

Универзитет у Новом Саду, Факултет техничких наука

ОАС Софтверско инжењерство и информационе технологије

Организација података

# Оперативни систем у раду над подацима

# Садржај

- **Увод**
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- Накнадна разматрања
- Ресурси

- Оперативни систем (Mogin, 2008)
  - оперативни систем је скуп програма и механизма који су намењени за управљање радом хардверских и софтверских ресурса рачунарског система
  - врсте оперативних система у односу на ниво услуга при управљању разменом података
    - **оперативни системи с услугама ниског нивоа**
    - **оперативни системи с услугама високог нивоа**

- Оперативни систем (Mogin, 2008)
  - врсте оперативних система у односу на ниво услуга при управљању разменом података
    - оперативни системи с услугама ниског нивоа
      - услуге ниског нивоа
        - услуге посвећене обезбеђењу меморијског простора за датотеку на екстерном меморијском уређају (ЕМУ) и управљању разменом података између оперативне меморије (ОМ) и ЕМУ
    - примери
      - разни савремени оперативни системи
        - оперативни системи фамилије *Unix*, оперативни системи фамилије *Linux*...

- Оперативни систем (Mogin, 2008)
  - врсте оперативних система у односу на ниво услуга при управљању разменом података
    - оперативни системи с услугама високог нивоа
      - услуге високог нивоа
        - услуге ниског нивоа
        - манипулација логичком структуром података датотеке
        - изградња посебних помоћних структура података за побољшање ефикасности коришћења датотеке
        - тражење у датотеци засновано на вредности података
      - примери
        - оперативни системи мејнфрејм рачунара

# Увод

- Оперативни систем (Mogin, 2008)
  - главни задаци рутина и механизма оперативног система које су посвећене управљању подацима
    - управљање простором на ЕМУ с директним приступом
    - управљање каталогом
    - припрема и управљање физичком разменом података између ОМ и ЕМУ
    - успостављање везе (спреге) с датотеком на ЕМУ
    - пружање услуга корисничким програмима путем системских позива за рад над датотекама
    - пружање услуга управљања и коришћења датотека с комплексном организацијом путем метода приступа
      - доступно у оперативним системима високог нивоа

# Увод

- Оперативни систем

- датотечки систем

(Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.)

- датотечки систем (енгл. *file system*) је део оперативног система у чијој надлежности је управљање перзистентним подацима
    - два основна аспекта датотечног система
      - подаци
        - структуре података на ЕМУ за потребе организовања података и метаподатака
      - методе
        - операције путем којих могу над структурама података бити остварени системски позиви за потребе рада над датотекама
    - додатна значења појма датотечки систем
      - уређај или партиција где су ускладиштени подаци као датотеке
      - врста датотечног система

# Увод

- Оперативни систем

(Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.; Linux Kernel Documentation, n.d.; BTRFS Documentation, n.d.; Microsoft, 2025)

- датотечки систем

- примери врста датотечких система

- *FFS (Fast File System)*
      - *ext (Extended Filesystem)*
      - *ext2*
      - *ext3*
      - *ext4*
      - *btrfs*
        - употреба *B*-стабала за складиштење метаподатака
      - *FAT (File Allocation Table)*
      - *HPFS (High Performance File System)*
      - *NTFS (NT File System)*



# Увод

- Увод

(Linux Kernel Archives, n.d.; Austin Common Standards Revision Group, 2020; Austin Common Standards Revision Group, n.d.)

- рад над подацима представљен на примерима оперативних система фамилија *Unix* и *Linux*
  - фамилија оперативних система *Linux*
    - оперативни систем *Unix* као узор
    - усмереност на компатибилност са стандардом *POSIX* (*Portable Operating System Interface*)
      - стандардизација оперативних система заснованих на оперативном систему *Unix*
      - основни циљ подршка развоју апликација које су портабилне на нивоу изворног кода

# Садржај

- Увод
- **Простор на екстерном меморијском уређају**
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- Накнадна разматрања
- Ресурси

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Wirzenius et al., n.d.)
    - форматирање (форматирање ниског нивоа)
      - основна припрема уређаја
        - случај магнетних уређаја
          - постављање ознака за стазе и секторе на магнетни медијум
          - провера присуства неисправних сектора
            - често на нивоу уређаја аутоматски долази до избацавања неисправних сектора из употребе и њихове замене исправним секторима
      - обично већ изведено у фабрици производње уређаја

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Wirzenius et al., n.d.)
    - партиционисање
      - поступак поделе простора уређаја на партиције
      - партиција представља део простора уређаја који даље може бити засебно уређиван
        - на различитим партицијама истог уређаја по потреби могу бити креирани различити датотечки системи, па даље постављани и различити оперативни системи
      - партиције могу бити креиране, брисане и мењане у погледу величине
        - помоћу посебних софтверских алата могуће је манипулисати партицијама уређаја

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Wirzenius et al., n.d.)
    - партиционисање
      - табела партиција је структура која садржи основне податке о партицијама
        - примери података о партицији
          - обухваћени распон из простора уређаја
          - врста партиције у односу на оперативни систем и намену
        - контекст постојања табеле партиција
          - на нивоу уређаја
          - могуће и на нивоу појединачне партиције
            - случај проширене партиције, што је партиција која даље може бити партиционисана на логичке партиције

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Wirzenius et al., n.d.)
    - партиционисање
      - табела партиција уређаја традиционално се налази у главном покретачком сектору (енгл. *master boot record*, *MBR*), који је и први сектор уређаја
        - покретачки сектори партиција су први сектори појединачних партиција уређаја
      - при покретању рачунарског система, основни улазно-излазни систем (енгл. *basic input-output system*, *BIOS*) користи садржај главног покретачког сектора
        - очекивано долази до учитавања табеле партиција уређаја и, на основу тога, учитавања покретачког сектора активне партиције (очекивано оне с постављеним оперативним системом)

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.)
    - креирање датотечног система (форматирање високог нивоа)
      - помоћу посебних софтверских алата могуће је за дату партицију ЕМУ креирати нови (празни) датотечки систем одређене врсте
        - разна подешавања могу бити доступна, зависно од врсте датотечног система
          - нпр. величина блока
    - на ЕМУ долази до формирања структура података које су потребне за нови датотечки систем

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Mogin, 2008)
    - заузимање и ослобађање простора за податке
      - расподела простора за коришћење
        - потребно је разликовати који је део простора искоришћен за податке (заузет) а који је део неискоришћен (слободан)
          - иницијално је простор у целости слободан
        - део слободног простора може бити заузет за потребе смештања података датотеке
        - када део заузетог простора више није потребан за одговарајућу датотеку, тај део може бити ослобођен
        - гранула представља јединицу простора за заузимање или ослобађање
          - гранула по капацитету може одговарати позитивном целобројном умношку капацитета сектора, блока, стазе или цилиндра



# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Mogin, 2008)
    - заузимање и ослобађање простора за податке
      - радње које се тичу простора датотеке
        - додела простора за нову датотеку
        - проширивање простора постојеће датотеке
        - ослобађање простора постојеће датотеке
          - делимично или потпуно ослобађање
      - традиционални приступ је да датотеци буду додељене блиске грануле

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Mogin, 2008)
    - заузимање и ослобађање простора за податке
      - потребно је одржавати податке о заузећу простора
      - примери механизма за евидентирање заузећа простора (случај блока као грануле)
        - листа слободних блокова
          - слободни блокови су спрегнути путем показивача у листу
          - једноставан механизам али потенцијално неефикасан
            - потребно пролажење кроз листу ради проналажења слободног блока који је близак блоковима датотеке
        - мапа статуса блокова
          - мапа садржи за сваки блок по једну вредност индикатора заузећа (заузет или слободан), нпр. један бит по блоку
          - мапа је обично релативно мале величине и може бити доступна у ОМ

# Простор на екстерном меморијском уређају

- Управљање простором на ЕМУ с директним приступом
  - карактеристичне активности (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - дефрагментација
      - прераспоређивање података на ЕМУ ради бољих перформанси у раду

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - примери концепата из једноставнијих датотечких система
    - блок
    - област

(Arpaci-Dusseau & Arpaci-Dusseau, 2023)

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - блок (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - простор је подељен у блокове
      - блокови могу бити исте величине (нпр. 4 KB)
    - блок обично представља јединицу алокације (доделе)
      - нпр. један или више блокова може бити алоцирано (додељено) за чување садржине неке датотеке



*поједностављена представа организације простора на ЕМУ*

# Простор на екстерном меморијском уређају

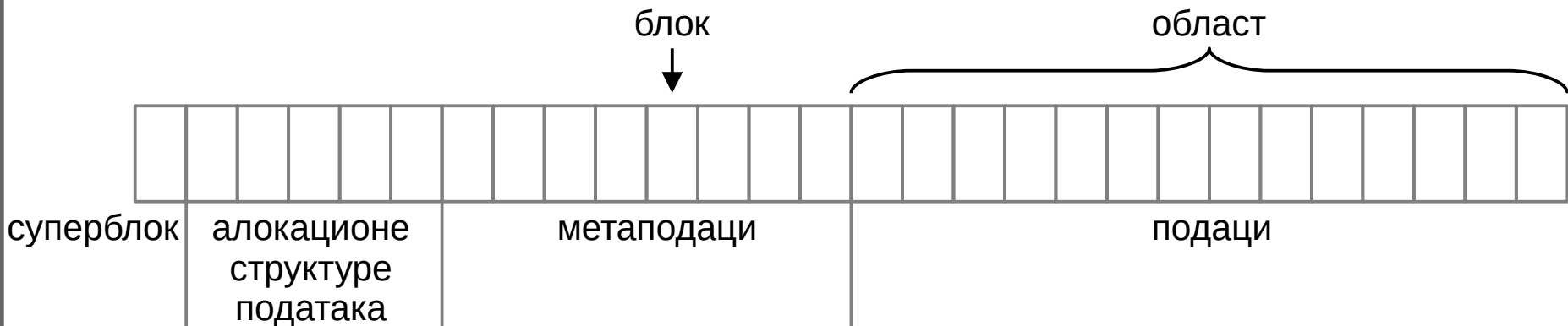
- Основни концепти управљања простором на ЕМУ

- област (Arpaci-Dusseau & Arpaci-Dusseau, 2023)

- блокови су организовани у области зависно од тога какве податке треба да садрже

- примери врста области

- област података (корисничких података)
        - област метаподатака
        - област алокационих структура података
        - суперблок



*поједностављена представа организације простора на ЕМУ*

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – област података (корисничких података) (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - намењена смештању корисничких података, који представљају садржине датотека
    - искоришћени блокови представљају блокове података који су алоцирани за конкретне датотеке

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – област метаподатака (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - намењена смештању података о датотекама (датотечких метаподатака)
      - за појединачну датотеку могу бити евидентирани разноврсни подаци
        - нпр. величина, дозволе, временске одреднице, евиденција додељених блокова података
    - организована као табела системских табела датотека
      - енгл. *inode table*
      - традиционално као низ системских табела датотека



# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – област метаподатака (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Mogin, 2008)
    - **системска табела датотеке**
      - позната као чвор индекса или *i*-чвор (енгл. *index node* или *inode*)
      - структура која садржи основне податке о датотеци
        - нпр. врста, величина, карактеристичне временске одреднице, подаци у вези с блоковима података датотеке
      - структура која традиционално садржи алокациону табелу датотеке
        - алокациона табела датотеке је намењена чувању адреса почетака гранула додељених датотеци
      - **бројчана ознака системске табеле датотеке**
        - позната као број чвора индекса, број *i*-чвора или *i*-број (енгл. *inode number* или *i-number*)
        - број који традиционално представља индекс одговарајућег елемента у низу системских табела датотека

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – област алокационих структура (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - намењена смештању података о заузећу простора
      - потребно пратити стање блокова у областима података и метаподатака у погледу алокације
        - да ли је блок слободан или алоциран за неку намену

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – област алокационих структура (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - примери метода за праћење заузећа простора
      - употреба листе слободног простора
        - слободни блокови су увезани у листу путем показивача
        - глава листе одговара првом слободном блоку
      - употреба бит мапе
        - бит мапа је структура састављена од битова
        - појединачни битови односе се на појединачне блокове
          - вредност 0 – ознака да је одговарајући блок слободан
          - вредност 1 – ознака да је одговарајући блок алоциран
        - постојање засебних бит мапа за област података и табелу системских табела датотека

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - област – суперблок (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - намењен смештању података о датотечком систему
      - нпр. ознака врсте датотечног система, број системских табела датотека, место почетка неке области

# Простор на екстерном меморијском уређају

- Основни концепти управљања простором на ЕМУ
  - датотечки системи могу се драстично разликовати у погледу општег начина распоређивања података и врсте коришћених структура података на ЕМУ (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
    - примери разлика
      - разлике у структури системске табеле датотеке
      - разлике у начину евидентирања распореда података датотеке
      - разлике у начину праћења општег заузећа простора
      - разлике у погледу блока и области

# Простор на екстерном меморијском уређају

- Поједностављени пример датотечног система првобитног оперативног система *Unix* (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - дело Кена Томпсона
  - подршка за датотеке и хијерархију директоријума
  - структуре података на ЕМУ
    - главне области
      - **суперблок**
      - **област системских табела датотека**
      - **област података**

# Простор на екстерном меморијском уређају

- Поједностављени пример датотечног система првобитног оперативног система *Unix* (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - структуре података на ЕМУ
    - суперблок
      - део намењен чувању општих података о датотечком систему
        - нпр. величина простора, број системских табела датотеке, подаци у вези с евиденцијом слободног простора
      - очекивано најмања област

