Система идентификации событий работы протоколов при обмене данными между двумя сетевыми ЭВМ — DaCoPAn Analyzer

Докладчики:

Кирилл Кулаков, Михаил Крышень, Андрей Ананьин

Организация:

- Петрозаводский государственный университет, каф. Информатики и математического обеспечения

Руководители проекта:

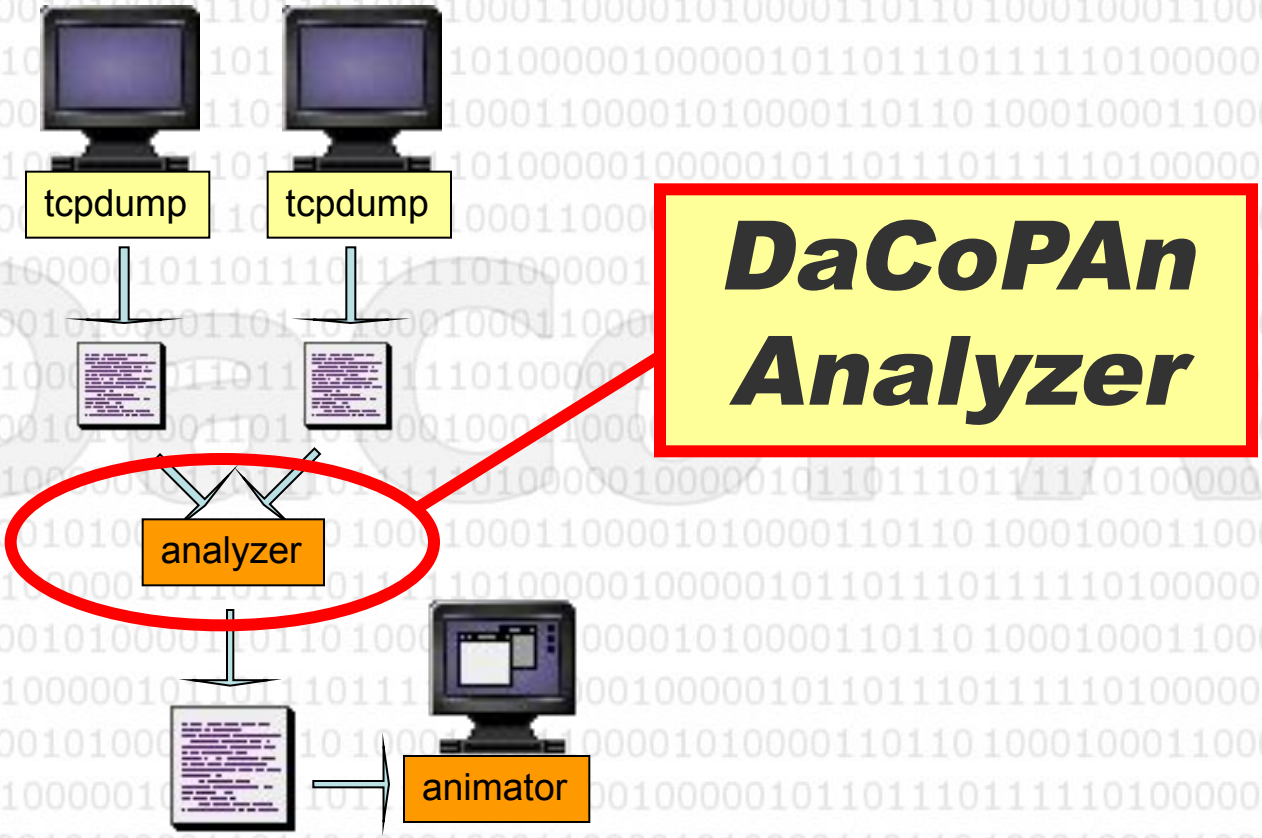
- Юрий Анатольевич Богоявленский (зав. каф., доцент, к.т.н.)
- Дмитрий Жоржевич Корзун (ст. преп., к.ф.-м.н., инструктор)

ПО для разработки

Система DaCoPAn Analyzer:

