je textovej podoby, nie je možné ľahko citovať. Pri revízií zdrojového kódu potom nie je možné odpovedať na ktorúkoľvek časť mailu.

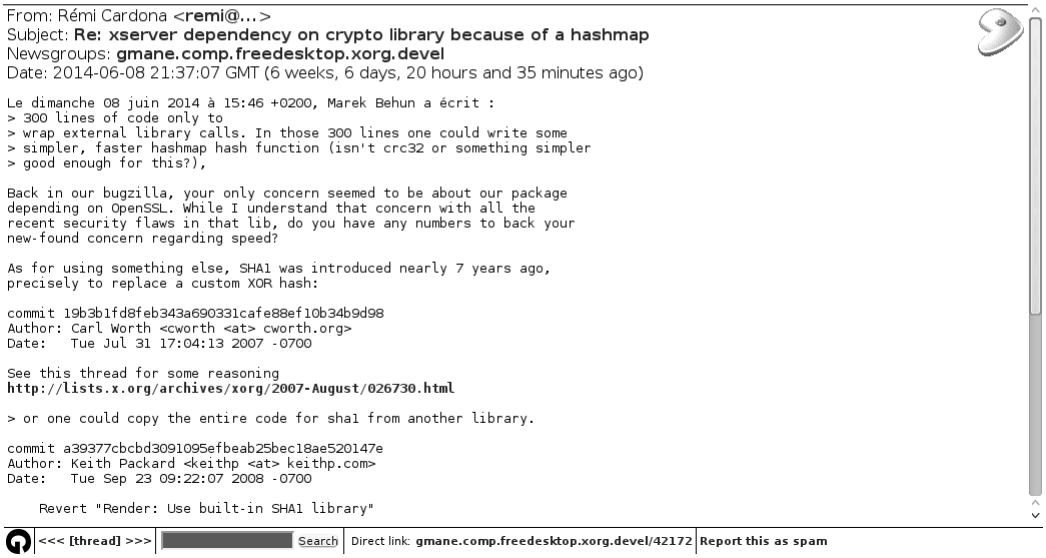
Webové rozhranie služby LKML.ORG predpokladá, že maily budú tieto podmienky spĺňať. Z tohoto dôvodu webové rozhranie automaticky odstraňuje prílohy z mailov a obsah chápe ako čistý obyčajný text, v ktorom sa môže nachádzať zdrojový kód vo formáte Unified diff.

Medzi výhody rozhrania služby LKML.ORG patrí podpora zobrazovania vlákien na ľavej strane obrazovky pre práve zobrazovaný mail. Takisto aj schopnosť detegovať Unified diff v tele mailu a umožniť ho stiahnuť ako externý súbor. Ako pri ostatných službách, medzi nevýhody patrí zlá podpora MIME, národných znakových sád a rôznych kódovaní.

3.3 Gmane

Ďalšia pomerne známa služba na archiváciu verejných mailových konferencií je Gmane. V júli 2014 mala archivovaných takmer 15 tisíc mailových konferencií a dokopy spracovaných viac než 142 miliónov mailových správ. Táto služba o sebe hovorí, že má pokročilé filtrovanie spamu, detekciu mailových správ, ktoré boli poslané naraz do viacerých konferencií a v reálnom čase zvláda indexovať obsah mailov na neskoršie prehľadávanie. Služba je na adrese `http://gmane.org/`.





Čo sa týka funkčnosti, tak táto služba na veľmi dobrej úrovni podporuje zoskupovanie mailov do vlákien. Okrem toho vlákna zobrazuje vo webovom prehliadači ako stromy a teda je dobre vidno, ktorý mail je odpoveďou na ktorý. Avšak nájdu sa konverzácie, kde je strom vybudovaný zle a odpovede sú pripojené pod koreň stromu namiesto správneho miesta. Služba zvláda zobraziť aj iné znaky ako

z ASCII tabuľky a nemá veľké problémy ani s národnými znakmi v mailových hlavičkách, čo je oproti MARC veľké plus. Na druhej strane táto služba dokáže zobraziť iba textové časti mailov, so zložitejšími MIME objektami má problémy a s prílohami pracovať nevie vôbec. Nezobrazí ani informáciu o tom, že mail pôvodne obsahoval nejakú prílohu.

3.4 Pipermail

Pipermail nie je webová služba ale software, ktorý slúži na archiváciu a zobrazenie mailov. Teda implementuje potrebné funkcie pre webový server na spojazdnenie služby. Pipermail je súčasťou softwaru Mailman, ktorý implementuje na serveri službu mailovej konferencie. Prijíma a rozposiela maily a okrem toho ich aj posúva ďalej pre archiváciu v Pipermail-i.

Keďže Pipermail je súčasťou Mailman-a, tak je používaný skoro na každom serveri, ktorý používa software Mailman. Používa sa teda na verejné aj privátne mailové konferencie a je dosť rozšírený.

Pipermail má nejakú podporu pre vlákna, ale nedosahuje ani zďaleka kvalít Gmane. Maily zvyčajne archivuje po mesiacoch a teda s vláknami na rozhraní mesiacov si veľmi poradiť nevie. Má problémy aj s veľkými a hlbokými vláknami.

Pipermail štandardne podporuje iba textové časti mailov, ale má aj základnú podporu pre sťahovanie príloh. Maily zobrazuje podľa predmetov a v prípade, že predmet je strašne dlhý, tak orientácia na stránke so zoznamom mailov nie je jednoduchá. V tomto prípade, si užívateľ musí zmenšiť písmo alebo použiť väčší monitor.

July 2014 Archives by thread

- **Messages sorted by:** [subject] [author] [date]
- **More info on this list...**

Starting: *Tue Jul 1 09:12:20 CEST 2014*
Ending: *Sun Jul 27 12:59:02 CEST 2014*
Messages: 30

- [Mailman-Developers] Disabling SELinux *Aurelien Bompard*
- [Mailman-Developers] Adding CSS to mailman archives *Andrew Daviel*
  - [Mailman-Developers] Adding CSS to mailman archives *Mark Sapiro*
    - [Mailman-Developers] Adding CSS to mailman archives *Andrew Daviel*
- [Mailman-Developers] DMARC discussions at IETF, WG likely to form *Stephen J. Turnbull*
- [Mailman-Developers] [CLI Project] Initial Commit on Command Line Shell *Rajeev S*
- [Mailman-Developers] Mailman3 user management *ML mail*
- [Mailman-Developers] gsoc ci tool updates *varun sharma*
- [Mailman-Developers] [CLI Project] Create object and Delete object functionalities *Rajeev S*
- [Mailman-Developers] user table in Mailman3 with PostgreSQL *ML mail*
  - [Mailman-Developers] user table in Mailman3 with PostgreSQL *Daniel Kahn Gillmor*
    - [Mailman-Developers] user table in Mailman3 with PostgreSQL *ML mail*
      - [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL *Daniel Kahn Gillmor*
      - [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL *ML mail*
      - [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL *Barry Warsaw*
      - [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL *ML mail*
      - [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL *ML mail*
- [Mailman-Developers] listarchiver table missing from postgresql schema *ML mail*
  - [Mailman-Developers] listarchiver table missing from postgresql schema *Aurelien Bompard*
    - [Mailman-Developers] listarchiver table missing from postgresql schema *ML mail*
- [Mailman-Developers] [CLI Project] Completed Subscribe and Unsubscribe commands *Rajeev S*
- [Mailman-Developers] [CLI Project] Update Preference Command added *Rajeev S*
- [Mailman-Developers] [CLI Project] Command Parsing method switched to YACC *Rajeev S*
- [Mailman-Developers] Add moderators from the command line *ML mail*
  - [Mailman-Developers] Add moderators from the command line *Barry Warsaw*
    - [Mailman-Developers] Add moderators from the command line *ML mail*
- [Mailman-Developers] REST API documentation (list of all possible parameters) *ML mail*
  - [Mailman-Developers] REST API documentation (list of all possible parameters) *Barry Warsaw*
    - [Mailman-Developers] REST API documentation (list of all possible parameters) *ML mail*
- [Mailman-Developers] [CLI Project] Switched command Parsing to YACC and LEX *Rajeev S*

[Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL

Barry Warsaw barry at list.org
Wed Jul 16 19:20:17 CEST 2014

- Previous message: [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL
- Next message: [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL
- **Messages sorted by:** [date] [thread] [subject] [author]

```
On Jul 16, 2014, at 08:20 AM, 'ML mail' via barry wrote:

>Thanks for the details but actually I was now only asking in which table are
>stored the e-mail addresses of a mailing's list subscribers? I thought they
>would be stored in the "user" table but I can't see any e-mail addresses
>there, neither in the member table.

Correct.  It's useful to understand the model.

http://pythonhosted.org//mailman/src/mailman/docs/8-miles-high.html#user-model

Users have an id and a display name.  Addresses are separate objects linked to
at most one user.  A user can be linked to many addresses.  Members associate
an address/user with a mailing list[*] under a specific role.  Rosters are
"magical" objects which perform queries to answer questions like "all the
regular delivery members of a mailing list" or "all of the list's moderators".

So the information you're looking for isn't captured in a single table.
You'll have to do a query to get the email addresses of a mailing list's
subscribers.  Take a look at the roster implementations for details.

Cheers,
-Barry

[*] A member links a user only if that user has a preferred address.
```

- Previous message: [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL
- Next message: [Mailman-Developers] OT: Re: user table in Mailman3 with PostgreSQL
- **Messages sorted by:** [date] [thread] [subject] [author]

More information about the Mailman-Developers mailing list

Program Pipermail zvykne byť niekedy na serveroch nahradzovaný programami Hypermail alebo Mhonarc. Tieto programy majú veľmi podobné vlastnosti a majú takmer totožné rozhranie ako Pipermail až na to, že pridávajú podporu pre maily vo formáte HTML.

3.5 Ďalšie služby

MARC a Gmane nie sú jediné internetové služby, ktoré ponúkajú archiváciu mailov z verejných mailových konferencií. Ďalšie služby majú podobné vlastnosti ako vyššie uvedené a veľmi sa od nich nelíšia. V nasledujúcej tabuľke sú uvedené názvy pomerne známych služieb, ktoré archivujú maily z verejných konferencií a odkazy na ich webové stránky.

MARC	`http://marc.info/`
Gmane	`http://gmane.org/`
The Mail Archive	`http://www.mail-archive.com/`
MarkMail	`http://markmail.org/`
osdir.com mailing list archive	`http://osdir.com/ml/`
Nabble	`http://www.nabble.com/`
Spinics	`http://www.spinics.net/lists/`

4. Návrh programu

V tejto kapitole sú popísané interné formáty, ktoré sa používajú na uchovávanie mailov. Ďalej je popísaný algoritmus, ktorý zobrazuje vlákna. Samotný program je navrhnutý tak, že spracováva maily po jednom a ukladá si ich na disk. Už pri spracovávaní si jednotlivé maily zaraďuje do vlákien.

4.1 Formát na uchovanie jedného mailu

Nakoľko je formát mailu podľa dokumentu RFC2822 [1] príliš komplikovaný, program v tejto práci používa vlastný formát na uchovávanie a spracovanie mailov. Medzi hlavné nevýhody spracovávania mailov podľa formátu RFC2822 patria: Nerešpektovanie štandardu veľkou väčšinou mailových programov, nutná podpora pre rôzne typy znakových sád a kódovaní, pomalé dekódovanie rôznych častí mailu či veľká redundancia niektorých položiek v hlavičke a iných dát v štruktúre mailu. Pôvodný štandard RFC822 [2] nepočítal s binárnymi dátami alebo prílohami v mailoch čo taktiež značne komplikuje spracovávanie príloh, či binárnych častí mailu.

Z týchto dôvodov a hlavne kvôli komplikovanosti štandardu RFC2822 sa pre účely tejto práce používa vlastný interný formát na uchovanie mailov. Tento formát sa skladá z textovej hlavičky, ktorá je v kódovaní UTF-8. Hneď za hlavičkou nasledujú binárne dáta, ktoré tvoria jednotlivé časti mailu. Podobne, ako v štandarde MIME, aj v tomto formáte tvorí jeden mail strom jednotlivých častí.