# Простор на екстерном меморијском уређају

- Поједностављени пример датотечног система првобитног оперативног система *Unix* (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Mogin, 2008; Wirzenius et al., n.d.)
  - структуре података на ЕМУ
    - **област системских табела датотека**
      - део намењен чувању системских табела датотека
      - системска табела датотеке
        - позната као чвор индекса или *i*-чвор (енгл. *index node* или *inode*)
        - садржани основни подаци о датотеци
          - назив датотеке се чува у одговарајућем директоријуму
        - евидентирани су и бројеви који представљају блокове података што су алоцирани за датотеку
          - традиционално је могуће чувати само одређени број бројева, а чување додатних бројева могуће је путем блокова индирекције (динамички алоцирани блокови за складиштење додатних бројева)



# Простор на екстерном меморијском уређају

- Поједностављени пример датотечног система првобитног оперативног система *Unix* (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - структуре података на ЕМУ
    - **област података**
      - део намењен чувању садржина датотека
      - очекивано највећа област

# Простор на екстерном меморијском уређају

- Фрагментација (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - подаци који се тичу једне датотеке често су расути по простору ЕМУ
    - радње над таквом датотеком могу дуже трајати у случају употребе магнетног диска
      - системска табела датотеке може бити удаљена од делова датотеке
      - делови једне те исте датотеке могу бити међусобно удаљени

# Простор на екстерном меморијском уређају

- Фрагментација (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - делови слободног простора често су расути по простору ЕМУ
    - проширење постојеће датотеке може довести до заузимања слободног сегмента који је удаљен од постојећих података који се тичу те датотеке
    - иако постоји потребан број сегмената слободног простора, може бити неизводиво пронаћи толики број делова слободног простора који су нанизани један за другим

# Простор на екстерном меморијском уређају

- Фрагментација (Van der Meer et al., 2021)
  - врсте фрагментације у датотечком систему
    - фрагментација датотеке
      - подаци датотеке нису нанизани један за другим
      - до фрагментације датотеке може доћи при креирању нових датотека и проширивању постојећих
    - фрагментација слободног простора
      - слободни делови простора нису нанизани један за другим
      - до фрагментације слободног простора може доћи при брисању и смањењу датотека

# Простор на екстерном меморијском уређају

- Фрагментација (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.)
  - дефрагментација
    - поступак прераспоређивања података на ЕМУ како би подаци датотека били постављени узастопно, као и делови слободног простора
    - очекивано долази до померања података датотека и ажурирања метаподатака датотека и алокационих структура
    - разни датотечки системи су већ тако направљени да да фрагментације датотека буде што мање

# Простор на екстерном меморијском уређају

- Фрагментација (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.)
  - интерна фрагментација
    - појава да алоцирани блок није у потпуности искоришћен
      - количина података коју треба чувати може бити мања од величине блока, али није могуће алоцирати део блока
    - у случају великог броја веома малих датотека (појединачне величине мање од величине блока), много расположивог простора може остати недоступно
    - што је мањи блок, мања је и интерна фрагментација али је и мања ефикасност која се тиче преноса података с диска
    - што је већи блок, већа је и ефикасност која се тиче преноса података с диска, али је већа и интерна фрагментација

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- **Каталог**
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- Накнадна разматрања
- Ресурси

# Каталог

- Каталог (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - каталог је евиденција датотека
    - познат и као директоријум
  - каталог је организован као структура типа стабло
    - хијерархија директоријума (стабло директоријума)



# Каталог

- Хијерархија директоријума (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - коренски чвор представља главни директоријум
    - коренски директоријум
  - некоренски чворови представљају поддиректоријуме
  - директоријум (главни директоријум или поддиректоријум) обухвата низ назива својих поддиректоријума и датотека
    - показивачи су придружени називима
      - ако назив одговара поддиректоријуму, показивач садржи адресу тог директоријума на следећем нивоу у стаблу
      - ако назив одговара датотеци, показивач садржи адресу системске табеле датотеке за ту датотеку
  - лисни чвор је или празан или садржи називе датотека с одговарајућим показивачима

# Каталог

- Хијерархија директоријума (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - врсте назива
    - **кориснички назив**
      - **назив**
      - **пуни назив**
    - **интерни назив (назив ниског нивоа)**

- Хијерархија директоријума – називи (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - кориснички назив – назив
    - представља назив намењен корисницима при обичном раду над подацима
      - нпр. `proslava.png`
    - назив датотеке обично има два дела раздвојена посебним знаком .
      - основни назив (почетни део назива), који представља назив у ужем смислу
        - нпр. `proslava`
      - екстензија (крајњи део назива), који очекивано указује на врсту садржаја датотеке
        - нпр. `png`

# Каталог

- Хијерархија директоријума – називи (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - кориснички назив – пуни назив
    - у односу на назив, пуни назив указује и на место у хијерархији
    - путања од коренског до посматраног чвора, при чему су називи на путањи раздвојени посебним знаком /
      - нпр. `/multimedija/slike/proslava.png`
      - коренски директоријум представљен знаком /
    - путања
      - апсолутна путања – комплетна путања, која започиње у коренском директоријуму
        - нпр. `/multimedija/slike/proslava.png`
      - релативна путања – путања дата у односу на текући директоријум
        - нпр. `slike/proslava.png` (за текући директоријум `/multimedija`)

# Каталог

- Хијерархија директоријума – називи (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - интерни назив (назив ниског нивоа)
    - бројчана ознака одговарајуће системске табеле датотеке
      - позната као број индекса чвора, број чвора или *i*-број
    - интерни назив је јединствен у границама датотечког система
    - различитим називима може бити придружен исти интерни назив
    - пример
      - у многим оперативним системима фамилије *Unix*, *i*-број за коренски директоријум је 2

# Каталог

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023; Wirzenius et al., n.d.)
  - увезивање (постављање) датотечког система
    - енгл. *mount*
    - датотечки систем који је већ креиран на одређеној партицији бива увезан у хијерархију директоријума на дату путању
      - оперативни систем иницијално користи податке из одговарајућег суперблока
      - коренски директоријум датотечког систем бива повезан с датом путањом
      - нови датотечки систем бива придодат хијерархији директоријума као ново подстабло
    - постоји и могућност одвезивања увезаног датотечког система из хијерархије директоријума (енгл. *unmount*)
    - путем једне хијерархије директоријума могуће је једнообразно користити више датотечких система разних врста

# Каталог

- Хијерархија директоријума (man-pages, n.d.)
  - врсте датотека (интерна класификација)
    - обичне датотеке
    - директоријуми
    - симболичке везе
    - именоване преносне структуре (*FIFO*)
    - прикључци домена *UNIX*
    - блоковски уређаји
      - нпр. магнетни дискови
    - знаковни уређаји
      - нпр. траке
    - датотеке непознате врсте