- Идентификация событий работы протоколов при обмене данными между сетевыми ЭВМ.
- Представление данных в формате XML
- Разработана для совместного использования с системой DaCoPAn Animator

Проект DaCoPAn



Область применения

Научные исследования:

- Удобное представление событий при работе сетевых протоколов
- Использование файла событий для обработки данных сетевого трафика

Образование:

- Преподавание сетевых технологий
- Пробный трансграничный студенческий проект

Технология производства ПО:

- Технология трансграничной разработки программного продукта

Задачи

- ПО для обучения и исследования
- Стандарты ТППО
- Коллективная работа
- Распределенная трансграничная разработка ПО (Хельсинкский университет)
- Кросс-платформенные и переносимые технологии

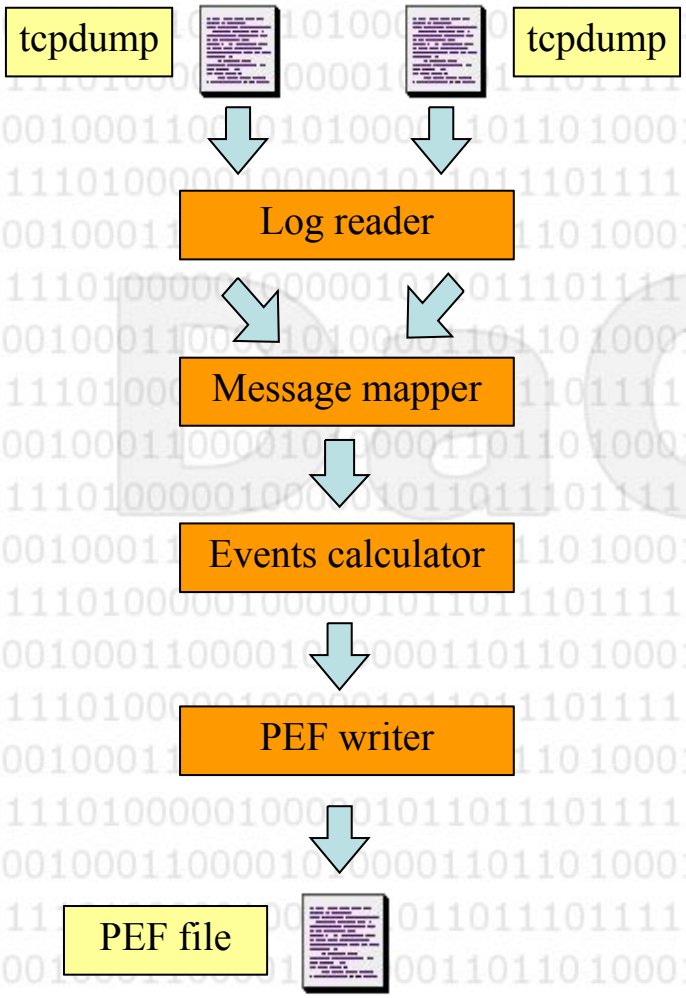
Концепция

- Проект по разработке реального ПО
- Консольное приложение
- Удобство использования
- Соответствие STD Интернет
- Работа с реальными данными

Предметная область

- Анализ взаимодействия двух сетевых ЭВМ
- Протоколы обмена данными
- Исходные данные в виде бинарных файлов утилиты tcpdump (WinDump)
- Анализ заголовков пакетов
- Поиск соответствующих событий для каждого пакета
- Обработка нестандартных ситуаций

Архитектура



Получение двух файлов tcpdump на двух компьютерах.

Чтение каждого файла tcpdump.

Поиск соответствующих пакетов протоколов и объединение пакетов в один список.

Преобразование последовательности пакетов в последовательность событий, вычисление переменных протоколов и дополнительных событий.

Запись последовательности сообщений в файл в формате PEF.

Получен файл событий протоколов.