4.1.1 Hlavička

Príklad hlavičky:

```
Parts:
 0 0 0 message message/rfc822
 0/0 0 0 alternative multipart/alternative
 0/0/0 0 41 view text/plain
 0/0/1 41 510 view text/html
 0/0/2 551 42 view text/plain-from-html
Part:
 0
From:
 address1@example.org Odosielateľ
To:
 address2@example.org Prijímateľ
Id:
 293bef87319b430af5a423d829a1918b1c15ab8d@nohost
Date:
 1403605054
Subject:
 Toto je predmet mailovej správy
Data:
```

Hlavička obsahuje zoznam dvojíc, ktorú tvoria kľúč a jeho hodnota. Názov kľúča je vždy na jednom samostatnom riadku, končí znakom dvojbodka a nesmie začínať znakom medzera. Všetky riadky, ktoré začínajú znakom medzera tvoria hodnotu posledného špecifikovaného kľúča. Hodnota kľúča môže byť teda viacriadková a úvodná medzera, ktorá sa vyskytuje na začiatku každého riadku, do hodnoty nepatrí. Nie je zakázané, aby hodnota kľúča bola prázdna a to aj spôsobom, že nebude špecifikovaná (Po riadku s názvom kľúča bude nasledovať ďalší riadok s názvom kľúča). Koniec riadku tvorí vždy práve znak `LF` podľa ASCII tabuľky, ktorý má hexadecimálnu hodnotu `0x0A`. Tento znak sa nesmie nikde inde v hlavičke vyskytovať, ináč sú povolené všetky znaky z kódovania UTF-8. Hlavička musí začínať kľúčom `Parts` a na poslednom riadku hlavičky sa musí nachádzať názov kľúča `Data`. Ten už bude bez hodnoty.

V programe sa používajú tieto kľúče: `Parts`, `Part`, `From`, `To`, `Cc`, `Reply`, `References`, `Id`, `Date`, `Subject`, `Data`. Všetky ostatné sú ignorované.

4.1.2 Kľúč `Parts`

Kľúč `Parts` tvorí zoznam jednotlivých častí mailu. Samotné dáta týchto častí mailu sa nachádzajú až za hlavičkou.

Príklad riadku `Parts`:

```
0/0/1 47 512 view text/plain file.txt Information about file
```

Každý riadok obsahuje päť až sedem hodnôt oddelených medzerou. Okrem poslednej siedmej hodnoty, žiadna iná nesmie obsahovať medzeru. Šiesta a siedma hodnota nemusí byť špecifikovaná, pričom ak je siedma, musí byť aj šiesta.

- Prvá hodnota je názov časti mailu. Nakoľko časť tvorí nejaký vrchol v strome častí mailu, názov časti presne špecifikuje vrchol. Konkrétne cestu z koreňa do vrchola. Každý vrchol obsahuje usporiadaný zoznam synov a tí sú očíslovaní nezápornými číslami. Koreň je špeciálny prípad, ktorý nie je synom žiadneho vrchola a je označený číslom 0. Názov tvorí jednoznačný zoznam vrcholov na ceste z koreňa a vrcholy sú oddelené znakom lomka. Napr. názov `0/0/2` udáva cestu: koreň, prvý syn, prvý syn, tretí syn. A reprezentuje časť mailu, ktorú tvorí posledný spomenutý vrchol.

- Druhá hodnota je nezáporné číslo a udáva pozíciu začiatku dát, ktoré tvoria tú konkrétnu časť mailu. Pozícia je relatívna od konca hlavičky. V prípade, že časť neobsahuje žiadne dáta, hodnota je `0`.

- Tretia hodnota je opäť číslo a určuje veľkosť dát. Ak časť dáta neobsahuje, potom je uvedená hodnota `0`.

- Štvrtá hodnota je reťazec, ktorý určuje typ časti. Možné typy sú: `message`, `multipart`, `alternative`, `view`, `attachment`. Typ `view` predstavuje dátovú časť mailu, ktorá by sa mala užívateľovi zobraziť a typ `attachment` predstavuje prílohu – dátovú časť, ktorá je zvyčajne väčšia a užívateľ by si ju mal vedieť stiahnuť. Ostatné typy `message`, `multipart` a `alternative` predstavujú časti, ktoré neobsahujú žiadne dáta, iba ďalších synov čo tvoria podstromy. Typ `alternative` predstavuje vrchol v strome častí, pričom všetci jeho

synovia tvoria alternatívy, ktoré sú medzi sebou zameniteľné (podobne ako keď je v MIME `Content-Type` položke uvedené `multipart/alternative`). Typy `multipart` a `message` predstavujú vrcholy, ktoré tvoria ďalšie podstromy. Obsahujú iba ďalších synov a žiadne dáta (podobne ako objekt MIME typu `multipart`). Typ `message` naviac predstavuje kompletnú mailovú správu a teda k tomu vrcholu sa viažu aj ďalšie údaje z mailových hlavičiek. V tomto prípade sa vyžaduje aby boli špecifikované aj ďalšie kľúče, ktorých význam je uvedený nižšie v ďalšej sekcii. Je priam vhodné aby koreň bol typu `message`.

- Piata hodnota určuje MIME-Type dát, ktoré časť obsahuje, poprípade hodnotu položku `Content-Type` z mailu vo formáte MIME.
- Šiesta hodnota je názov súboru, ktorý časť predstavuje a siedma je popis súboru. Posledné dve hodnoty dávajú zmysel iba pre dátové časti.

4.1.3 Ostatné kľúče

V prípade, že mail v tomto formáte obsahuje jednu alebo viac častí typu `message`, potom pre každú z nich musí existovať samostatný kľúč `Part`. Hodnota kľúča `Part` je názov konkrétnej časti mailu, ktorá je typu `message`. Ďalej, ak sa vyskytujú kľúče `From`, `To`, `Cc`, `Reply`, `References`, `Id`, `Date` alebo `Subject`, viažu sa k zvolenej časti v poslednom špecifikovanom kľúči `Part`.

Kľúče pre danú časť typu `message`:

Názov kľúča	*Význam hodnoty kľúča*
`From`	Odosielatelia mailovej správy
`To`	Prijímatelia mailovej správy
`Cc`	Prijímatelia kópie mailovej správy
`Reply`	Identifikátory správ z položiek `In-Reply-To`
`References`	Identifikátory správ z položiek `References`
`Id`	Jednoznačný identifikátor správy (napr. z položky `Messsge-Id`), je bez zátvoriek `<`, `>` a nesmie obsahovať znak lomka `/`
`Date`	Počet sekúnd odo dňa 01. 01. 1970, kedy bola správa vytvorená[1]
`Subject`	Predmet správy

[1] podľa časovej zóny UTC bez priestupných sekúnd

Formát hodnôt `From`, `To` a `Cc` nie je rovnaký ako v mailoch podľa RFC2822. Hodnota týchto kľúčov obsahuje ako prvé mailovú adresu (ktorá nesmie obsahovať medzeru) a za ňou môže nasledovať meno, ktoré je oddelené medzerou. Meno sa ale v kľúči nachádzať nemusí a teda nie je povinné.

4.1.4 Dáta

Dátové časti správ sú uložené hneď za hlavičkou. Binárne dáta nie sú žiadnym spôsobom kódované, nepoužíva sa žiadny prevod dát do ASCII hodnôt (ani *qouted-printable* ani *Base64* ako pri RFC2822 mailoch). Všetky textové časti sú uložené v UTF-8. V prípade, že originálny mail bol napísaný v inej znakovej sade, tak sa jeho textové časti patrične skonvertujú do UTF-8.

4.2 Formát na uchovanie zoznamu mailov

Formát *MBox* je viazaný na maily vo formáte RFC2822. Teda nie je ho možné použiť na uchovávanie zoznamu mailov, ktoré sú uložené v internom formáte použitom v tejto práci. Z tohoto dôvodu sa pre účely tejto práce používa vlastný interný formát na uloženie zoznamu mailov. Interný formát je veľmi jednoduchý, spája maily (ktoré sú v internom formáte) za seba a pred každý mail uloží 32-bitové binárne číslo. Toto číslo určuje veľkosť mailu, ktorý za číslom nasleduje.

Použitím presného popisu veľkosti mailu odpadáva problém úvodného riadku, ako v *MBox*-e, ako aj problém odstraňovania znakov `>` zo začiatku riadkov tela mailu. Nakoľko je interný formát pre jeden mail binárny (lebo môže obsahovať binárne dátové časti), tak nevadí, že veľkosť mailu je uvedená v tomto internom formáte pre zoznam mailov taktiež binárne. Aplikácie na spracovanie tohoto interného formátu musia tak či tak vedieť pracovať s binárnym obsahom (a nie len so znakmi z ASCII tabuľky).

Veľkosť mailu, ktorá je uvedená pred samotným mailom, umožňuje programu na spracovávanie tohoto formátu aby vedel veľmi rýchlo, preskočiť aktuálny mail a začať spracovávať ďalší. Formát *MBox* toto nedokáže, nakoľko musí prečítať celý mail aby našiel ďalší úvodný riadok, alebo v prípade *mboxcl* variánt musí správne interpretovať `Content-Length` z hlavičky mailu.

4.3 SQL databáza

Na niektoré operácie s mailami vyššie uvedené interné formáty nestačia. Program na archiváciu a zobrazenie mailov sa potrebuje veľmi rýchlo dostať k niektorému konkrétnemu mailu, ako aj zistiť všetky maily, ktoré patria do istého vlákna. Potrebuje taktiež vedieť aj nejak zobraziť graf, ktorý tvoria `In-Reply-To` a `References` položky. V neposlednom rade potrebuje aj vedieť odfiltrovať maily, ktoré spĺňajú určité podmienky (napr. majú konkrétneho odosielateľa alebo predmet).

Úplne bežne sa stáva, že nejaký mail, ktorý je v archíve, je odpoveďou na iný mail, ktorý už v archíve nie je. Stane sa aj, že sa nájde väčšia množina mailov, ktoré sú všetky odpovede na mail, ktorý v archíve nie je. V takomto prípade je dobré, aby program vedel dať dokopy takéto maily do jedného vlákna a rozumne zobrazil informáciu, že maily sú odpovede na mail, ktorý nie je archivovaný. Takýto mail, ktorý v archíve chýba budeme ďalej nazývať *implicitný* mail.

Ďalej sa stáva, že niekedy informácia o tom, že mail je odpoveď na niektorý iný, úplne chýba. V takomto prípade je dobré skúsiť nájsť mail s podobným predmetom a priradiť ho do vlákna, kde sa takýto mail s podobným predmetom nachádza.

Z týchto dôvodov si táto práca uchováva externe aj ďalšie informácie pre každý mail. Informácie využíva k tomu, aby sa dali veľmi rýchlo nájsť maily, ktoré spĺňajú niektoré vlastnosti a aby bolo možné veľmi rýchlo pridať do archívu nové maily a preskočiť duplicity.

Konkrétne je potrebné uložiť pre každý mail aspoň tieto informácie:

- jednoznačný identifikátor mailu – `Message-Id`
- normalizovaný názov predmetu bez začiatočných a koncových bielych znakov a bez začiatočných reťazcov `RE`, `FWD`, ...

- názov súboru a pozíciu v ňom, kde je mail na disku uložený
- informácia, či je mail skutočne archivovaný alebo je iba implicitný
- vlákno do ktorého mail patrí a či je odpoveďou na niektorý iný mail
- identifikátory všetkých ostatných mailov, na ktoré je tento mail odpoveďou
- spracované mailové adresy s menami odosielateľov a prijímateľov mailu

Vyššie uvedené informácie sa uchovávajú v relačnej SQL databáze. Podporované sú implementácie *SQLite* a *MySQL*. Nižšie je uvedený zoznam SQL tabuliek a čo sa v nich presne uchováva. Tabuľky sú navrhnuté tak, aby bolo možné na potrebné dopyty v implementácii *SQLite* odpovedať veľmi rýchlo (s použitím indexov na niektorých stĺpcoch). Boolean hodnoty sú v SQL tabuľkách uložené ako celé čísla 0 a 1. Dátumy sú uložené ako celé čísla udávajúce počet sekúnd odo dňa 01. 01. 1970 (v časovej zóne UTC bez priestupných sekúnd).

4.3.1 SQL tabuľka `emails`

Tabuľka `emails` uchováva základne informácie o každom maili.

SQL tabuľka `emails`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor správy (primárny kľúč)
`messageid`	Hodnota položky `Message-Id` v správe (jedinečná)
`date`	Dátum správy v sekundách (je nulový pre implicitné maily)
`subjectid`	Normalizovaný predmet mailu uložený ako odkaz do tabuľky `subjects`
`subject`	Pôvodný predmet uložený ako textový reťazec
`treeid`	Číselný identifikátor vlákna, do ktorého mail patrí
`list`	Názov súboru, v ktorom je mail uložený
`offset`	Pozícia v súbore, kde je mail uložený
`implicit`	Boolean hodnota predstavujúca, či je mail implicitný alebo nie
`hasreply`	Boolean hodnota udávajúca, či daný mail má nejaké hodnoty v položke `In-Reply-To`

4.3.2 SQL tabuľka `trees`

Tabuľka `trees` uchováva predpočítané informácie o každom vlákne. Jedno vlákno obsahuje maily, ktoré reprezentujú súvislý komponent grafu všetkých mailov. Hodnoty sú predpočítané z ostatných tabuliek kvôli rýchlejšiemu prístupu k týmto informáciám.

SQL tabuľka `trees`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor vlákna (primárny kľúč)
`emailid`	Odkaz na mail do tabuľky `emails`, ktorý tvorí koreň v zostavenom strome daného vlákna
`date`	Najmenší nenulový dátum v sekundách zo všetkých mailov, ktoré patria do daného vlákna
`count`	Počet mailov, ktoré patria do daného vlákna

4.3.3 SQL tabuľka `replies`

Tabuľka `replies` uchováva, aké hodnoty majú maily v položkách `In-Reply-To` a `References`. Reprezentuje hrany grafu tvoreného všetkými mailami v archíve.