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - директоријум
    - директоријум представља колекцију елемената
    - директоријум је датотека посебне врсте
      - директоријум има одговарајући  $i$ -број
      - ажурирање директоријума као датотеке корисник у основи изводи посредно, јер је датотечки систем задужен за одржавање исправности података у вези с директоријумом
    - посебне ставке унутар директоријума
      - . (представа самог директоријума)
      - . . (представа надређеног директоријума)



# Каталог

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - везе
    - веза (енгл. *link*)
      - механизам за вишеструко увезивање исте датотеке унутар хијерархије директоријума
    - врсте веза
      - јаке везе
      - симболичке (слабе) везе

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - везе – јака веза
    - јака веза је назив у хијерархији директоријума који се односи на исту датотеку као неки други назив
      - оба назива могу бити равноправно коришћена у раду
    - више назива може се односити на исту датотеку
      - више назива бива повезано с истим интерним називом
      - аутоматски бива праћен број веза по датотеци
        - уклањањем једног назива за датотеку долази до смањења одговарајућег броја веза за један
        - смањењем броја веза на нулу долази до уклањања саме датотеке
    - ограничења јаке везе
      - јака веза не може се односити на директоријум и не може се односити на датотеку која је на другој партицији

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - везе – симболичка (слаба) веза
    - симболичка веза је датотека чији је очекивани садржај путања до неке датотеке
      - повезана датотека је датотека чију путању садржи симболичка веза
      - симболичка веза очекивано треба да буде додатна приступна тачка за повезану датотеку
    - симболичка веза је датотека посебне врсте
      - симболичка веза као датотека има сопствени назив
    - може доћи до прекида везе
      - енгл. *dangling reference*
      - путања која је сачувана у симболичкој вези не одговара никаквој датотеци
        - нпр. повезана датотека је преименована или обрисана

# Каталог

- Хијерархија директоријума (man-pages, n.d.)
  - именоване преносне структуре (*FIFO*)
    - преносна структура (енгл. *pipe*)
      - једносмерни канал за комуникацију између процеса
      - две компоненте обуваћене
        - компонента из које се чита
        - компонента у коју се пише

# Каталог

- Хијерархија директоријума (man-pages, n.d.)
  - прикључци домена *UNIX*
    - прикључак домена *UNIX* (енгл. *UNIX domain socket*)
      - структура намењена ефикасној комуникацији између процеса на истој машини

# Каталог

- Хијерархија директоријума (man-pages, n.d.; Wirzenius et al., n.d.)
  - уређаји (датотеке уређаја)
    - датотеке уређаја као представе физичких и виртуалних уређаја
      - читање из датотеке уређаја подразумева пријем података
      - писање у датотеку уређаја подразумева слање података
      - углавном се не ради непосредно над датотекама уређаја већ путем посебних програма
    - датотеке уређаја очекивано у директоријуму `/dev`
      - за партиције ЕМУ очекивано постоје засебне датотеке уређаја
    - врсте уређаја
      - блоковски уређаји
        - нпр. магнетни дискови, оптички дискови
      - знаковни уређаји
        - нпр. траке, џојстици

# Каталог

- Хијерархија директоријума (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - дозволе (пермисије) рада над датотекама
    - две карактеристичне врсте механизма за спецификацију дозвољених радњи над датотекама
      - **механизам битова дозволе**
      - **механизам листе управљања приступом**

# Каталог

- Хијерархија директоријума – дозволе (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - механизам битова дозволе
    - спецификација дозвољених радњи по корисничким нивоима
      - могуће радње
        - читање (r), писање (w), извршавање (x)
      - могући кориснички нивои
        - власник, група, остали



- Хијерархија директоријума – дозволе (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - механизам битова дозволе
    - текстуална представа дозволе
      - знаковна секвенца од девет елемената која је организована кроз три подсеквенце од по три елемента
        - три подсеквенце
          - 1. подсеквенца односи се на власника датотеке
          - 2. подсеквенца односи се на групу
          - 3. подсеквенца односи се на остале
        - три елемента једне подсеквенце
          - 1. елемент је за дозволу читања: r (има) или - (нема)
          - 2. елемент је за дозволу писања: w (има) или - (нема)
          - 3. елемент је за дозволу извршавања: x (има) или - (нема)

# Каталог

- Хијерархија директоријума – дозволе
  - механизам битова дозволе
    - текстуална представа дозволе
      - пример
        - `rwxr-xr-x`
          - власник има дозволе читања, писања и извршавања
          - група има дозволе читања и извршавања
          - остали имају дозволе читања и извршавања

- Хијерархија директоријума – дозволе (Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - механизам битова дозволе
    - бројчана представа дозволе
      - бинарна представа дозволе
        - за дату текстуалну представу ознаке  $r$ ,  $w$  и  $x$  могу бити замењене са 1 а ознака - са 0
        - бинарна представа може бити посматрана као основна
      - октална представа дозволе
        - за дату бинарну представу могу бити уочене три узастопне трочлане бинарне секвенце и свака од њих може бити замењена с једном окталном вредношћу

# Каталог

- Хијерархија директоријума – дозволе
  - механизам битова дозволе
    - пример
      - текстуална представа
        - `rwXr-xr-x`
      - бинарна представа
        - `111101101`
      - октална представа
        - `755`

- Хијерархија директоријума – дозволе (Arpaci-Dusseau & Arpaci-Dusseau, 2023; OpenAFS, n.d.-a)
  - механизам листе управљања приступом
    - листа управљања приступом (енгл. *access control list*)
      - унутар посебне листе могуће је специфицирати дозволе које имају корисници
    - подржан у неким датотечким системима
      - нпр. дистрибуирани датотечки систем *Andrew File System (AFS)*
        - *OpenAFS* је имплементација која је отвореног изворног кода

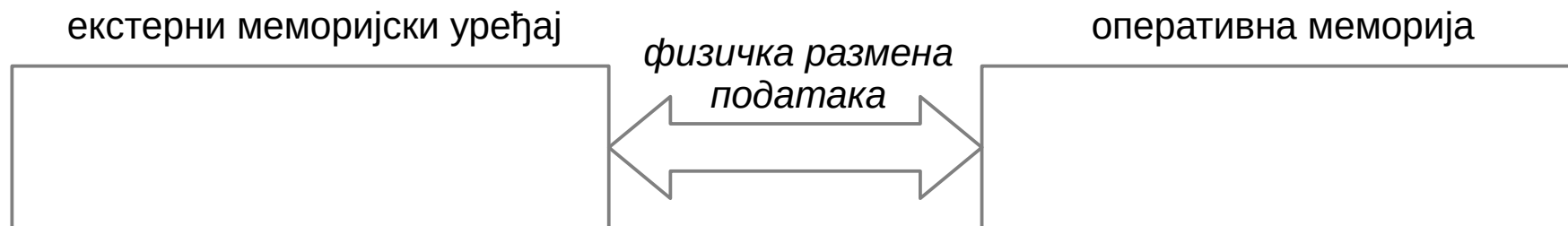
- Хијерархија директоријума – дозволе (Arpaci-Dusseau & Arpaci-Dusseau, 2023; OpenAFS, n.d.-b)
  - механизам листе управљања приступом – *OpenAFS*
    - датотеке немају листе управљања приступом
    - сваки директоријум има листу управљања приступом
      - могуће је специфицирати дозволе по корисницима и групама
        - основне дозволе на нивоу директоријума
          - преглед директоријума (l)
          - додавање датотека и поддиректоријума (i)
          - брисање датотека и поддиректоријума (d)
          - администрирање листе (a)
        - основне дозволе на нивоу садржаних датотека
          - читање садржаја датотека (r)
          - измена садржаја датотека (w)
          - закључавање датотека (k)