Вывод tcpdump

tcpdump log 1

```
17:13:45.955758 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF)
17:13:45.955933 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:45.956007 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF)
17:13:45.981674 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 51 (DF)
17:13:45.981817 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
17:13:47.524336 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 14 (DF) [tos 0x10]
17:13:47.524497 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:47.524648 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 34 (DF)
17:13:47.524675 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
17:13:50.956903 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 15 (DF) [tos 0x10]
17:13:50.996908 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:51.032471 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 33 (DF)
17:13:51.032499 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
```

tcpdump log 2

```
17:13:45.938659 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF)
17:13:45.938700 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:45.938902 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF)
17:13:45.964420 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 51 (DF)
17:13:45.964717 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
17:13:47.507242 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 14 (DF) [tos 0x10]
17:13:47.507264 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:47.507409 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 34 (DF)
17:13:47.507572 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
17:13:50.939818 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 15 (DF) [tos 0x10]
17:13:50.979665 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 0 (DF)
17:13:51.015232 zeta.cs.karelia.ru.ftp > iota.cs.prv.dcs: tcp 33 (DF)
17:13:51.015397 iota.cs.prv.dcs > zeta.cs.karelia.ru.ftp: tcp 0 (DF) [tos 0x10]
```

Файл событий протоколов

```
<layers>
    <layer id="L1" name="network">
        <protocol id="P2" name="net_unknown"/>
        <protocol id="P0" name="ipv4"/>
    </layer>
    ...
</layers>
...
<unit_sent id="U1" source="H1" destination="H2" protocol="P3"
time="0.000000"
        children="U2" flow="F1">
    <value name="sent_time">0.000000</value>
    <value name="trans_time">0.000117</value>
    <value name="source_port">1367</value>
    <value name="dest_port">21</value>
    <value name="seq">900322900</value>
    <value name="ack_seq">0</value>
    <value name="window">5840</value>
    <value name="urg_pointer">0</value>
...
```

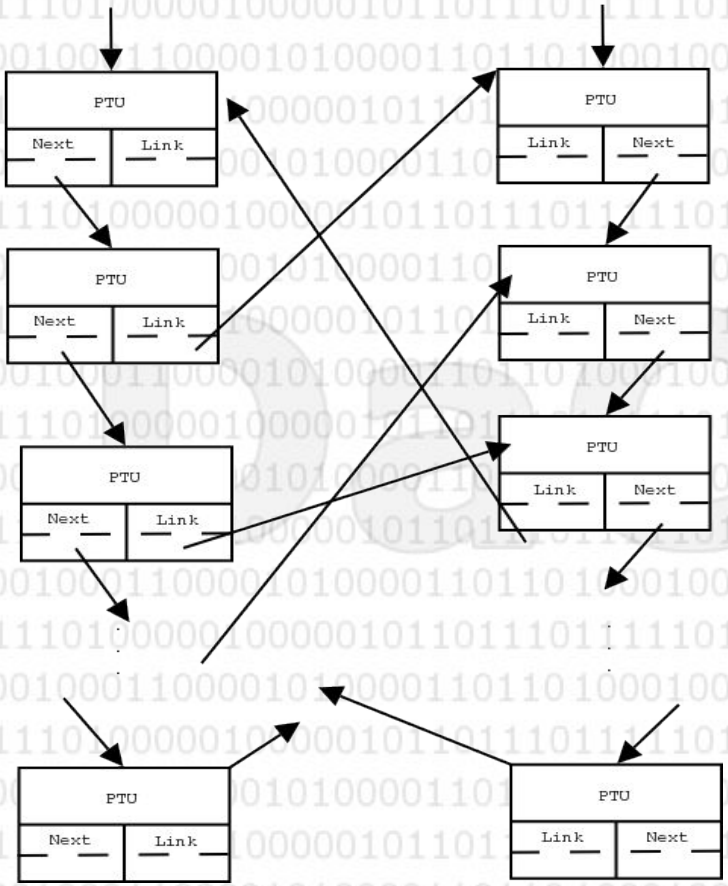
Файл событий протоколов

- Расширяемый формат в стандарте XML, описанный DTD
- Информация о структуре сети, используемых протоколах (многоуровневая модель), данные сетевого трафика в виде списка событий
- Структура файла
  - Список сетевых ЭВМ
  - Список соединений
  - Список потоков данных
  - Список протоколов по уровням
  - Список переменных протоколов (поля и вычисляемые переменные)
  - Список событий протоколов (отправка, получение, потеря сообщения)