SQL tabuľka `replies`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor riadku (primárny kľúč)
`emailid1`	Odkaz na mail do tabuľky `emails`
`emailid2`	Odkaz na mail do tabuľky `emails`, ktorý je uvedený v položke stĺpca `type` mailu `emailid1`
`type`	Číselná hodnota, ktorá udáva či sa jedná o položku `In-Reply-To` (hodnota 0) alebo `References` (hodnota 1)

4.3.4 SQL tabuľka `subjects`

Tabuľka `subjects` obsahuje normalizovaný predmet z každej správy. Ten sa dá využiť na nájdenie správ, ktoré majú podobné originálne predmety.

SQL tabuľka `subjects`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor predmetu (primárny kľúč)
`subject`	Normalizovaný predmet (jedinečný)

4.3.5 SQL tabuľka `address`

Tabuľka `address` obsahuje dvojice mailovú adresu a mena, ktoré prislúcha adrese.

SQL tabuľka `address`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor adresy (primárny kľúč)
`email`	Mailová adresa
`name`	Meno

4.3.6 SQL tabuľka `addressess`

Tabuľka `addressess` uchováva všetky informácie z položiek `From`, `To` a `Cc`.

SQL tabuľka `addressess`:

Názov stĺpca	*Význam hodnoty*
`id`	Číselný identifikátor riadku (primárny kľúč)
`emailid`	Odkaz na mail do tabuľky `emails`
`addressid`	Odkaz na meno a mailovú adresu do tabuľky `address`, ktorá udáva názov položky zo stĺpca `type` pre mail `emailid`
`type`	Číselná hodnota, ktorá udáva či sa jedná o položku `From` (hodnota 0), `To` (hodnota 1) alebo `Cc` (hodnota 2)

4.4 Adresár *index*

Nakoľko sa dáta jedného mailového archívu uchovávajú na rôznych miestach (súbory, databáza), je potrebné niekde špecifikovať, kde sa dáta nachádzajú. Z tohoto dôvodu dáva si význam na disku vyhradiť jeden adresár, v ktorom budú všetky potrebné súbory uložené. Implementácia *SQLite* uchováva databázu v jednom súbore a nedokončené transakcie a pomocné súbory v rovnakom adresári, kde je aj databázový súbor. Teda v prípade použitia *SQLite* je priam vhodné uložiť databázový súbor tiež do adresára, v ktorom budú ostatné súbory archívu.

Z týchto dôvodov je jeden archív s mailami uložený vo svojom vlastnom adresári, ďalej označený ako *index*. V tomto adresári je hlavný konfiguračný súbor `config` a určuje rôzne nastavenia, špeciálne použitú databázu a prihlasovacie údaje k nej. Do konfiguračného súboru je možné uložiť aj nastavenie pre ďalšie aplikácie. Detaily konfiguračného súboru sú popísané v užívateľskej dokumentácií.

Jednotlivé maily sú uložené v súboroch, ktoré sú v adresári *index* a používa sa pre ne interný formát popísaný v predchádzajúcej sekcii. Nakoľko názov súboru pre každý mail je uložený v SQL tabuľke `emails`, nie je potrebné mailové súbory špecifikovať zvlášť aj v konfiguračnom súbore. V tejto práci sa používajú názvy súborov s koncovku `.list` a samotný názov tvorí päťmiestne poradové číslo súboru (so začiatočnými nulami). Číslovať sa začína od nuly.

4.5 Vlákna

Medzi jednu z najdôležitejších častí tejto práce patrí priraďovanie mailov do jednotlivých vlákien a budovanie prislúchajúcich grafov. Takmer žiaden existujúci software ani služba to nerobí správne a poriadne. Popis najčastejšie používaných služieb je uvedený v tretej kapitole. Popis vlákien a grafov, ktoré maily tvoria, je uvedený v prvej kapitole, sekcii 1.1.1 Jednoduchá správa.

Pripomenieme, že všetky maily, ktoré máme v archíve nám predstavujú vrcholy jedného veľkého grafu. Hrany tohoto grafu pochádzajú z jednotlivých mailových položiek `In-Reply-To` a `References` (ktoré sa odkazujú na ďalšie maily). Slabo súvislé komponenty tohoto grafu považujeme za oddelené vlákna. Teda jedno vlákno nám predstavuje graf, ktorého vrcholy sú maily a hrany predstavujú položky `In-Reply-To` a `References`. V prípade dodržania mailových štandardov bude tento graf reprezentovať tranzitívny uzáver grafu tvoreného iba z `In-Reply-To` položiek. A graf z týchto položiek bude orientovaný a acyklický.

Na rozdiel od iných existujúcich služieb, táto práca tvorí vlákna zo všetkých mailov v archíve a nie len vlákna pre určitý mesiac či rok. Umožní to prezerať súvislo aj mailové konverzácie, ktoré sú na rozhraní mesiacov či rokov.

Naviac okrem údajov z `In-Reply-To` a `References`, vlákna sa tvoria aj z predmetu mailu (toto správanie je možné v programe vypnúť, ak si to správca konferencie neželá). Ak majú dve mailové správy podobný predmet, tak potom budú zaradené do rovnakého vlákna. Predmety sú podobné vtedy, ak po odstránení úvodných bielych znakov a iných reťazcov (napr. `RE` a `FWD`, ktoré zvyknú pridávať rôzne mailové programy) zo začiatku oboch predmetov sa stanú zhodnými.

V ideálnom prípade jedno vlákno tvorí tranzitívny uzáver orientovaného acyklického grafu. Bežne sa ale stane, že nie všetky maily sú v archíve dostupné, alebo, že nemajú špecifikované správne všetky hodnoty v položke `References`. Vtedy

nám bude vlákno tvoriť akýsi neúplný tranzitívny uzáver. V horšom prípade, ak položky chýbajú úplne, sa môže stať, že graf bude nesúvislý. A v najhoršom prípade, ak v hlavičke sú neplatné údaje, sa môže stať úplne čokoľvek a vlákno bude tvoriť ľubovoľný graf, pokojne aj s cyklami.

Za zmienku ešte stojí, že vďaka použitiu heuristiky s podobnými predmetmi, môže nastať, že vlákno bude tvoriť nesúvislý graf. Je to preto, že heuristika s podobnými predmetmi nespája dané maily novými hranami. Iba zlúči vlákna dokopy, z ktorých maily predtým pochádzali.

4.6 Generovanie stromu mailov

Zobraziť orientovaný acyklický graf a k tomu všetky maily, čo ho tvoria nie je úplne najprehľadnejšie. Vo väčšine prípadov ale vlákno bude tvoriť iba strom, keďže skoro všetky mailové programy umožňujú odpovedať iba na jeden mail. Strom s mailami sa už dá zobraziť jednoduchšie (napr. ako rozbaľovací zoznam). Z týchto dôvodov táto práca bude zobrazovať jednotlivé vlákna ako stromy.

V nasledujúcej sekcii je popísaný algoritmus, ktorý pre jedno konkrétne vlákno vybuduje strom. Algoritmus je motivovaný topologickým triedením a to z toho dôvodu, že ak dostaneme kompletný popis hrán nejakého rozumného vlákna, tak topologickým triedením dostaneme práve strom, ktorý vyzerá presne ako dané vlákno. Z menej rozumného popisu hrán niektorého vlákna týmto spôsobom získame aspoň čiastočné zoradenie mailov podľa poradia odpovedí (to nám zabezpečí práve topologické triedenie).

4.6.1 Popis algoritmu na vybudovanie stromu

Algoritmus na vybudovanie stromu mailov z konkrétneho jedného vlákna:

0. Ako vstup dostaneme orientovaný graf g. Vrcholy grafu g budú všetky maily v danom vlákne. Tieto maily vieme získať z SQL tabuľky `emails`, lebo pre každý riadok tabuľky máme poznačenú hodnotu, do ktorého vlákna ktorý mail patrí. Hrany grafu g budú dvoch typov. Prvý typ hrán budú tvoriť údaje z položiek `In-Reply-To` a druhý typ hrán budú tvoriť údaje z položiek `References`. Orientácia hrán bude od mailov na ktoré sa odpovedá. Viacnásobné hrany v grafe nedovolíme. V prípade, že by mali existovať viacnásobné hrany oboch typov, tak hranu typu `References` odstránime.

   Ďalej si spravíme kópiu grafu g a označíme ju ako g'. Graf g sa počas behu algoritmu meniť nebude.

   Výstup bude tvoriť orientovaný graf s (s rovnakou orientáciu hrán, teda od mailov na ktoré sa odpovedá). Po skončení algoritmu to bude zakorenený strom. Budovať ho začneme z prázdneho grafu a nikdy nedovolíme pridať hranu, ktorá by spravila neorientovaný cyklus v grafe. Detekciu budeme robiť pred každým pokusom pridať hranu do grafu a použijeme k tomu dátovú štruktúru Disjoint-set Union-Find. Tá uchováva jednotlivé komponenty grafu ako disjunktné množiny vrcholov. Štruktúra umožňuje spojiť dve množiny a o každom vrchole povedať v ktorej množine sa nachádza. Pred pridaním

každej hrany sa overí, že vrcholy, ktoré tvoria hranu sa nachádzajú v rôznych množinách (a teda v rôznych komponentoch grafu). Po pridaní hrany sa tieto množiny spoja do jednej.

Pri pridávaní novej hrany taktiež nedovolíme pridať hranu, ktorá by viedla do niektorého vrcholu, do ktorého už vedie iná hrana.

1. Z grafu g' si vyberieme vrchol v, z ktorého vedie najmenej hrán. Ak existuje viacero vrcholov s rovnakým počtom výstupných hrán, potom preferujeme taký vrchol v, ktorý reprezentuje existujúci mail v archíve. V prípade, že ich existuje stále viacero, deterministickým spôsobom z nich niektorý vyberieme. Môžeme použiť napr. mail s najmenším `Message-Id` v lexikografickom poradí (pretože máme zaručené, že každý mail ho bude mať jedinečný). Determinizmus potrebujeme kvôli tomu, aby algoritmus nezávisel na poradí vrcholov, alebo hrán, ktoré pochádzajú z databázy. Lebo poradie príchodu mailov do archívu nie je definované a môže byť už úplne ľubovolné.

   V rozumnom prípade sme si vybrali vrchol, z ktorého nevedie žiadna hrana. V prípade, že nie, tak v grafe g je cyklus a vrchol v leží na ňom. To ale nevadí, nakoľko podmienka v bode 0 zabezpečí, že výstupný graf s cyklus obsahovať nebude.

2. Postupne do s pridáme všetky hrany, ktoré vedú z vrcholu v do všetkých susedov v grafe g. Najprv pridáme hrany typu `In-Reply-To`, potom `References`. Poradie jednotlivých hrán bude opäť zvolené deterministicky.

3. Odstránime vrchol v z grafu g'. Pokiaľ existuje v grafe g' nejaký vrchol, pokračujeme v bode 1.

4. Zo všetkých vrcholov v grafe s, do ktorých nevedie žiadna hrana vyberieme ten, ktorý reprezentuje najstarší mail. Tento vrchol označíme ako *koreň*. Najstarší mail je ten, ktorý má najmenší nenulový dátum. V prípade, že taký mail neexistuje, povolíme aj mail s nulovým dátumom. Opäť, ak takýchto mailov existuje viac, vyberieme z nich jeden deterministickým spôsobom.

5. V grafe s vytvoríme nové hrany z *koreňa* do všetkých ostatných vrcholov, do ktorých zatiaľ neviedla žiadna hrana.

6. Z grafu s odstránime každý vrchol, pre ktorý platia všetky nasledujúce podmienky:
   - vedie do neho hrana z koreňa
   - nevedie z neho žiadna hrana
   - reprezentuje implicitný mail

4.6.2 Rozbor algoritmu na vybudovanie stromu

- Algoritmus sa vždy zastaví.

  Nakoľko je graf g' konečný a v každom kroku iterácie algoritmu odstránime z neho jeden vrchol, tak po konečnom počte iterácií sa algoritmus musí zastaviť.

- Výstupom algoritmu bude vždy zakorenený strom. A ten bude obsahovať práve vrcholy, ktoré budú reprezentovať maily v danom vlákne.

  Vďaka bodu 0 sa vo výstupnom grafe s nebude nachádzať neorientovaný cyklus a do každého vrcholu bude viesť maximálne jedna hrana. Vďaka bodu 5 nebude existovať v grafe vrchol, do ktorého by neviedla hrana až na koreň. Keďže je graf konečný a do každého vrcholu (až na koreň) vedie práve jedna hrana, tak graf bude súvislý. Súvislý graf bez (neorientovanej) kružnice už tvorí strom a zvolený vrchol ako *koreň* v bode 4 bude tvoriť naozaj koreň stromu podľa udanej orientácie hrán v grafe.