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- **Физичка размена података**
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- Накнадна разматрања
- Ресурси

# Физичка размена података

- Припрема и управљање физичком разменом података између ОМ и ЕМУ (Mogin, 2008)
  - посебне рутине оперативног система управљају физичком разменом података
    - по добијању налога за извршење неког системског позива који се тиче преноса података између ОМ и ЕМУ, оперативни систем преузима контролу над током размене података
      - размена може наступити непосредно по добијању налога за извршење системског позива или тек након неког времена од тога





# Физичка размена података

- Припрема и управљање физичком разменом података између ОМ и ЕМУ (Mogin, 2008; Arpacı-Dusseau & Arpacı-Dusseau, 2023)
  - посебне рутине оперативног система управљају физичком разменом података
    - размена података тече између бафера у ОМ и датотеке на ЕМУ
      - на страни ОМ постоји део меморијског простора за привремени смештај података – бафер
      - на страни ЕМУ очекивано постоји меморијски простор за привремени смештај података – кеш
    - параметри размене података
      - врста операције (читање или писање података у односу на ЕМУ)
      - адреса бафера
      - капацитет бафера
      - редни број блока на ЕМУ

# Физичка размена података

- Бафери (Mogin, 2008; Wirzenius et al., n.d.)
  - бафер је део ОМ који служи за привремено смештање података приликом улазно-излазних операција
  - употреба бафера очекивано побољшава перформансе рада
    - читање података с ЕМУ
      - исти подаци могу бити потребни више пута у релативно краћем периоду
        - захтевани подаци бивају прочитани с ЕМУ и смештени у бафер
        - накнадна захтевања истих података подразумеваће читање из бафера, под условом да су подаци и даље доступни у баферу
    - писање података на ЕМУ
      - подаци за писање бивају смештени у бафер и из бафера могу бити накнадно уписани на ЕМУ подаци који су управо уписани могу бити убрзо потребни за читање

# Физичка размена података

- Бафери (Mogin, 2008)
  - врсте бафера према нивоу на којем се налазе
    - системски бафери
    - бафери стандардних библиотечких функција
    - бафери корисничког програма

# Физичка размена података

- Бафери (Mogin, 2008)
  - системски бафери
    - у надлежности оперативног система
      - нису непосредно доступни стандардним библиотечким функцијама и наредбама корисничких програма
    - зона бафера је део ОМ намењен за системске бафере
    - капацитет системског бафера је очекивано исти као капацитет блока на ЕМУ
    - читање података с ЕМУ
      - за случај системског позива за читање података, подаци за читање треба да буду пренети с уређаја у системски бафер
    - писање података на ЕМУ
      - за случај системског позива за писање података, подаци за упис треба да буду у системском баферу пре преноса на уређај

# Физичка размена података

- Бафери (Mogin, 2008)
  - системски бафери – управљање баферима
    - примери поступака за управљање баферима
      - поступак фиксне доделе бафера
      - поступак удруживања бафера

# Физичка размена података

- Бафери (Mogin, 2008)
  - системски бафери – управљање баферима
    - поступак фиксне доделе бафера
      - датотеци при отварању бива додељен бар један бафер
        - у случају датотеке отворене и за читање и за писање, бивају додељена бар два бафера
          - један бафер за читање а други за писање
        - бафери остају додељени датотеци до њеног затварања
      - једноставан поступак али потенцијално неефикасан у погледу употребе меморије
        - неефикасност у случају слабог коришћења датотеке, јер су бафери резервисани али слабо коришћени

# Физичка размена података

- Бафери (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - системски бафери – управљање баферима
    - поступак удруживања бафера (енгл. *buffer pooling*)
      - датотеци бафери бивају додељени по потреби
        - ако при раду над датотеком неопходни подаци нису доступни у неком од бафера, из зоне бафера бива изабран одређени бафер, који даље бива додељен датотеци за потребе конкретне радње
        - избор бафера за доделу може бити по принципу најдуже некоришћености (енгл. *least recently used*)
          - бива изабан бафер чији је садржај најдуже некоришћен
          - скорашњост коришћења бафера може бити евидентирана путем реда
            - почетак реда одговара најдуже некоришћеном баферу
            - крај реда одговара најскорије коришћеном баферу

# Физичка размена података

- Бафери (Mogin, 2008)
  - бафери стандардних библиотечких функција
    - коришћени у раду стандардних библиотечких функција
      - нпр. функције из заглавља *stdio.h* у програмском језику C
    - извршавање стандардних библиотечких функција може обухватити употребу интерних бафера
      - у раду стандардних библиотечких функција може доћи до давања налога за извршење неког системског позива који се тиче преноса података између ОМ и ЕМУ
      - при извршењу системског позива који се тиче преноса података између ОМ и ЕМУ може доћи до преноса података између системског бафера и интерног бафера (бафера стандардне библиотечке функције)
        - код читања података – смер од системског до интерног бафера
        - код писања података – смер од интерног до системског бафера