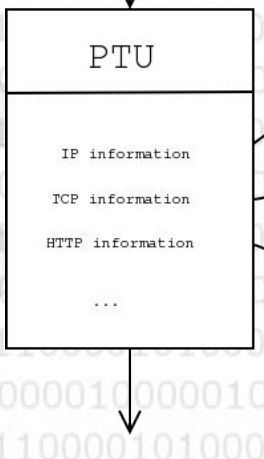
Алгоритмы

- Поиск соответствующих друг другу пакетов в двух файлах tcpdump
- Определение потерянных пакетов
- Вычисление переменных протоколов
- Преобразование последовательности пакетов в последовательность событий:
  - Дефрагментация
  - Построение дерева инкапсуляции

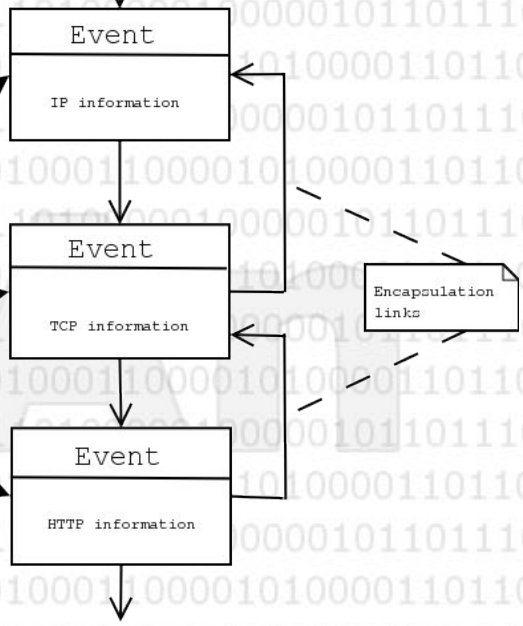
Структуры данных



Списковая структура для представления и обработки исходных tcpdump файлов

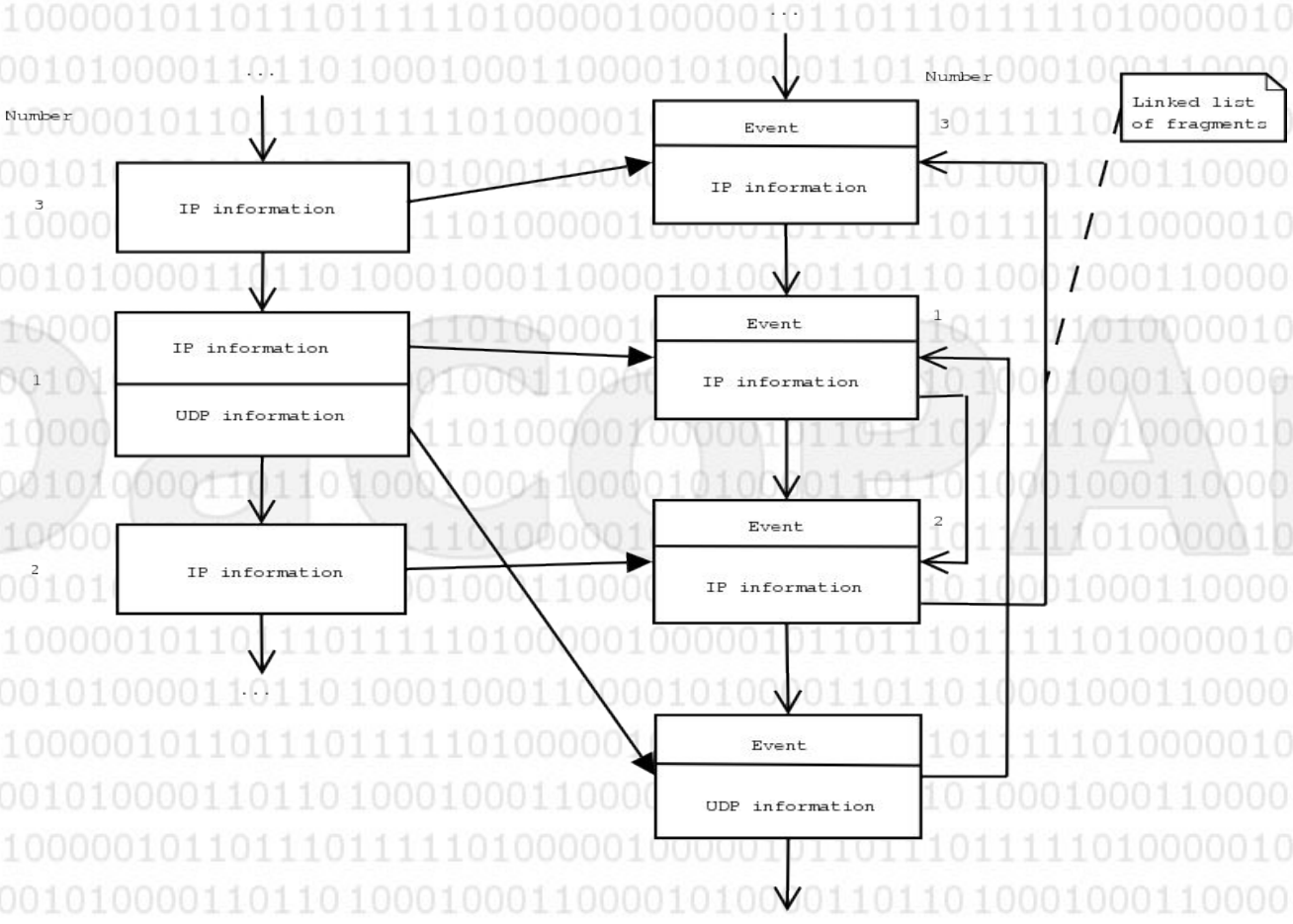


Элемент списка



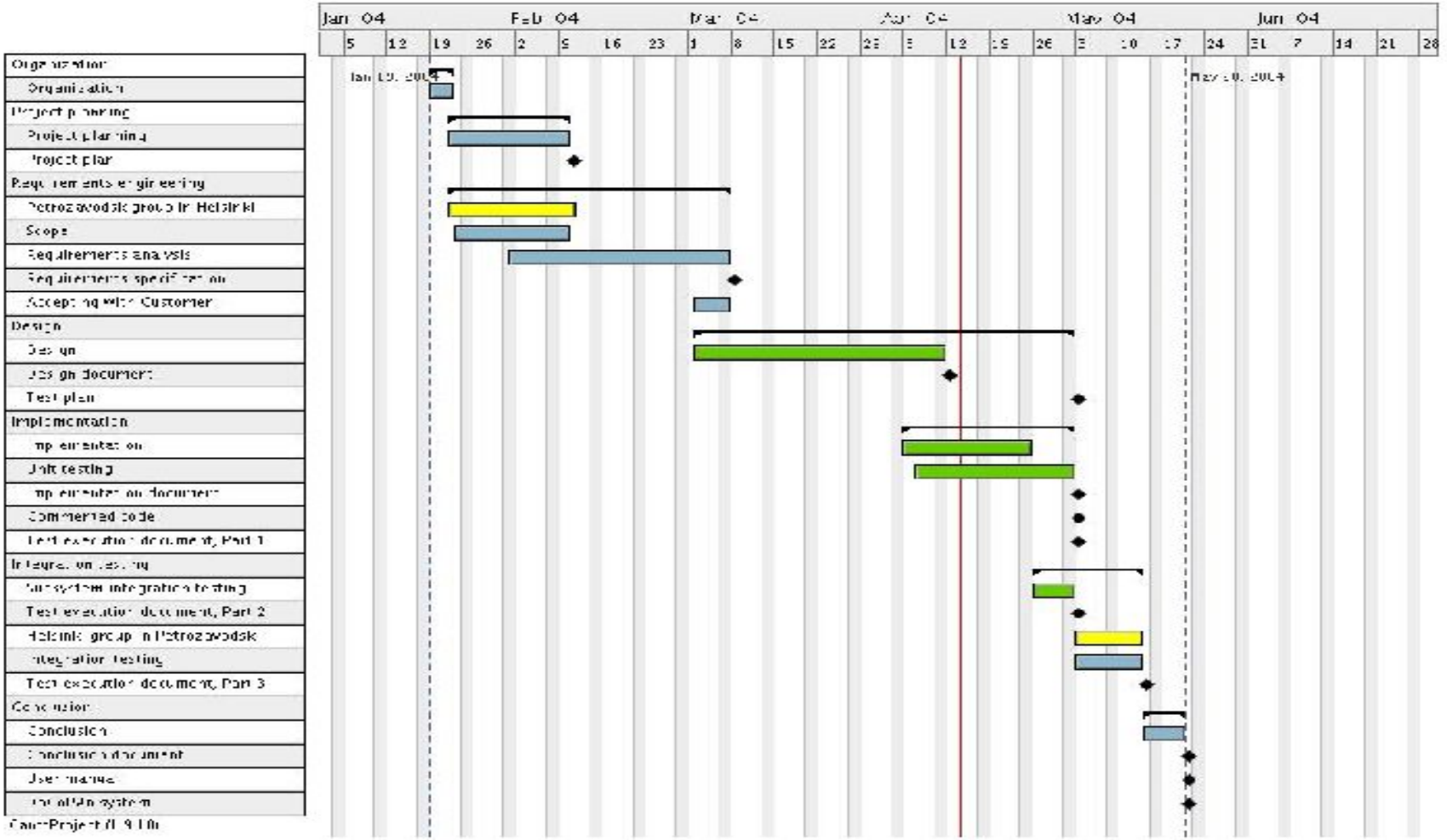
Списковая структура для представления событий протоколов