- V bode 2 sa musí naozaj pracovať s grafom g a nie s grafom g'. Ináč by v grafe s pred bodom 5 pre väčšinu vstupov nebola žiadna hrana.

  Keďže algoritmus odoberá z grafu g' vrcholy topologicky zospodu, tak synovia každého vrcholu sú spracovaní skôr ako samotné vrcholy. To znamená, že ak by sme v bode 2 pracovali s grafom g', tak do grafu s by sme nepridali žiadnu hranu. Lebo vrchol v by už nemohol mať žiadnych susedov, do ktorých by viedla hrana – tí by už boli z g' odstránení v predchádzajúcich iteráciách algoritmu.

  Toto pozorovanie je aj zároveň dôvodom, prečo je potrebné mať kópiu pôvodného vstupného grafu.

- Ak vlákno obsahuje iba maily, ktoré majú práve jednu hodnotu v položke `In-Reply-To` až na koreň a celé to tvorí strom, tak algoritmus vydá na výstup rovnaký strom.

  Vďaka bodu 1 budeme strom s budovať od listov a vďaka bodu 2 budeme pridávať iba hrany typu `In-Reply-To`. Až sa dostaneme ku koreňu a výstupom bude požadovaný strom.

- Ak pôvodné vlákno mailov obsahuje lineárne usporiadanú vetvu, na ktorej je jeden implicitný mail, potom algoritmus dá na výstup strom, kde bude táto vetva opäť reprezentovaná lineárnym usporiadaním.

  Keďže sa implicitný mail nenachádza v archíve, tak bude pre neho chýbať aj informácia o `In-Reply-to` hrane. Vďaka bodu 1 sa skôr spracuje otec implicitného mailu ako mail samotný a vďaka bodu 2 sa pridá hrana typu `References` z otca na syna do výstupného grafu. Keďže sa informácia o `In-Reply-To` hrane z otca implicitného mailu na samotný mail v archíve nenachádza, tak toto riešenie je dostatočne rozumné – netrhá vlákna. Pri spracovávaní implicitného mailu sa nepridá už žiadna hrana, keďže do syna implicitného mailu hrana vedie. V bode 5 sa pridá hrana z koreňa do vrcholu implicitného mailu a v bode 6 sa tento vrchol odstráni. Teda naozaj daná časť vetvy bude na výstupe tvoriť lineárne usporiadanie (síce bez chýbajúceho implicitného mailu).

  V takomto jednoduchom prípade, kedy chýba vo vlákne práve jeden mail a vie sa, že chýba niekde na konkrétnej vetve, tak by bolo možné relatívne presne určiť kde chýba. A potom túto informáciu zobraziť. Avšak oveľa častejšie sa stáva, že vlákna tvoria veľmi košatý strom a mailov chýba o dosť viac. V tomto prípade je pomerne ťažké vymyslieť heuristiku, ako presne

určiť, kde mail chýba. A z tohoto dôvodu je algoritmus vymyslený tak, aby namiesto hádania a používania rôznych heuristík využil iba informácie, ktoré sú dostupné a odstránil z výsledného stromu nepotrebné vrcholy.

Za povšimnutie ešte stojí aj nasledovný fakt. Ak vlákno tvorí niekoľko mailov, z nich práve jeden je implicitný a zároveň ten jeden je vybratý za koreň (napr. pretože všetky ostatné maily majú s nim `In-Reply-To` hranu), tak tento mail naozaj ostane koreňom a z výsledného grafu s sa neodstráni. Teda, vo výsledkom grafe sa môžu nachádzať aj vrcholy, ktoré reprezentujú implicitné maily. Je to vtedy, ak prinášajú nejakú podstatnú informáciu v danom vlákne.

5. Vývojová dokumentácia

Na implementáciu programu pre túto prácu som zvolil programovací jazyk Perl. Dôvody pre moju voľbu tohoto jazyka sú: Vynikajúca podpora pre regulárne výrazy, práca s nimi je priamo súčasťou jazyka. Interpreter jazyka je pomerne rýchly a na systémové zdroje až tak nenáročný. Perl sa už dlho používa na webové ako aj terminálové aplikácie. Je dostupný pre radu rôznych operačných systémov a jednoducho sa v ňom píšu multiplatformové aplikácie. V neposlednom rade medzi dôvody patrí aj široká dostupnosť rôznych prídavných modulov, ktoré implementujú rôzne funkcie na prácu s rôznymi údajmi.

5.1 Súčasti implementácie

Implementácia je rozdelená do niekoľkých perlových modulov. Tie sú navrhnuté objektovým spôsobom a obsahujú funkcie potrebné k vytvoreniu spustiteľných aplikácií. Ďalej implementácia obsahuje terminálovú a webovú aplikáciu, ktoré sú uložené v samostatných súboroch.

Zoznam jednotlivých modulov a súborov:

- Modul `PList::Email` je abstraktné rozhranie na prácu s mailami. Odvodzujú sa z neho moduly, ktoré reprezentujú maily.
- Modul `PList::Email::MIME` implementuje spracovávanie mailov vo formáte MIME. Dokáže maily iba čítať.
- Modul `PList::Email::Binary` implementuje načítavanie a ukladanie mailov v internom formáte použitom v tejto práci.
- Modul `PList::Email::View` slúži na vytvorenie HTML dokumentu z objektu, ktorý je odvodený od `PList::Email` (teda `PList::Email::MIME` a `PList::Email::Binary`).
- Modul `PList::List` je abstraktné rozhranie na prácu so zoznamom mailov. Odvodzujú sa z neho moduly, ktoré reprezentujú zoznamy mailov.
- Modul `PList::List::MBox` implementuje spracovávanie mailových archívov v rôznych variantách typu *MBox*. Archívy dokáže iba čítať.
- Modul `PList::List::Binary` implementuje prácu so zoznamom mailov, ktoré sú v internom formáte použitom v tejto práci. Dokáže maily postupne čítať ako aj nové pripisovať na koniec súboru.
- Modul `PList::Index` slúži na prácu s adresárom *index*. Dokáže takýto adresár vytvoriť spolu so správnym nastavením databázy. Ďalej vie pridať nové maily do zoznamu, odstrániť ich, ako aj pridať dáta do databázy. Taktiež slúži na rôzne dopyty na databázu a implementuje aj algoritmy, ktoré slúžia na prácu s vláknami.

- Súbor `plist.pl` je spustiteľný skript, ktorý tvorí terminálovú aplikáciu. Umožňuje využívať funkcie vyššie uvedených modulov a slúži na kompletnú správu mailových archívov.

- Súbor `plist.cgi` je CGI skript, ktorý implementuje webovú aplikáciu. Typicky tento skript spúšťa webový server a slúži na zobrazenie mailov vo webovom prehliadači. Taktiež využíva vyššie uvedené moduly na čítanie mailov a vlákien z archívov.

5.2 Problémy pri implementácií

Pri implementácií samotného návrhu som sa stretol s niekoľkými problémami. Medzi ne patria aj samotné problémy vyplývajúce zo spracovávania mailov (ako je popísané v kapitole 2).

5.2.1 Dátum

Dátum sa zisťuje postupne z položiek `Date`, `Received` a `Resent-Date` a použije sa prvý, ktorý nie je menší ako 5 dní od dátumu uvedeného v úvodnom riadku z *MBox* archívu. V prípade, že dátum nie je dostupný v žiadnej položke, použije sa ten z úvodného riadku. A v prípade, že dátum nie je dostupný v úvodnom riadku, kontrola 5 dní sa nevykonáva. Na spracovanie dátumu, ktorý sa vyskytuje v rôznych formátoch som použil perlový modul `Date::Parse`. Ten zvláda rozpoznať rôzne formáty, ktoré sa bežne používajú.

5.2.2 Identifikátory

Problémami s identifikátormi a ich riešením sa už zaoberá druhá kapitola. Okrem riešených problémov môže nastať ešte ďalší a to, že platný identifikátor `Message-Id` sa v správe vôbec nenachádza. Keďže archivačný program potrebuje pre každú správu unikátny identifikátor, tak pre tento prípad som ako identifikátor zvolil reťazec *sum*`@nohost`, kde *sum* je SHA1 hash celého mailu v šestnástkovej sústave. Táto hashovacia funkcia je relatívne rýchla a odolná voči kolíziám, čo poslúži na vytvorenie jednoznačného identifikátoru mailu celkom dobre.

5.2.3 Mailové adresy

Na spracovanie mailových adries a im prislúchajúcich mien z položiek `From`, `To` a `Cc` som použil perlový modul `Email::Address`. Výkon tohoto modulu silno závisí od hĺbky vnorených komentárov v mailových adresách. Z tohoto dôvodu som sa rozhodol vnorené komentáre ignorovať a nastaviť modul tak, aby vnorené komentáre sa nesnažil rozpoznať. Má to za cieľ oveľa väčšiu rýchlosť rozpoznávania mena a adresy z reťazca (hoci nie úplne správne). So zapnutými vnorenými komentármi rozpoznanie adries niekedy trvalo aj viac ako 10 minút. Napr. pre reťazec, ktorý obsahoval 10 adries a veľa medzier. S vypnutím vnorených komentárov to bolo pod jednu sekundu. Program na spracovávanie a archivovanie mailov má byť rýchly a takéto spomalenie nie je pri veľkom množstve mailov prípustné.

Keďže dokumentácia pre modul `Email::Address` neuvádza, ako sa má správne vypnúť rozpoznávanie vnorených komentárov, tak tu uvádzam príklad. Modul nie je možné načítať kľúčovým slovom `use`, nakoľko je nutné hodnotu premennej `COMMENT_NEST_LEVEL` zmeniť na 1 ešte pred samotným načítaním modulu. O tomto sa v dokumentácií nepíše, ale z nahliadnutia do zdrojových kódov modulu je to jasné.

Načítať modul s vypnutím vnorených komentárov sa dá napr. takto:

```
BEGIN {
        local $Email::Address::COMMENT_NEST_LEVEL = 1;
        require Email::Address;
        import Email::Address;
}
```

5.2.4 MIME mail

Jeden z najväčších problémov je rozumne spracovať obsah MIME mailu. Na prácu s mailom som zvolil modul `Email::MIME`. Tento modul poskytuje vysokoúrovňové funkcie, ktoré automaticky dekódujú MIME, ako aj použité kódovanie. Ďalej poskytuje aj nízkoúrovňové funkcie, ktoré vrátia konkrétnu hodnotu niektorej položky v hlavičke a voliteľne dekódujú aj MIME.

Vysokoúrovňové funkcie predpokladajú, že mail vyhovuje presne MIME štandardom a pre nekorektné maily teda nemusia vrátiť ani rozumné hodnoty. Častokrát sa stane, že vrátia prázdne dáta. Z tohoto dôvodu som sa rozhodol nie všade používať vysokoúrovňové funkcie, ale niekde postupne volať nízkoúrovňové a snažiť sa čo najlepšie interpretovať dáta z mailov, ktoré nevyhovujú štandardom. Má to za následok, že dostanem viac informácií aj o nekorektných mailoch.

Na spracovanie položky `Content-Type`, v ktorej sa nachádza informácia o znakovej sade a MIME type, som zvolil perlový modul `Email::MIME::ContentType`. Ten pri nastavení premennej `STRICT_PARAMS` sa dokáže vysporiadať aj s hodnotami, ktoré nie sú úplne podľa štandardu.

Na autodetekciu znakovej sady som použil modul `Encode::Detect`. Ten sa využíva v prípade, že informácia o znakovej sade v `Content-Type` chýba.

Na autodetekciu MIME type, teda typu obsahu príloh, som zvolil modul `File::MimeInfo::Magic`, ktorý používa databázu `shared-mime-info`. Táto databáza zvykne byť súčasťou systému a nachádzajú sa v nej informácie a definície pre rôzne typy súborových formátov. Definície sú vyjadrené zvyčajne pomocou prvých pár bajtov súboru, niekedy označovaných ako „magických“ znakov. Tento modul sa používa v prípade, že MIME type nie je v maili špecifikovaný, alebo keď je nastavený na niektorú generickú hodnotu, ktorá nehovorí nič o obsahu prílohy.

Pre bezpečnosť pri zobrazovaní mailov sa štandardne zobrazuje alternatíva čistého textu. Niektoré maily obsahujú iba HTML obsah a pre zachovanie bezpečnosti je nutné spraviť konverziu do čistého textu. Na tento problém som použil perlový modul `HTML::Strip`, ktorý odstraňuje HTML značky z textu. Tento modul má ale menší problém so spracovaním textu v UTF-8 a keďže všetky textové časti si program ukladá výhradne v UTF-8, je potrebné tento problém vyriešiť. Podľa dostupných informácií je nutné pred použitím funkcie `decode_entities()`

z modulu `HTML::Entities` (ktorá dekóduje špeciálne HTML znaky) dekódovať samotný UTF-8 text [13].