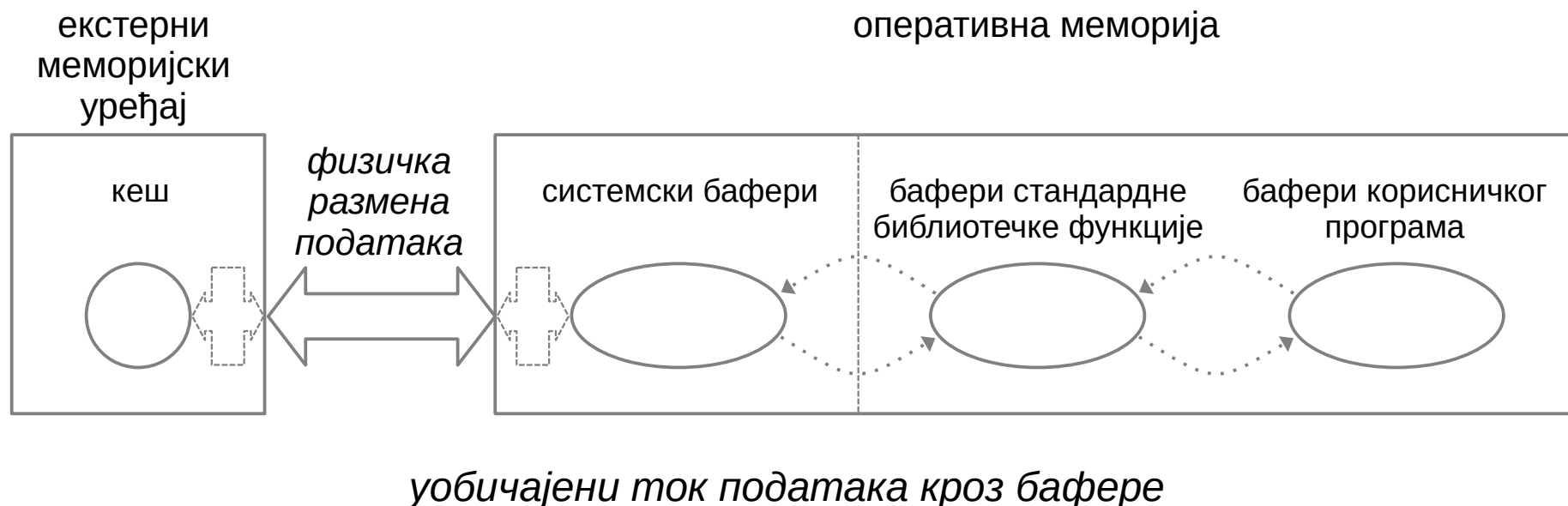


# Физичка размена података

- Бафери (Mogin, 2008)
  - бафери корисничког програма
    - декларисани у корисничком програму
      - рад над баферима корисничког програма изводи се унутар тог корисничког програма
    - извршавање корисничког програма може довести до преноса података између бафера корисничког програма и других бафера
      - у случају коришћења стандардних библиотечких функција, може доћи до преноса података између бафера корисничког програма и бафера стандардних библиотечких функција, па и између бафера стандардних библиотечких функција и системских бафера
      - у случају давања налога за извршење системских позива, може доћи до преноса података између бафера корисничког програма и системских бафера

# Физичка размена података

- Бафери (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - перформансе
    - употреба бафера стандардне библиотечке функције или бафера корисничког програма може резултовати смањењем броја стварних преноса података између ОМ и ЕМУ



# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- **Спрега између програма и датотека**
- Системски позиви
- Методе приступа
- Накнадна разматрања
- Ресурси

# Спрега између програма и датотека

- Остваривање спреге између програма и датотека (Mogin, 2008)
  - у раду над датотекама потребно је чувати податке о самим датотекама, као и о њиховој употреби
    - нпр. како су размештени подаци датотеке, које су датотеке тренутно у употреби, које све датотеке тренутно користи неки програм
  - оперативни систем успоставља и одржава разне табеле
    - табеле представљају евиденције у вези с подацима и активностима

# Спрега између програма и датотека

- Остваривање спреге између програма и датотека (Mogin, 2008)
  - примери табела које се тичу остваривања спреге између програма и датотека
    - **системска табела датотеке**
    - **табела описа датотеке**
    - **табела отворених датотека**
    - **табела логичких имена датотека**

# Спрега између програма и датотека

- Остваривање спреге између програма и датотека (Mogin, 2008)
  - системска табела датотеке
    - садржи основне податке о датотеци и алокациону табелу датотеке
    - трајно смештена на ЕМУ
      - системска табела датотеке настаје при креирању датотеке
      - ради употребе датотеке садржај одговарајуће системске табеле датотеке са ЕМУ бива учитан у ОМ а евентуалне измене до којих дође при коришћењу очекивано бивају уписане на ЕМУ

# Спрега између програма и датотека

- Остваривање спреге између програма и датотека (Mogin, 2008)
  - табела описа датотеке
    - садржи податке о конкретној датотеци у употреби
      - нпр. дозволе, величина, показивачи према деловима простора датотеке на ЕМУ
      - потребни подаци бивају учитани из одговарајуће системске табеле датотеке
    - смештена у ОМ у делу за оперативни систем
      - табела описа датотеке бива попуњена при отварању датотеке

# Спрега између програма и датотека

- Остваривање спреге између програма и датотека (Mogin, 2008; man-pages, n.d.)
  - табела отворених датотека
    - садржи податке о свим датотекама у употреби на нивоу система
      - за сваку отворену датотеку као унос у табели садржана је дескрипција отворене датотеке
        - дескрипција отворене датотеке садржи одређене податке који се тичу употребе отворене датотеке
          - нпр. вредност текућег показивача, режим приступа, показивач према одговарајућој табели описа датотеке
    - смештена у ОМ у делу за оперативни систем



# Спрега између програма и датотека

(Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023; man-pages, n.d.)

- Остваривање спреге између програма и датотека
  - табела логичких имена датотека
    - позната и као табела дескриптора датотека
    - садржи податке о датотекама које су у употреби на нивоу конкретног програма (процеса)
      - за сваку датотеку у употреби на нивоу процеса присутни су подаци
        - логичко име датотеке (дескриптор датотеке)
          - цео број који на нивоу процеса означава датотеку у употреби
          - представља индекс уноса у табели
        - показивач ка одговарајућем уносу у табели отворених датотека
      - смештена у ОМ у делу за конкретни процес
        - табела логичких имена датотека формирана и делимично попуњена од стране компајлера, а ажурирана од стране оперативног система при отварању датотеке у оквиру процеса

# Спрега између програма и датотека

(Arpaci-Dusseau & Arpaci-Dusseau, 2023; man-pages, n.d.)

- Остваривање спреге између програма и датотека
  - табела логичких имена датотека
    - посебни дескриптори датотека
      - 0 – *stdin* (ток улаза)
      - 1 – *stdout* (ток излаза)
      - 2 – *stderr* (ток грешке)

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- **Системски позиви**
- Методе приступа
- Накнадна разматрања
- Ресурси

# Системски позиви

- Системски позив (Mogin, 2008)
  - кориснички програм путем системског позива захтева одговарајућу услугу од оперативног система
  - у извршавању системског позива долази до преноса управљања радом централног процесора с корисничког програма на оперативни систем

# Системски позиви

- Системски позиви (man-pages, n.d.)
  - употреба системских позива у савременим оперативним системима фамилије *Linux*
    - системски позиви обично не бивају непосредно активирани, већ путем посредничких (омотачких) функција које су доступне у одговарајућој библиотеци
      - нпр. посредничке функције у стандардној библиотеци за програмски језик C (пакет *GNU C library – glibc*)
    - посредничке функције за системске позиве
      - назив често исти као код одговарајућег системског позива
      - имплементација често обухвата само припрему аргумената, налог за извршење системског позива и уређивање повратне информације у вези с евентуалном грешком
        - нпр. променљива *errno* целобројног типа из заглавља *errno.h* користи се за евидентирање појаве грешке