Структуры данных



Представление фрагментов в списке событий

Расписание проекта



Команда

Заказчик:

Маркку Койо

Руководители проекта:

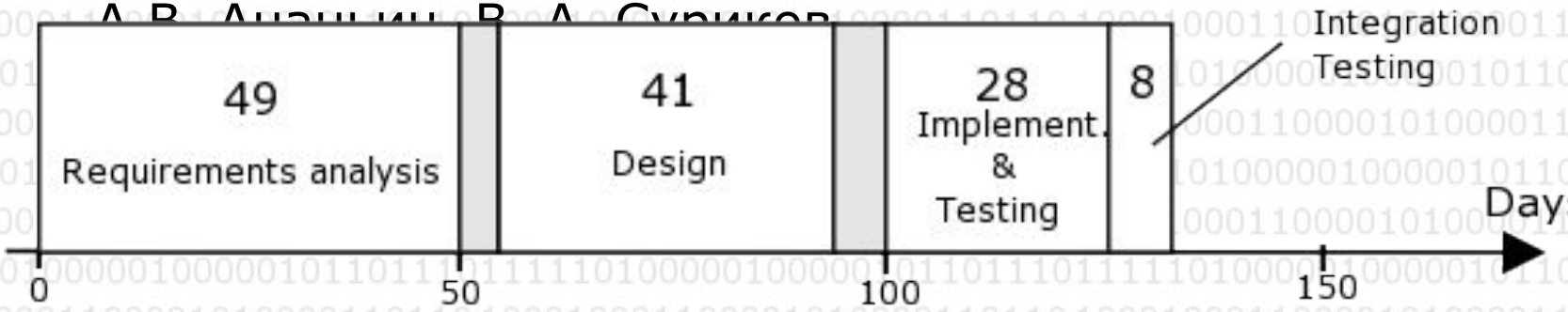
Ю.А. Богоявленский

Д. Ж. Корзун (инструктор)

Разработчики:

К.А. Кулаков, М.А. Крышень, А.Ю. Сало,

А.В. Ананьин, В.А. Суриков



Инструменты

Переносимость:

- ANSI C, и стандарт POSIX
- Microsoft Visual C++

Моделирование:

- UML – разработка модели прецедентов

Командная работа:

- CVS репозиторий, Web-сайт проекта, форум, e-mail

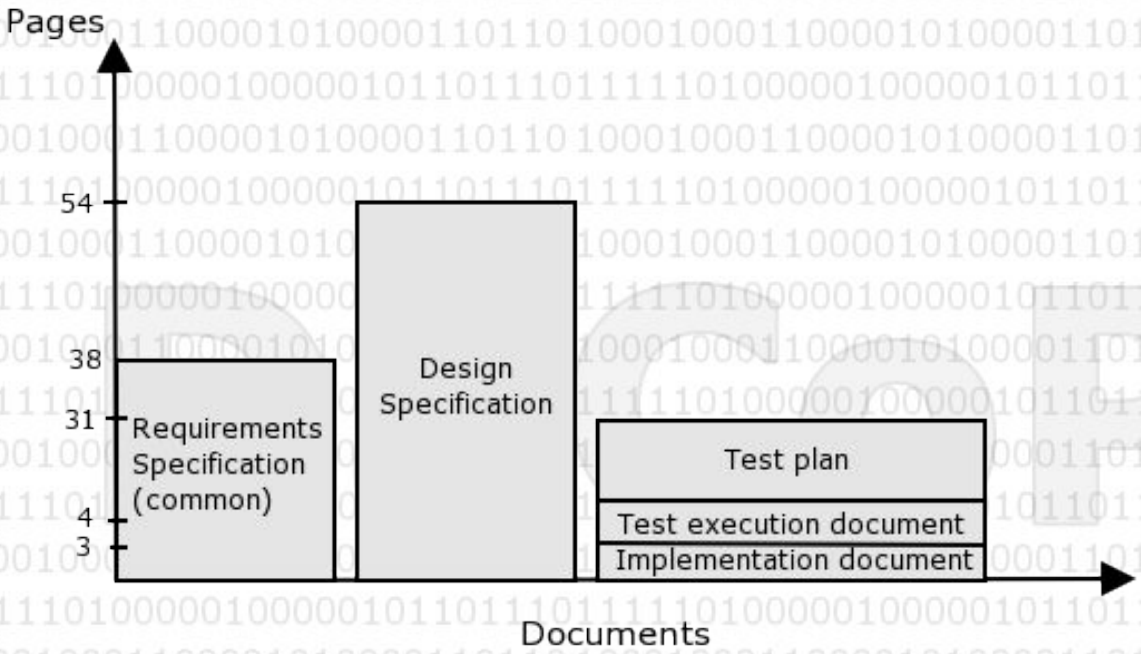
Метрики проекта:

- SCLC – подсчет количества строк исходного кода
- Gantt Project – расписание проекта

Инструменты разработки:

- automake, autoconf, WinPcap, libpcap, WinDump, tcpdump
- Служебные программы и шаблоны документов, предоставленные университетом Хельсинки

Размер артефактов



- 27 модулей
- 111 подпрограмм

	Program. language	Lines	Blank	Comments	NCSL
Analyzer	ANSI C	6333	898	1667	3768

Тестирование

Тестирование блоков:

Автоматизированная система тестирования (automake)

Интеграционное тестирование:

Тестирование программы Analyzer

Совместное тестирование программ Analyzer и Animator

Проверка требований:

На основе сценариев tcpdump-файлы, предоставленные заказчиком и реальные данные

Строки кода тестов:	1416
Тесты блоков:	18
Интеграционные тесты:	78
Проверочные тесты:	9

Возможности

- Поддержка двух сетевых ЭВМ
- Работа с реальными данными
- Поддержка протоколов ARP, IP, UDP, TCP, FTP, DNS, HTTP
- Восстановление данных прикладного уровня из сегментов транспортного уровня
- Возможность задания номеров портов для протоколов прикладного уровня
- Расширяемость при добавлении новых протоколов
- Проверка соответствия времени на двух сетевых ЭВМ и возможность задания поправки

Технологии Microsoft

- Переносимое приложение (Windows и UNIX)
- Реализация на языке C (ANSI, POSIX)
- Windows версия реализована на платформе Microsoft Visual C++
- Соответствие стандартам STD Интернет

Заключение

- Система анализа трафика, соответствующая стандартам STD Интернет
- Расширяемый формат для представления данных трафика
- Аспекты образования и подготовки специалистов
- Международные стандарты ТППО