Samotné dekódovanie MIME mailu je dosť zdĺhavý proces, kvôli nutnosti použitia rôznych autodetekcií, ako aj nutnosti vedieť pracovať s rôznymi kódovaniami, ako je *quoted-printable* alebo *Base64*. Niekedy je dobré tento proces dekódovania mailu oddialiť a spustiť ho až keď to bude nutné. Niektoré informácie o maili je možné veľmi rýchlo získať aj bez toho (napr. `Message-Id` pri spracovávaní *MBox* archívu). Tento oneskorený proces je implementovaný v perli pomocou funkcie `AUTOLOAD`. Pri vytváraní objektu mailu sa takmer nič nespraví a až pri prvom zavolaní niektorej funkcie (implementované pomocou `AUTOLOAD`), ktorá vyžaduje dáta z mailu sa použije modul `Email::MIME` na dekódovanie mailovej správy. Tento spôsob práce zrýchlil preskakovanie duplicitných mailov, ktoré už boli spracované asi o polovicu času.

5.2.5 *MBox* archívy

Problémy s *MBox* archívmi a ich riešenia sú popísané v druhej kapitole. Okrem týchto problémov som sa pri implementácii stretol aj s inými a to dosť závažnými. Pre Perl existujú 3 známe moduly, ktoré dokážu čítať niektoré varianty *MBox* archívov. Každý z nich má vážne problémy a ani jeden z nich nie je možné použiť na čítanie neznámeho *MBox* archívu. Žiaľ, pre účely tejto práce nie je možné bez modifikácie použiť ani jeden z nich.

Popis jednotlivých *MBox* modulov:

- Prvý modul je `Mail::Mbox::MessageParser`. Ten by mal zvládať spracovať *mboxcl2* aj *mboxrd* varianty. Na niektorých súboroch sa ale zacyklí a niektoré iné súbory číta príliš pomaly. Špeciálne sa to prejavuje pri súboroch, ktoré mixujú konce riadkov `CRLF` a `LF`. Z dôvodu možnosti zacyklenia, ako aj pomalého čítania, je modul na všeobecné použitie veľmi nepoužiteľný. Správca modulu bol o tomto probléme informovaný, ale kvôli tomu ako je modul navrhnutý nie je ho možné rozumne opraviť.

- Druhý modul je `Mail::Box::Mbox`. Ten pri testovaní na archívoch, ktoré mixujú *mboxcl2* a *mboxrd* varianty dokopy v jednom súbore obstál zle. Takéto súbory nevedel správne prečítať a ani rozpoznať, kde je koniec mailu. Naviac modul má problém pri čítaní veľkého súboru (nad 500 MB). Pri niektorých *MBox* archívoch dokázal využiť viac ako 5 GB operačnej pamäte a úplne spomaliť až zastaviť bežiaci server. Z tohoto dôvodu je, ako univerzálny modul, tiež nepoužiteľný. Väčšie archívy majú veľkosť aj 1 GB.

- Tretí modul je `Email::Folder::Mbox`. Ten ako jediný sa na testovacích archívoch nezacyklil, výrazne nespomalil, nevyužil všetku operačnú pamäť systému a až na pár problémov dokázal dosť dobre rozpoznať a prečítať *mboxcl2* a *mboxrd* varianty. Na rozdiel od predchádzajúcich dvoch modulov, tento nedokáže vrátiť hodnotu úvodného riadku, ktorý obsahuje napr. dôležitý údaj a to dátum. Pri čítaní *mboxcl* variánt sa objavil problém, ak položka `Content-Length`, alebo `Lines` obsahuje pred udávajúcim číslom nejaké biele znaky (napr. medzery). V takomto prípade modul nedokázal položky správne interpretovať.

Okrem toho ani jeden modul nedokázal pri *mboxrd* a *mboxcl* variante odstrániť úvodné znaky `>` z tela správ ako je popísané v druhej kapitole.

Nakoľko tretí modul sa mi zdal asi najjednoduchší a čitateľne napísaný, rozhodol som sa problém s *mboxcl* variantou opraviť a pridať do neho potrebné vyššie uvedené chýbajúce funkcie, tak aby bol pre účely tejto práce použiteľný.

Opravu problému s *mboxcl* variantou ako aj zmeny, ktoré pridávajú novú funkcionalitu som zaslal autorom a správcom projektu `Email-Folder`, ktorého je modul `Email::Folder::Mbox` súčasťou. Avšak do dňa 23. 07. 2014 sa mi k zmenám nevyjadrili a ani nevydali novú verziu, ktorá by problém s *mboxcl* riešila. Posledná verzia k tomu dňu je 0.858. Z tohto dôvodu je súčasťou práce aj modifikovaná verzia modulu `Email::Folder::Mbox`.

5.3 Profilovanie implementácie

Pre Perl existuje veľmi rýchly a výkonný profiler *NYTProf*, v podobe modulu `Devel::NYTProf`. Pomocou neho je dosť jednoduché určiť, ktoré časti programu sú pomalé a ktoré sú rýchle. Zvláda podať informácie až na úroveň jednotlivých riadkov zdrojového kódu. Modul je potrebné načítať ako prvý a potom počas celého behu programu zbiera dáta. Zaznamenáva si ako dlho trvá vykonávanie každej funkcie aj ako dlho trvá vykonávanie kódu na každom riadku. Práve tento nástroj mi umožnil dostať o dosť väčší výkon z programu a zrýchliť spracovávanie obrovského množstva mailov viac ako 10-násobne.

6. Užívateľská dokumentácia

Program sa skladá z dvoch častí. Prvá časť je webová aplikácia, ktorá slúži na prezeranie mailov a druhá časť je terminálová aplikácia. Tá slúži pre administráciu archívu. Najprv popíšeme inštaláciu programu, potom jeho konfiguráciu a nakoniec ovládanie webovej a terminálovej časti spolu s príkladmi pre unixový systém.

6.1 Inštalácia

K behu webovej aj terminálovej aplikácie je potrebné mať v systéme nainštalovaný interpreter jazyka Perl a ďalšie externé perlové moduly, ktoré aplikácia využíva. Konkrétne sú potrebné tieto moduly:[1]

`CGI`, `Cwd`, `Date::Format`, `Date::Parse`, `DBD::mysql`, `DBD::SQLite`, `DBI`, `Digest::SHA`, `Email::Address`, `Email::MIME`, `Email::MIME::ContentType`, `Encode::Detect`, `Encode`, `File::MimeInfo::Magic`, `File::Path`, `FindBin`, `HTML::Entities`, `HTML::FromText`, `HTML::Strip`, `HTML::Template`, `Number::Bytes::Human`, `Time::Piece`

Inštalácia interpreteru a modulov závisí na použitom systéme. Správcovia systémov nech postupujú podľa užívateľskej dokumentácie toho konkrétneho systému. Jednotlivé moduly je možné stiahnuť zo stránok `http://www.perl.org/`, `http://www.cpan.org/` alebo aj z `http://metacpan.org/`. Na serveroch so systémom Debian by malo stačiť nainštalovať nasledovné balíčky:[2]

`libdbd-mysql-perl`, `libdbd-sqlite3-perl`, `libdbi-perl`, `libemail-mime-perl`, `libencode-detect-perl`, `libfile-mimeinfo-perl`, `libhtml-fromtext-perl`, `libhtml-parser-perl`, `libhtml-strip-perl`, `libhtml-template-perl`, `libnumber-bytes-human-perl`, `libtimedate-perl`, `perl`

Okrem externých modulov, aplikácia používa ešte aj interné. Tie sú dodávané spoločne s aplikáciou. Pre správnu funkčnosť aplikácie, sa musia interné moduly nachádzať buď v rovnakom adresári kde je aplikácia alebo v niektorom adresári, v ktorom perlový interpreter moduly hľadá. Konkrétne adresáre, v ktorých sa perlové moduly nachádzajú sa líšia od použitého operačného systému ako aj verzie interpreteru. Správcovia systémov opäť nech postupujú podľa dokumentácie systému a nainštalovaného interpreteru perl.

Medzi modulmi dodávanými s aplikáciou je aj pozmenený perlový modul `Email::Folder::Mbox`. Dôvody sú uvedené v piatej kapitole.

Terminálová aplikácia je nastavená tak, aby preferovala použitie perlového modulu `Email::Folder::Mbox` z adresára, kde sú uložené interné moduly. Správcovia systémov by sa mali uistiť, že perlový interpreter používa naozaj modifikovanú verziu, ktorá je dodávaná s terminálovou aplikáciu. Pôvodnú nemodifikovanú verziu nie je možné použiť a terminálová aplikácia vypíše pri spúšťaní chybu.

[1] Aplikácia funguje aj bez modulu `Number::Bytes::Human` avšak veľkosti príloh budú udávané vždy iba v bajtoch

[2] Balík `libnumber-bytes-human-perl` nie je nutný, ale obsahuje voliteľný perlový modul `Number::Bytes::Human`, ktorý sa používa na udávanie veľkosti príloh v čitateľnejších jednotkách

6.2 Konfigurácia

Terminálovú aplikáciu nie je potrebné nijak konfigurovať. Tvorí ju spustiteľný perlový skript s názvom `plist.pl`. Je iba potrebné aby bola umiestnená v rovnakom adresári, kde sú aj dodané interné moduly.

Webová aplikácia úzko súvisí s terminálovou. Využíva rovnaké moduly a prostriedky na prístup do archívu ako terminálová. Webová aplikácia je CGI skript, pod názvom `plist.cgi` a pre správnu funkčnosť musí byť v rovnakom adresári ako terminálová aplikácia. Pre spustenie webovej aplikácie je potrebné mať správne nainštalovaný a nakonfigurovaný webový server HTTP, ktorý má podporu pre CGI skripty. Nakoľko konfigurácia webového servera a spustenia CGI skriptov sa líši skoro v každom operačnom systéme, administrátori nech postupujú podľa dokumentácie k použitému webovému serveru a operačnému systému.

Webová aplikácia bola testovaná so serverom Apache vo verzii 2.2. Pre lepšiu čitateľnosť webových adries so serverom apache2, je možné použiť pripravený konfiguračný súbor `.htaccess`, ktorý je súčasťou programu.

6.3 Webová aplikácia

Webová aplikácia slúži užívateľom či členom mailových konferencií na prezeranie mailových archívov. Pomocou nej sa dajú čítať maily, zobraziť jednotlivé vlákna konverzácií (stromy mailov) ako aj stiahnuť prílohy mailov. Ovládanie webovej aplikácie je priamočiare a malo by byť jednoduché. Základná stránka obsahuje zoznam archívov, z ktorej je možné ľubovolný z nich otvoriť. Archívy, ktoré začínajú bodkou sú skryté a v zozname sa nezobrazujú.

Stránka konkrétneho archívu zobrazuje posledných 10 mailov a odkazy, pomocou ktorých sa dá prehliadať archív mailov, či vyhľadávať konkrétne maily. V prípade, že niektorý mail je nekompletný, chýba, alebo ho nie je možné zobraziť, tak webová aplikácia zobrazí v stĺpci `Subject` iba text `unknown`. Takýto mail nie je ho možné otvoriť. Slúži to napríklad na zobrazenie chýbajúceho mailu, na ktorý existujú odpovede.

Archive netdev

Latest emails:

Subject	From	Date
Re: [PATCH net-next] net/mlx4_core: Handle null return by pci_match_...	Or Gerlitz <or.gerlitz@gmail....	2014-03-30 20:08:06
[tip:x86/vdso] x86, vdso: Fix the symbol versions on the 32-bit vDSO	tip-bot for Andy Lutomirski ...	2014-03-30 19:28:27
Re: [PATCH v2 10/29] ktap: add string handling code(kernel/trace/ktap...	Andi Kleen <andi@firstfloor....	2014-03-30 19:19:57
Re: [PATCH v2 08/29] ktap: add bytecode reader(kernel/trace/ktap/kp...	Andi Kleen <andi@firstfloor....	2014-03-30 19:17:03
Re: [PATCH] x86: Fix the symbol versions on the 32-bit vDSO	H. Peter Anvin <hpa@zytor....	2014-03-30 18:59:17
Re: [Regression, bisected 9e30cc] "sysfs, kernfs: no need to kern_mo...	Alexandre Demers <alexand...	2014-03-30 18:53:56
[PATCH 6/6] x86: Remove vestiges of VDSO_PRELINK and some outd...	Andy Lutomirski <luto@ama...	2014-03-30 18:53:32
[PATCH 5/6] x86: Move the vvar and hpet mappings next to the 64-bit...	Andy Lutomirski <luto@ama...	2014-03-30 18:53:31
[PATCH 4/6] x86: Move the 32-bit vdso special pages after the text	Andy Lutomirski <luto@ama...	2014-03-30 18:53:30
[PATCH 3/6] x86: Reimplement vdso.so preparation in build-time C	Andy Lutomirski <luto@ama...	2014-03-30 18:53:29

Actions:

Search emails
Show all trees
Show all emails
Show all roots of emails

Quick Search

Show list of archives

Zoznam mailov sa dá prepínať medzi tromi možnými zobrazeniami. V každom sú maily zoradené podľa dátumu a to buď zostupne alebo vzostupe. V stromovom zobrazení sa dá naviac nastaviť aj zoradenie koreňov jednotlivých stromov.