# Системски позиви

- Системски позиви
  - системски позиви за рад над датотекама
    - представљени на примеру савремених оперативних система из фамилије *Linux* и посредничких функција из стандардне библиотеке за програмски језик *C*
      - кернел *Linux 6.8*
      - стандард *POSIX.1-2008*

# Системски позиви

- Системски позиви (man-pages, n.d.)
  - примери посредничких функција за рад над датотекама
    - `open` – отварање датотеке
    - `creat` – креирање датотеке
    - `close` – затварање датотеке
    - `lseek` – позиционирање у датотеци
    - `read` – читање из датотеке
    - `write` – писање у датотеку
    - `fsync` – снимање измена у вези с датотеком
    - `unlink` – брисање назива датотеке
    - `truncate` – димензионисање датотеке
    - `stat` – добављање информација о датотеци

# Системски позиви

- Системски позиви – `open` (man-pages, n.d.)
  - отварање датотеке
    - уз могућност креирања датотеке
  - функција `int open( . . . )` из заглавља *fcntl.h*
    - стандардна библиотека за програмски језик C



# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - параметри
      - `pathname` – путања
      - `flag` – флегови за специфицирање појединости у вези с режимом приступа датотеци, поступком отварања (и евентуално креирања) и накнадних УИ операција
        - потребно је задати режим приступа
      - `mode` – спецификација режима датотеке
        - у случају креирања датотеке

# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - параметри – флегови
      - режими приступа датотеци
        - O\_RDONLY – могуће само читање
        - O\_WRONLY – могуће само писање
        - O\_RDWR – могуће и читање и писање

# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - параметри – флегови
      - примери осталих флегова
        - O\_APPEND – отварање у режиму додавања
          - писање у датотеку се одвија писањем садржаја на крај датотеке
        - O\_CREAT – креирање регуларне датотеке, ако задата путања не постоји
        - O\_TRUNC – скраћење датотеке на нулту дужину (у случају постојеће регуларне датотеке и примене неког од режима приступа који дозвољавају писање)

# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - параметри – режими датотеке
      - подешавање дозвољених радњи над датотеком по корисничким нивоима
        - радње
          - читање
          - писање
          - извршавање
        - кориснички нивои
          - корисник (власник датотеке)
          - група
          - остали

# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - повратна вредност
      - успешно извршење
        - нови дескриптор датотеке
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – open (man-pages, n.d.)
  - `int open(const char * pathname, int flags, ... /* mode_t mode */);`
    - повратна вредност – дескриптор датотеке
      - дескриптор датотеке је ненегативан цео број који представља индекс уноса у процесној табели дескриптора отворених датотека
      - дескриптор датотеке служи као веза према дескрипцији отворене датотеке
      - дескрипција отворене датотеке је унос у системској табели отворених датотека
        - дескрипција отворене датотеке очекивано настаје при позиву `open()`
      - дескриптор датотеке служи у разним другим системским позивима као ознака датотеке која је предмет активности

# Системски позиви

- Системски позиви – `creat` (man-pages, n.d.)
  - креирање датотеке
    - функција `int creat(...)` из заглавља *fcntl.h*
      - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `creat` (man-pages, n.d.)
  - `int creat(const char * pathname, mode_t mode);`
    - параметри
      - `pathname` – путања
      - `mode` – спецификација режима датотеке
    - повратна вредност
      - успешно извршење
        - нови дескриптор датотеке
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)
    - позив `creat()` има еквивалентан ефекат као позив `open()` с одређеним подешавањима флегова
      - `open(pathname, O_CREAT|O_WRONLY|O_TRUNC, mode);`



# Системски позиви

- Системски позиви – `close` (man-pages, n.d.)
  - затварање дескриптора датотеке
    - ако је задати дескриптор датотеке уједно и последњи за своју дескрипцију отворене датотеке, ресурси који се тичу те дескрипције отворене датотеке бивају ослобођени
  - функција `int close(...)` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `close` (man-pages, n.d.)
  - `int close(int fd);`
    - параметри
      - `fd` – дескриптор датотеке
    - повратна вредност
      - успешно извршење
        - 0
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `lseek` (man-pages, n.d.; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - позиционирање у датотеци
    - постављање помака у датотеци у односу на референтну тачку
    - извршење позиционирања само по себи не обухвата извршење физичког позиционирања на ЕМУ
      - тражена позиција треба да буде евидентирана у одговарајућој дескрипцији отворене датотеке
  - функција `off_t lseek( . . . )` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `lseek` (man-pages, n.d.)
  - `off_t lseek(int fd, off_t offset, int whence);`
    - параметри
      - `fd` – дескриптор датотеке
      - `offset` – помак (у бајтовима)
      - `whence` – референта тачка
        - основни скуп референтних тачака
          - `SEEK_SET` – почетак датотеке
          - `SEEK_CUR` – тренутна позиција
          - `SEEK_END` – крај датотеке

# Системски позиви

- Системски позиви – `lseek` (man-pages, n.d.)
  - `off_t lseek(int fd, off_t offset, int whence);`
    - повратна вредност
      - успешно извршење
        - резултујући помак изражен у бајтовима у односу на почетак датотеке
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `read` (man-pages, n.d.)
  - читање из дескриптора датотеке
    - покушај читавања одређеног броја бајтова из дескриптора датотеке и смештање прочитаних бајтова у бафер
      - читање почиње од постојеће позиције у датотеци
  - функција `ssize_t read( . . . )` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – read (man-pages, n.d.)
  - `ssize_t read(int fd, void buf [.count], size_t count);`
    - параметри
      - `fd` – дескриптор датотеке
      - `buf` – почетак бафера
      - `count` – број бајтова за читавање

# Системски позиви

- Системски позиви – read (man-pages, n.d.)
  - `ssize_t read(int fd, void buf [.count], size_t count);`
    - повратна вредност
      - успешно извршење
        - број прочитаних бајтова, који може бити и мањи од задатог броја
          - позиција у датотеци бива повећана за број прочитаних бајтова
    - појава грешке
      - -1 (врста грешке је назначена у променљивој `errno`)



# Системски позиви

- Системски позиви – read (man-pages, n.d.)
  - функција `ssize_t pread( . . . )` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – read (man-pages, n.d.)
  - `ssize_t pread(int fd, void buf [.count], size_t count, off_t offset);`
    - параметри
      - `fd` – дескриптор датотеке
      - `buf` – почетак бафера
      - `count` – број бајтова за читавање
      - `offset` – позиција од које треба да почне читање
        - позиција дата као помак изражен у бајтовима у односу на почетак датотеке