Obyčajné zobrazenie – lineárne usporiada maily podľa dátumu:

Archive netdev - Emails (1 - 100)

View: Trees Emails Roots

Subject	From	Date (DESC)
Re: [PATCH v4 0/6] Krait L1/L2 EDAC driver	Borislav Petkov <bp@alien8....	2014-01-04 11:19:01
Re: [PATCH 1/2] Staging: lustre: Fix line length exceeding 80 characters	Joe Perches <joe@perches....	2014-01-05 21:30:51
[PATCH v5 0/7] Getting rid of get_unused_fd() / enable close-on-exec	Yann Droneaud <ydroneaud...	2014-01-05 21:36:27
[PATCHv5 1/7] ia64: use get_unused_fd_flags(0) instead of get_unuse...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:28
[PATCHv5 2/7] ppc/cell: use get_unused_fd_flags(0) instead of get_un...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:29
[PATCHv5 3/7] binfmt_misc: use get_unused_fd_flags(0) instead of ge...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:30
[PATCHv5 4/7] file: use get_unused_fd_flags(0) instead of get_unused...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:31
[PATCHv5 5/7] fanotify: enable close-on-exec on events' fd when requ...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:32
[PATCHv5 6/7] perf: introduce a flag to enable close-on-exec in perf_e...	Yann Droneaud <ydroneaud...	2014-01-05 21:36:33
[PATCHv5 7/7] file: remove macro get_unused_fd()	Yann Droneaud <ydroneaud...	2014-01-05 21:36:34
Re: [PATCH v3 1/6] locks: consolidate common code in the flock_to_p...	J. Bruce Fields <bfields@fiel...	2014-01-05 21:39:34
[PATCH] locks: fix posix lock range overflow handling	J. Bruce Fields <bfields@fiel...	2014-01-05 21:42:21
Re: [PATCH V2 0/4] misc: xgene: Add support for APM X-Gene SoC Q...	Ravi Patel <rapatel@apm.co...	2014-01-05 21:48:36
Re: [PATCH V2 0/4] misc: xgene: Add support for APM X-Gene SoC Q...	Ravi Patel <rapatel@apm.co...	2014-01-05 21:52:05
[PATCH] Staging: rtl8187se: fix styling issues in r8180_wx.c	Dan LaManna <dan.lamanna...	2014-01-05 22:17:36
Re: [PATCH] Staging: tidspbridge: Use hashtable implementation	Dan Carpenter <dan.carpen...	2014-01-05 22:18:04
Re: How does a newbie find work?	One Thousand Gnomes <gn...	2014-01-05 22:29:51
Re: [Patch v2 00/13] Introduce ACPI _DSM helper functions to simplify...	Rafael J. Wysocki <rjw@rjwy...	2014-01-05 22:45:11
Re: [PATCH] Staging: rtl8187se: fix styling issues in r8180_wx.c	Dan Carpenter <dan.carpen...	2014-01-05 22:45:55
Re: [PATCH 25/38] pnp: card: add missing put_device call	Rafael J. Wysocki <rjw@rjwy...	2014-01-05 22:50:49
Re: [PATCH] Staging: rtl8187se: fix styling issues in r8180_wx.c	Dan Carpenter <dan.carpen...	2014-01-05 22:56:03
Re: [PATCH] kobject: provide kobject_put_wait to fix module unload race	Mikulas Patocka <mpatocka...	2014-01-05 23:04:31
Re: [dm-devel] [PATCH] kobject: provide kobject_put_wait to fix modul...	Mikulas Patocka <mpatocka...	2014-01-05 23:04:58
Re: [PATCH] kobject: provide kobject_put_wait to fix module unload race	Mikulas Patocka <mpatocka...	2014-01-05 23:11:23
Re: mdadm raid1 regression	NeilBrown <neilb@suse.de>	2014-01-05 23:11:24
Re: [patch core/stackprotector] stackprotector: Fix build when compil...	Sam Ravnborg <sam@ravn...	2014-01-05 23:13:04
Re: [PATCH] xen-netback: Include header for vmalloc	Wei Liu <wei.liu2@citrix.com>	2014-01-05 23:22:21
Re: [PATCH] kobject: provide kobject_put_wait to fix module unload race	Greg Kroah-Hartman <greg...	2014-01-05 23:23:43

Stromové zobrazenie – zoskupuje maily do stromovej štruktúry:

Archive netdev - Browse trees (1 - 8)

View: Trees Emails Roots

Subject	From	Date (thr DESC) (msg DESC)
• unknown		
• Re: [PATCH v4 0/6] Krait L1/L2 EDAC driver	Borislav Petkov <bp@alien8....	2014-01-04 11:19:01
• Re: [PATCH v4 0/6] Krait L1/L2 EDAC driver	Stephen Boyd <sboyd@cod...	2014-01-06 23:09:48
• Re: [PATCH v4 4/6] devicetree: bindings: Document Krait L1/L2 ED...	Lorenzo Pieralisi <lorenzo.pi...	2014-01-07 11:54:39
• Re: [PATCH v4 4/6] devicetree: bindings: Document Krait L1/L2 ...	Stephen Boyd <sboyd@cod...	2014-01-07 21:12:39
• Re: [PATCH v4 4/6] devicetree: bindings: Document Krait L1/L...	Lorenzo Pieralisi <lorenzo.pi...	2014-01-08 11:05:38
• Re: [PATCH v4 4/6] devicetree: bindings: Document Krait L1/...	Stephen Boyd <sboyd@cod...	2014-01-09 21:52:21
• Re: [PATCH v4 4/6] devicetree: bindings: Document Krait L...	Lorenzo Pieralisi <lorenzo.pi...	2014-01-10 11:54:51
• unknown		
• Re: [PATCH v4 1/6] edac: Don't try to cancel workqueue when it'...	Borislav Petkov <bp@alien8....	2014-01-07 18:19:55
• Re: [PATCH v4 2/6] genirq: export percpu irq functions for module...	Borislav Petkov <bp@alien8....	2014-01-08 00:02:41
• Re: [PATCH v4 3/6] ARM: Add Krait L2 accessor functions	Borislav Petkov <bp@alien8....	2014-01-08 00:07:25
• Re: [PATCH v4 3/6] ARM: Add Krait L2 accessor functions	Stephen Boyd <sboyd@cod...	2014-01-08 00:09:59
• Re: [PATCH v4 5/6] edac: Add support for Krait CPU cache error d...	Borislav Petkov <bp@alien8....	2014-01-08 00:43:27
• Re: [PATCH v4 3/6] ARM: Add Krait L2 accessor functions	Courtney Cavin <courtney.c...	2014-01-09 01:51:59
• Re: [PATCH v4 3/6] ARM: Add Krait L2 accessor functions	Stephen Boyd <sboyd@cod...	2014-01-09 02:54:28
• Re: [PATCH v4 3/6] ARM: Add Krait L2 accessor functions	Borislav Petkov <bp@alien8....	2014-01-09 12:03:59
• unknown		
• Re: [PATCH 1/2] Staging: lustre: Fix line length exceeding 80 chara...	Joe Perches <joe@perches....	2014-01-05 21:30:51
• Re: [PATCH 1/2] Staging: lustre: Fix line length exceeding 80 cha...	Greg KH <gregkh@linuxfou...	2014-01-09 00:54:52
• [PATCH v3 1/2] Staging: lustre: Refactor the function interval_eras...	Monam Agarwal <monamag...	2014-01-11 12:00:33
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the function interva...	Dan Carpenter <dan.carpen...	2014-01-11 12:19:31
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the function inter...	Monam Agarwal <monamag...	2014-01-11 12:26:44
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the function int...	Dan Carpenter <dan.carpen...	2014-01-11 12:39:59
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the function i...	Monam Agarwal <monamag...	2014-01-11 12:44:35
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the functio...	Greg KH <gregkh@linuxfou...	2014-01-11 21:33:58
• Re: [PATCH v3 1/2] Staging: lustre: Refactor the functi...	Xiong, Jinshan <jinshan.xion...	2014-01-14 09:01:44

Zobrazenie koreňov – zobrazí iba korene stromov (zo stromového zobrazenia):

Archive netdev - Roots of trees (1 - 100)

View: Trees Emails Roots

Subject	Date (DESC)
[PATCH v4 0/6] Krait L1/L2 EDAC driver	2014-01-04 11:19:01
[PATCH 1/2] Staging: lustre: Fix line length exceeding 80 characters	2014-01-05 21:30:51
[PATCH v5 0/7] Getting rid of get_unused_fd() / enable close-on-exec	2014-01-05 21:36:27
[PATCH v3 1/6] locks: consolidate common code in the flock_to_posix_lock routines	2014-01-05 21:39:34
[PATCH V2 0/4] misc: xgene: Add support for APM X-Gene SoC Queue Manager/Traffic Manager	2014-01-05 21:48:36
[PATCH] Staging: rtl8187se: fix styling issues in r8180_wx.c	2014-01-05 22:17:36
[PATCH] Staging: tidspbridge: Use hashtable implementation	2014-01-05 22:18:04
How does a newbie find work?	2014-01-05 22:29:51
[Patch v2 00/13] Introduce ACPI _DSM helper functions to simplify code	2014-01-05 22:45:11
[PATCH 25/38] pnp: card: add missing put_device call	2014-01-05 22:50:49
[PATCH] kobject: provide kobject_put_wait to fix module unload race	2014-01-05 23:04:31
mdadm raid1 regression	2014-01-05 23:11:24
[patch core/stackprotector] stackprotector: Fix build when compiler lacks support	2014-01-05 23:13:04
[PATCH] xen-netback: Include header for vmalloc	2014-01-05 23:22:21
n900 led compiler (was Re: [PATCH] Add LED pattern trigger)	2014-01-05 23:23:46
[PATCH] lib/vsprintf: add %pT[C012] format specifier	2014-01-05 23:27:52
Propose for LINUX kernel and PERL	2014-01-05 23:32:08
[PATCH v4] Bluetooth: Add hci_h4p driver	2014-01-05 23:32:50
[PATCH 01/11] drivers: acpi: Mark the function acpi_table_checksum() as static in osl.c	2014-01-06 00:16:33
linux-next: manual merge of the arm-soc tree with the arm tree	2014-01-06 00:20:31
[PATCH] drivers: pnp: Mark the function pnp_build_option() as static in resource.c	2014-01-06 00:26:41
[PATCH 1/1] acpi_pad: call schedule() when need_resched is true	2014-01-06 00:29:07
linux-next: manual merge of the imx-mxs tree with the arm-soc tree	2014-01-06 00:38:50
spidev: fix hang when transfer_one_message fails	2014-01-06 00:39:26
linux-next: manual merge of the imx-mxs tree with the tree	2014-01-06 00:49:38
[PATCH v3 03/14] mm, hugetlb: protect region tracking via newly introduced resv_map lock	2014-01-06 01:12:37
[RFC] lib: raid: New RAID library supporting up to six parities	2014-01-06 01:15:29
possible regression on 3.13 when calling flush_dcache_page	2014-01-06 01:26:48

Po kliknutí na predmet mailu v obyčajnom alebo stromovom zobrazení sa otvorí konkrétny mail. V zobrazení koreňov sa po kliknutí na predmet otvorí celý strom mailov, ktorému patrí daný koreň.

Zobrazenie stromu pre jeden mail:

Archive netdev - Tree for email 20140105223250.GB27517@amd.pavel.ucw.cz

Subject	From	Date (DESC)
• unknown		
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-05 23:32:50
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Sebastian Reichel <sre@ring...	2014-01-06 00:01:57
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-06 01:27:55
• unknown		
• Re: [PATCH v2] Bluetooth: Add hci_h4p driver	Pali Rohár <pali.rohar@gmai...	2014-01-08 22:36:24
• Re: [PATCH v2] Bluetooth: Add hci_h4p driver	Pali Rohár <pali.rohar@gmai...	2014-02-13 16:33:28
• Re: [PATCH v2] Bluetooth: Add hci_h4p driver	Sebastian Reichel <sre@deb...	2014-02-14 18:28:03
• Re: [PATCH v2] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@denx....	2014-02-15 23:30:06
• Re: [PATCH v2] Bluetooth: Add hci_h4p driver	Ben Hutchings <ben@deca...	2014-02-19 02:12:21
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-10 00:38:43
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Sebastian Reichel <sre@ring...	2014-01-10 01:32:31
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-10 13:18:05
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Sebastian Reichel <sre@ring...	2014-01-10 14:44:26
• Re: [PATCH v4] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-10 15:49:18
• [PATCH v5] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-10 15:52:07
• Re: [PATCH v5] Bluetooth: Add hci_h4p driver	Joe Perches <joe@perches....	2014-01-10 18:33:04
• Re: [PATCH v5] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-11 01:19:52
• [PATCH v6] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-11 01:28:01
• Re: [PATCH v6] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-16 01:22:24
• Re: [PATCH v6] Bluetooth: Add hci_h4p driver	Marcel Holtmann <marcel@...	2014-01-16 04:00:40
• Re: [PATCH v6] Bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-17 13:14:51
• [PATCH v7] staging/bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-17 14:29:10
• [PATCH] Bluetooth: add module name for btwilink	Pavel Machek <pavel@ucw....	2014-01-17 14:34:30
• [PATCH v7] staging/bluetooth: Add hci_h4p driver	Pavel Machek <pavel@ucw....	2014-01-21 21:56:27

Show archive netdev
Show list of archives

Stránka pre konkrétny mail:

From: Linus Walleij <linus.walleij@linaro.org>
To: Pavel Machek <pavel@ucw.cz>
Cc: One Thousand Gnomes <gnomes@lxorguk.ukuu.org.uk>, Pali Rohár <pali.rohar@gmail.com>, Sebastian Reichel <sre@ring0.de>, Kim, Milo <milo.kim@ti.com>, Bryan Wu <cooloney@gmail.com>, David Lang <david@lang.hm>, Joe Xue <lgxue@hotmail.com>, rpurdie@rpsys.net <rpurdie@rpsys.net>, rob@landley.net <rob@landley.net>, linux-leds@vger.kernel.org <linux-leds@vger.kernel.org>, linux-kernel@vger.kernel.org <linux-kernel@vger.kernel.org>, linux-doc@vger.kernel.org <linux-doc@vger.kernel.org>
Date: Tue, 07 Jan 2014 16:40:06 +0100
Subject: Re: n900 led compiler (was Re: [PATCH] Add LED pattern trigger)
Message-Id: CACRpkdZT+fd0nZmzqwO+PVZWG_4M4Fas1SqG9CRAho6LwSAFgQ@mail.gmail.com

```
On Sun, Jan 5, 2014 at 11:23 PM, Pavel Machek <pavel@ucw.cz> wrote:

> I implemented compiler for it (should we put it into tools/ somewhere?)

We have a precedent for putting firmware compilers into the kernel
tree:
https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/drivers/scsi/aic7xxx/aicasm

But that one used FLEX. And I think the assumption is that then
you store the firmware source with the driver, and the sysfs (or similar)
interface would just load and trigger one of the pre-defined firmware
programs, not have it be sent in from userspace.

Yours,
Linus Walleij
--
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org
More majordomo info at  http://vger.kernel.org/majordomo-info.html
Please read the FAQ at  http://www.tux.org/lkml/
```

Stránka pre konkrétny mail obsahuje informácie o odosielateľoch, prijímateľoch ako aj dátum odoslania mailu, predmet a identifikátor mailu. Po kliknutí na niektorého odosielateľa či prijímateľa sa otvorí stránka vyhľadávania mailov podľa mena či mailovej adresy, na ktorú bolo kliknuté. Po kliknutí na predmet mailu sa otvorí zobrazenie stromu pre daný mail.

6.4 Terminálová aplikácia

Terminálová aplikácia slúži správcom mailových konferencií na vytvorenie a spravovanie archívu, ďalej na samotné pridávanie či mazanie mailov ako aj prácu s internými formátmi aplikácie. Terminálová aplikácia má názov súboru `plist.pl`. V ďalších sekciách tejto kapitoly, budeme predpokladať, že všetky potrebné kroky inštalácie boli úspešne vykonané a terminálovú aplikáciu môžeme spúšťať príkazom, ktorý sa zhoduje s názvom súboru. Samotný archív je adresár a ten obsahuje niekoľko súborov. Tento adresár, ktorý obsahuje špecifické súbory, budeme ďalej označovať ako indexovaný archív alebo index.

Terminálová aplikácia berie vždy ako prvý argument použitý mód, ako druhý argument príkaz a ďalšie argumenty sú parametre pre konkrétny príkaz.

6.4.1 Práca s indexovaným archívom – mód `index`

Na prácu s indexovaným archívom slúži mód `index`. Dokáže vytvárať archívy, pridávať maily do archívov, odstraňovať maily, čítať jednotlivé časti mailov a generovať HTML dokumenty z mailov.

Všetky príkazy módu `index` majú povinný prvý parameter `dir` a je to názov archívu. Názov archívu predstavuje adresár a až na výnimku príkazu `create`, musí existovať. Pre príkaz `create` naopak nesmie existovať. Špecifikovaný príkaz pracuje s archívom, ktorý je zvolený práve parameterom `dir`.

Príkazy módu `index`:

- Príkaz `create` slúži na vytvorenie prázdneho archívu. Jeho parametre sú:

```
create <dir> [<driver>] [<params>] [<username>] [<password>]
```

  Parameter `dir` predstavuje indexovaný adresár archívu, ktorý sa vytvorí. Parameter `driver` určuje typ databázy (podporované sú `SQLite` a `mysql`). Ak nie je typ databázy zadaný, použije sa `SQLite`. Parametre `params`, `username` a `password` špecifikujú spôsob akým sa aplikácia bude pripájať k databáze. Pri použití `SQLite` sa databáza ukladá do súboru `sqlite.db` v zvolenom adresári a tieto parametre určené na pripojenie k databáze sa nepoužívajú. Pri zvolení databázy `mysql`, parameter `params` určuje názov databázy a adresu servera oddelenými dvojbodkou. Parametre `username` a `password` určujú prihlasovacie meno a heslo do databázy.

- Príkaz `add-mbox` slúži na pridanie mailov z *MBox* archívu a má parametre:

```
add-mbox <dir> [<mbox>] [silent]
```

  Parameter `mbox` predstavuje vstupný *MBox* súbor, z ktorého sa maily budú čítať. Ak nie je uvedený, použije sa štandardný vstup. Posledný nepovinný parameter `"silent"` spôsobí, že aplikácie nevypíše chyby, ak sa niektorý mail do archívu pridať nepodarí. Vhodné ak *MBox* súbor obsahuje duplicitné maily, nakoľko nie je možné do archívu pridať maily s rovnakým identifikátorom (`Message-Id`).

- Príkaz `add-mail` slúži na pridanie jedného mailu do archívu a má parametre:

```
add-mail <dir> [<mail>]
```

  Parameter `mail` predstavuje vstupný súbor s mailom, ktorý sa pridá do archívu a má byť vo formáte RFC2822 [1]. Ak vstupný súbor nie je zadaný, použije sa štandardný vstup.

- Príkaz `del` slúži na odstránenie mailu z archívu. Jeho parametre sú:

```
del <dir> <id>
```

  Parameter `id` je identifikátor mailu – konkrétne `Message-Id`.

- Príkaz `get-bin` vráti špecifikovaný mail v internom formáte. Parametre sú:

```
index get-bin <dir> <id> [<bin>]
```

  Parameter `id` predstavuje identifikátor mailu (`Message-Id`) a parameter `bin` názov výstupného súboru. Ak nie je súbor zadaný, použije sa štandardný výstup.

- Príkaz `get-part` vráti niektorú časť mailu a má parametre:

```
index get-part <dir> <id> <part> [<file>]
```

  Parameter `id` predstavuje identifikátor mailu (`Message-Id`), parameter `part` identifikátor zvolenej časti a parameter `file` je názov výstupného súboru. Ak súbor nie je zadaný, použije sa štandardný výstup. Identifikátor jednotlivých častí mailu je možné zobraziť príkazom `view` v móde `bin`.

- Príkaz `gen-html` slúži na vygenerovanie HTML dokumentu pre jeden konkrétny mail. Jeho parametre sú:

```
index gen-html <dir> <id> [<html>]
```

  Parameter `id` predstavuje identifikátor mailu (`Message-Id`) a parameter `html` názov výstupného súboru. Ak výstupný súbor nie je zadaný, použije sa štandardný výstup.

Príklady módu `index`:

Vytvorenie nového prázdneho archívu s názvom `lkml` a s požitím *SQLite*:

```
$ plist.pl index create lkml
```

Vytvorenie nového prázdneho archívu s názvom `test` a s požitím databázy *MySQL*. Konkrétne zvolená *MySQL* databáza má názov `testdb`, databázový server beží na adrese `localhost` a na pripojenie sa používa prihlasovacie meno `user` a prihlasovacie heslo `password`:

```
$ plist.pl index create test mysql testdb:localhost user password
```

Pridanie jedného mailu zo štandardného vstupu do archívu `lkml`:

```
$ plist.pl index add-mail lkml
```

Pridanie jedného mailu zo súboru `mail.rfc822` do archívu `test`:

```
$ plist.pl index add-mail test mail.rfc822
```

Pridanie celého *MBox* archívu zo súboru `archive.mbox` do archívu `lkml` bez oznamovania informácií o chybách:

```
$ plist.pl index add-mbox lkml archive.mbox silent
```

Odstránenie mailu s identifikátorom `201406241206@example.org` z archívu databázy `test47`:

```
$ plist.pl index del test47 201406241206@example.org
```

Uloženie mailu `id4247@test` z archívu `arch` do súboru `file.bin` v internom formáte:

```
$ plist.pl index get-bin arch id4247@test file.bin
```

Uloženie časti `0/0/1` z mailu `id4742@test` z archívu `arch` do súboru `file.pdf`:

```
$ plist.pl index get-part arch id4742@test 0/0/1 file.pdf
```

Vygenerovanie HTML stránky do súboru `mail.html` pre mail s identifikátorom `201406241205@example.org` z archívu `test`:

```
$ plist.pl index gen-html test 201406241205@example.org mail.html
```

6.4.2 Práca s jedným mailom – mód `bin`

Na prácu s jedným mailom slúži mód `bin`. Za normálnych okolností to správca ani užívateľ nepotrebuje. Práca s jedným mailom má význam vtedy, ak je potrené vygenerovať HTML stránku z konkrétneho MIME mailu. Alebo vtedy, keď je potrebné odladiť nejakú chybu vo formátovaní mailu, alebo zistiť aké dáta sú uložené v tomto internom formáte programu. Mód `bin` obsahuje tieto príkazy:

- Príkaz `view` slúži na zobrazenie štruktúry mailu, ktorý je uložený v internom formáte aplikácie. Ako jediný parameter je názov súboru, kde je mail uložený. Ak názov súboru nie je zadaný, mail v internom formáte sa bude očakávať na štandardnom vstupe.
- Príkaz `from-mail` slúži na konverziu mailu z formátu MIME do interného formátu. Prvý parameter je názov vstupného súboru, druhý je názov výstupného. Ak parametre nie sú zadané, použije sa štandardný vstup a výstup.
- Príkaz `get-part` slúži na získanie konkrétnej časti mailu (zoznam častí vypíše príkaz `view`). Parametre sú: názov časti, názov vstupného súboru, názov výstupného súboru.
- Príkaz `gen-html` slúži na vygenerovanie HTML stránky z mailu. Prvý parameter je vstupný súbor mailu, druhý parameter je názov výstupného html súboru. Ak parametre nie sú zadané, použije sa na prácu štandardný vstup a výstup.

Príklad:

Vygenerovanie HTML stránky na štandardný výstup z MIME mailu, ktorý je na štandardnom vstupe:

```
$ plist.pl bin from-mail | plist.pl bin gen-html
```

6.4.3 Práca so zoznamom mailov – mód `list`

Aplikácia používa vlastný formát na interné ukladanie zoznamu mailov. Za normálnych okolností s týmito archívmi pracovať nie je potrebné, avšak v prípade problémov sú dostupné príkazy na prácu s týmto typom formátu archívov. Slúži k tomu mód `list` a obsahuje tieto príkazy:

- Príkaz `view` pre špecifikovaný súbor so zoznamom mailov zobrazí informácie o každom jednom maili. Zobrazí aj dôležitú položku `offset`, ktorá je potrebná pre ďalšie príkazy, ktoré extrahujú konkrétny mail z archívu.
- Príkaz `add-mbox` pridá na koniec súboru maily z *MBox* archívu. Prvý argument je názov súboru so zoznamom mailov, druhý je názov *MBox* súboru, z ktorého sa maily pridajú. V prípade, že výstupný súbor s mailami na disku neexistuje, tak sa automaticky vytvorí.
- Príkaz `add-bin` slúži na pridanie jedného mailu v internom formáte (viď mód `bin`) do súboru s mailami. Prvý argument je názov výstupného súboru, do ktorého sa mail pridá, druhý argument je názov vstupného súboru.
- Príkaz `get-bin`, podobne ako v móde `index` vráti mail v internom formáte módu `bin`. Prvý argument je názov zoznamu, druhý je pozícia v súbore kde mail začína a tretí nepovinný je názov výstupného súboru. Pozíciu, kde konkrétny mail začína, udáva položka `offset` z výstupu príkazu `view`.
- Príkaz `get-part` pre názov súboru so zoznamom, pozíciu mailu a názov časti (v tomto poradí) vráti binárny obsah špecifikovanej časti.
- Príkaz `gen-html` pre názov súboru so zoznamom a pozíciu mailu vygeneruje HTML stránku.

Príklady:

Konverzia *MBox* archívu `file.mbox` do interného formátu `file.list`:

```
$ plist.pl list add-mbox file.list file.mbox
```

Vyextrahovanie mailu, ktorý začína na pozícií 1024 v súbore `file.list` do súboru `file.bin`:

```
$ plist.pl list get-bin file.list 1024 file.bin
```

6.4.4 Pokročilé nastavenia indexovaného archívu

Indexovaný archív obsahuje konfiguračný súbor `config`. Tento súbor obsahuje na každom riadku práve jednu dvojicu kľúča a hodnoty oddelenými znakom `=`. Každý riadok sa končí znakom `LF`. V tomto súbore je možné meniť pokročilé nastavenia akým webová a terminálová aplikácia pracujú s archívom. Nachádzajú sa tu dôležité nastavenia potrebné na pripojenie k databáze.

Kľúče na pripojenie k databáze:

Názov kľúča	*Význam hodnoty kľúča*
`driver`	Použitá databáza: `SQLite` alebo `mysql`. Štandardne sa používa `SQLite`.
`params`	Parametre pre pripojenie k databáze. Pri použití `SQLite` je to názov databázového súboru, ktorý je uložený v adresári *index*. Ak nie je zadaný, použije sa súbor `sqlite.db`. Pri `mysql` je to názov databázy a adresa servera oddelená dvojbodkou. Pre tento parameter `params` je ešte možné použiť alternatívnu syntax podľa použitého perlového modulu (`DBD::SQLite` alebo `DBD::mysql`).
`username`	Prihlasovacie meno na pripojenie do databázy. Pri `SQLite` je kľúč ignorovaný a nemusí byť zadaný.
`password`	Prihlasovacie heslo na pripojenie do databázy. Pri `SQLite` je kľúč ignorovaný a nemusí byť zadaný.

Ostatné kľúče:

Názov kľúča	*Význam hodnoty kľúča*
`description`	Popis mailového archívu alebo mailovej konferencie. Zobrazuje sa vo webovej aplikácií a štandardne je prázdny.
`listsize`	Orientačná maximálna veľkosť jedného súboru so zoznamom mailov. Štandardne je to 100 MB.
`nomatchsubject`	Ak je hodnota nastavená na 1, potom každé dva maily s podobným predmetom sa budú nachádzať v rovnakom vlákne. Túto hodnotu je možné zmeniť, iba ak je archív úplne prázdny a neobsahuje zatiaľ žiadne maily. Je to preto, že zaraďovanie mailov do vlákien sa robí už v čase, keď sa mail pridáva do archívu a databázy. Zmena hodnoty pre neprázdny archív spôsobí problémy so zaraďovaným nových mailov do vlákien. Štandardne je táto hodnota nastavená na 0.

Pomocou kľúča `listsize` je možné nastaviť orientačnú veľkosť, ktorú môže jeden súbor na disku dosiahnuť. Tento kľúč je vhodné nastaviť, ak sú súbory na disku uložené v súborovom systéme, ktorý je optimalizovaný buď na veľa malých súborov alebo málo veľkých. Veľkosť je orientačná preto, lebo terminálová aplikácia sa môže rozhodnúť pridať do súboru ešte jeden mail, ak má súbor malú veľkosť. Je to kvôli tomu, aby sa predišlo vytváraniu nových súborov kvôli veľkým mailom. Nestane sa ale, že aplikácia bude pridávať ďalší mail do súboru, ktorý už prekročil veľkosť definovanú v `listsize`.

6.4.5 Pomocný skript na spracovanie *MBox* súborov

Pre účely pridávania mailov z veľkého počtu *MBox* súborov do indexovaného archívu existuje pomocný skript s názvom `plist-import-mboxes.pl`. Ten je dodávaný spoločne s terminálovou aplikáciu a pre správnu funkčnosť sa musí nachádzať v rovnakom adresári ako samotná terminálová aplikácia.

Skript slúži na rýchle preskočenie *MBox* archívov, ktoré už boli do indexovaného archívu pridané a od posledného spracovania sa nezmenili. Je vhodný na periodické spúšťanie za účelom pridávania nových mailov z veľkej množiny *MBox* archívov, nakoľko pracuje iba s novými dátami.

Skript používa v adresári indexovaného archívu súbor `timestamps`. Doňho si pre každý spracovaný *MBox* súbor ukladá dátum jeho poslednej zmeny. Na každom riadku tohoto súboru je dvojica oddelená medzerou: dátum a názov súboru. Dátum je udaný v počte sekúnd od 01. 01. 1970. Na pridanie *MBox* archívu, ktorý ešte nebol spracovaný (alebo sa zmenil), skript volá priamo terminálovú aplikáciu `plist.pl` so správnymi parametrami.

Skript `plist-import-mboxes.pl` očakáva ako prvý parameter názov adresára indexovaného archívu. Ďalšie parametre predstavujú *MBox* súbory, z ktorých sa budú postupne pridávať maily. Ich počet je obmedzený iba maximálnym počtom argumentov, ktoré vie operačný systém a interpreter Perl spracovať. Je odporúčané uvádzať názvy súborov s ich plnou cestou, nakoľko sa argumenty priamo ukladajú do súboru `timestamps`. Za posledným súborom môže byť ešte špecifikovaný parameter `"silent"`, ktorý má rovnaký význam ako pri terminálovej aplikácií (nebudú sa oznamovať chyby pri pridávaní mailov). Použitie tohoto argumentu je odporúčané, lebo `plist.pl` sa snaží pridať všetky maily z *MBox* súboru a nie len posledný pridaný.

Príklady:

Pridanie všetkých mailov do indexovaného archívu `lkml` z dvoch *MBox* súborov `/201401.mbox` a `/201402.mbox` a nevypisovať pri tom žiadne chyby:

```
$ plist-import-mboxes.pl lkml /201401.mbox /201402.mbox silent
```

Pridanie všetkých mailov do indexovaného archívu `lkml` z *MBox* súborov, ktoré majú koncovku `.mbox` a sú v adresárovej hierarchii `/lkml/` (bez vypisovania chýb):

```
$ plist-import-mboxes.pl lkml $(find /lkml/ -name *.mbox) silent
```

Záver

V tejto práci sú dôkladne popísané problémy, ktoré musí program na spracovávanie mailov z mailových konferencií vedieť vyriešiť. K jednotlivým problémom sú popísané aj možné riešenia, prevažne tie, ktoré sa snažia použiť čo najviac dostupných informácií z mailov a položiek v ich hlavičkách. V práci sú popísané formáty spájajúce sa s mailami a súvisiace postupy ako ich rozumne spracovať. Okrem toho je súčasťou práce referenčná implementácia programu, ktorý archivuje a zobrazuje maily z mailových konferencií.

Medzi hlavné prínosy tejto práce patrí:

- Dodanie efektívneho programu na archiváciu mailov z mailových konferencií, ktorý dokáže spracovať aj veľké množstvo mailov, ktoré nespĺňajú požadované mailové štandardy
- Oboznámenie čitateľa s používanými mailovými formátmi a problémami, ktoré tieto formáty v praxi často prinášajú
- Oboznámenie čitateľa s možnými riešeniami, ako interpretovať a spracovať poškodené maily a mailové archívy
- Popis a implementácia algoritmu na rozumné zostavenie stromov mailových konverzácií – čo je použiteľné aj v iných mailových aplikáciach (ako sú mailoví klienti na prácu so súkromnými mailovými správami)

Práca sa síce zaoberá mailami a spôsobom, ako ich archivovať ale nezaoberá sa obsahom a významom jednotlivých mailov. Z tejto skutočnosti vyplýva hneď niekoľko možných rozšírení programu, ktorý maily spracováva.

Program by mohol podľa obsahu správ vedieť detegovať spam, automaticky takéto správy filtrovať a nezobrazovať ich ani vo webovej aplikácií. Uľahčil by tým čítanie archívnych správ účastníkom mailových konferencií. Ďalej by program mohol vedieť detegovať citácie mailov a umožniť ich užívateľovi vo webovej aplikácii zobraziť, alebo skryť. Citácie mailov by sa mohli dať využiť k zisteniu, na ktorý mail je daný mail odpoveďou, ak chýbajú informácie v položke `In-Reply-To`. Možné rozšírenie programu využívajúce obsah mailu by mohlo byť aj detegovanie patch súborov v tele mailu, ako to má služba *LKML.ORG*.

Za zmienku stoja aj iné možné rozšírenia programov, ktoré nevyužívajú obsah jednotlivých mailov. Môže to byť napríklad podpora pre relačnú databázu *PostgreSQL*, ktorá je popri *MySQL* na serveroch dosť používaná.

Ďalej sa tu naskytuje možnosť integrovať program tejto práce priamo do serverového programu *Mailman*, ktorý sa stará o chod mailových konferencií. Program *Mailman* má webové rozhranie, v ktorom si jednotliví členovia konferencie môžu po prihlásení meniť rôzne nastavenia. Stálo by za to, aby sa rovnaké prihlasovacie údaje dali použiť pri prístupe k privátnym archívom mailových konferencií.

Keďže je program modulárny a používa abstraktné rozhrania na prácu s mailami a archívmi, tak do budúcna je možné vytvoriť ďalšie moduly, ktoré by vedeli spracovávať maily aj v iných formátoch. Niektoré servery môžu používať iné formáty na ukladanie mailov (napr. *MailDir*) a vďaka modularite programu by bolo možné pridať podporu aj pre ne.

Zoznam použitej literatúry

[1] Resnick, P., *Internet Message Format*, RFC 2822, 2001. Dostupné z: http://tools.ietf.org/html/rfc2822

[2] Crocker, D., *Standard for ARPA Internet Text Messages*, STD 11, RFC 822, 1982. Dostupné z: http://tools.ietf.org/html/rfc822

[3] Freed, N., Borenstein N., *Multipurpose Internet Mail Extensions MIME Part One: Format of Internet Message Bodies*, RFC 2045, 1996. Dostupné z: http://tools.ietf.org/html/rfc2045

[4] Freed, N., Borenstein N., *Multipurpose Internet Mail Extensions MIME Part Two: Media Types*, RFC 2046, 1996. Dostupné z: http://tools.ietf.org/html/rfc2046

[5] Moore, K., *MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text*, RFC 2047, 1996. Dostupné z: http://tools.ietf.org/html/rfc2047

[6] Troost, R., Dorner, S., Moore, K., *Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field*, RFC 2183, 1997. Dostupné z: http://tools.ietf.org/html/rfc2183

[7] Freed, N., Moore, K., *MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations*, RFC 2231, 1997. Dostupné z: http://tools.ietf.org/html/rfc2231

[8] Yang, A., Steele, S., Freed, N., *Internationalized Email Headers*, RFC 6532, 2012. Dostupné z: http://tools.ietf.org/html/rfc6532

[9] Bernstein, D. J., *Using maildir format* [online]. Dostupné z: http://cr.yp.to/proto/maildir.html

[10] Bernstein, D. J., Dhesi R., Message Delimiters. In: *Usenet group: comp.mail.headers* [online]. 1995–1996. Dostupné z: http://groups.google.com/d/msg/comp.mail.headers/Q6GXtTBBJys/Tuz6FnKHub8J

[11] Hall, E., *The application/mbox Media Type*, RFC 4155, 2005. Dostupné z: http://tools.ietf.org/html/rfc4155

[12] How to Get Your Change Into the Linux Kernel, Section 1, Subsection 7. In: *Linux Kernel Documentation* [online]. 2014 [cit. 2014-07-17]. Dostupné z: http://www.kernel.org/doc/Documentation/SubmittingPatches

[13] Bug #42834, HTML::Strip breaks UTF-8. In: *Bugs for HTML-Strip* [online]. 2009 [cit. 2014-07-17]. Dostupné z: http://rt.cpan.org/Public/Bug/Display.html?id=42834#txn-705624

Prílohy

Súčasťou práce je optické médium, ktoré obsahuje zdrojové kódy vypracovaného programu, testovacie dáta a elektronickú verziu tejto práce. V nasledujúcej tabuľke je uvedená adresárová štruktúra a súbory priloženého optického média.

`/Email/Folder/`	Modifikovaný perlový modul `Email::Folder`
`/PList/`	Interné perlové moduly potrebné pre program
`/.htaccess`	Konfiguračný súbor pre webový server Apache
`/COPYING`	Licencia pre používanie a šírenie programu
`/lkml-archive.mbox`	Testovací MBox súbor z konferencie LKML
`/plist-import-mboxes.pl`	Pomocný skript na spracovávanie MBox súborov
`/plist.cgi`	Webová aplikácia vo formáte CGI skriptu
`/plist.pl`	Terminálová aplikácia programu v jazyku Perl
`/praca.pdf`	Elektronická verzia práce vo formáte PDF