# Системски позиви

- Системски позиви – read (man-pages, n.d.)
  - `ssize_t pread(int fd, void buf [.count], size_t count, off_t offset);`
    - повратна вредност
      - успешно извршење
        - број прочитаних бајтова, који може бити и мањи од задатог броја
          - позиција у датотеци не бива промењена
    - појава грешке
      - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `write` (Mogin, 2008; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - писање у дескриптор датотеке
    - покушај уписивања одређеног броја бајтова из бафера у датотеку на коју се односи дескриптор датотеке
      - писање почиње од постојеће позиције у датотеци
        - ако је датотека отворена у режиму надодавања, пре писања бива позиција постављена на крај датотеке
    - не мора се подразумевати да ће успешно извршавање системског позива непосредно резултовати уписивањем података на ЕМУ
      - на располагању је функција `fsync( . . . )`
  - функција `ssize_t write( . . . )` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `write` (man-pages, n.d.)
  - `ssize_t write(int fd, const void buf [.count], size_t count);`
    - параметри
      - `fd` – дескриптор датотеке
      - `buf` – почетак бафера
      - `count` – број бајтова за уписивање

# Системски позиви

- Системски позиви – `write` (man-pages, n.d.)
  - `ssize_t write(int fd, const void buf [.count], size_t count);`
    - повратна вредност
      - успешно извршење
        - број уписаних бајтова, који може бити и мањи од задатог броја
          - позиција у датотеци бива повећана за број уписаних бајтова
          - нема гаранција да су подаци уписани на ЕМУ
            - може бити искоришћена функција `fsync( . . . )`
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `write` (man-pages, n.d.)
  - функција `ssize_t pwrite(...)` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `write` (man-pages, n.d.)
  - `ssize_t pwrite(int fd, const void buf [.count], size_t count, off_t offset);`
    - параметри
      - `fd` – дескриптор датотеке
      - `buf` – почетак бафера
      - `count` – број бајтова за уписивање
      - `offset` – позиција од које треба да почне писање



# Системски позиви

- Системски позиви – `write` (man-pages, n.d.)
  - `ssize_t pwrite(int fd, const void buf [.count], size_t count, off_t offset);`
    - повратна вредност
      - успешно извршење
        - број уписаних бајтова, који може бити и мањи од задатог броја
          - позиција у датотеци не бива промењена
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `fsync` (man-pages, n.d.; Arpaci-Dusseau & Arpaci-Dusseau, 2023)
  - синхронизација стања датотеке у ОМ и на ЕМУ
    - измењени подаци датотеке на коју се односи дескриптор датотеке треба да буду пренесени из ОМ на ЕМУ
      - пренос података и метаподатака датотеке
    - у неким ситуацијама може бити потребно посебно извршити синхронизацију стања за директоријум у којем се налази датотека
    - функција `int fsync(...)` из заглавља *unistd.h*
      - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `fsync` (man-pages, n.d.)
  - `int fsync(int fd);`
    - параметри
      - `fd` – дескриптор датотеке
    - повратна вредност
      - успешно извршење
        - 0
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – truncate<sup>(man-pages, n.d.)</sup>
  - подешавање дужине датотеке на задату вредност
    - случај промене дужине
      - могућност продужења датотеке
        - продужење додавањем нултих бајтова
      - могућност скраћења датотеке
    - функција `int truncate(...)` из заглавља *unistd.h*
      - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – truncate<sup>(man-pages, n.d.)</sup>
  - `int truncate(const char * path, off_t length);`
    - параметри
      - `path` – путања
        - потребно је да постоји могућност писања у датотеку
      - `length` – циљна дужина датотеке у бајтовима
    - повратна вредност
      - успешно извршење
        - 0
        - позиција у датотеци не бива промењена
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – truncate<sup>(man-pages, n.d.)</sup>
  - функција `int ftruncate( . . . )` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – truncate<sup>(man-pages, n.d.)</sup>
  - `int ftruncate(int fd, off_t length);`
    - сличност с функцијом `truncate(...)`
      - главне разлике
        - први параметар се односи на дескриптор датотеке
        - одговарајућа датотека треба да буде отворена на начин који дозвољава писање

# Системски позиви

- Системски позиви – `unlink` (man-pages, n.d.)
  - брисање назива из датотечког система
    - случај када је назив последња веза према датотеци
      - датотека бива обрисана
        - ако је датотека отворена, до брисања долази тек када и последњи дескриптор датотеке за ту датотеку буде затворен
  - функција `int unlink(...)` из заглавља *unistd.h*
    - стандардна библиотека за програмски језик C



# Системски позиви

- Системски позиви – `unlink` (man-pages, n.d.)
  - `int unlink(const char * pathname)`
    - параметри
      - `pathname` – путања
    - повратна вредност
      - успешно извршење
        - 0
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `stat` (man-pages, n.d.)
  - читавање статуса датотеке
    - добављање информација о датотеци и њихово смештање у бафер
      - примери информација о датотеци
        - ознака уређаја, бројчана ознака датотеке у датотечком систему, ознака корисника који је власник, величина датотеке, најскорије време приступа, најскорије време модификације
  - функција `int stat( . . . )` из заглавља `sys/stat.h`
    - стандардна библиотека за програмски језик C

# СИСТЕМСКИ ПОЗИВИ

- Системски позиви – stat (man-pages, n.d.)
  - `int stat(const char *restrict pathname, struct stat *restrict statbuf);`
    - параметри
      - `pathname` – путања
      - `statbuf` – бафер
    - повратна вредност
      - успешно извршење
        - 0
      - појава грешке
        - -1 (врста грешке је назначена у променљивој `errno`)

# Системски позиви

- Системски позиви – `stat` (man-pages, n.d.)
  - функција `int fstat(...)` из заглавља `sys/stat.h`
    - стандардна библиотека за програмски језик C

# Системски позиви

- Системски позиви – `stat` (man-pages, n.d.)
  - `int fstat(int fd, struct stat *statbuf);`
    - сличност с функцијом `stat(...)`
      - главна разлика
        - први параметар се односи на дескриптор датотеке

# Системски позиви

- Системски позиви (man-pages, n.d.)
  - системски позиви за рад над датотекама – додатни примери посредничких функција
    - `dup` – дупликација дескриптора датотеке
    - `fcntl` – манипулација дескриптором датотеке
    - `link` – формирање јаке везе према датотеци
    - `mkdir` – креирање директоријума
    - `mknod` – креирање обичне или посебне датотеке
    - `rename` – промена назива или локације датотеке
    - `rmdir` – брисање директоријума

# Системски позиви

- Системски позиви (man-pages, n.d.)
  - системски позиви за рад над датотекама – додатни примери посредничких функција
    - `symlink` – формирање симболичке везе према датотеци
    - `sync` – снимање несачуваних измена метаподатака датотечког система и кешираних података датотека у одговарајуће датотечке системе

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- **Методе приступа**
- Накнадна разматрања
- Ресурси



# Методе приступа

- Метода приступа (Mogin, 2008)
  - скуп рутина у неком програмском језику које су посвећене изградњи и одржавању структуре датотеке и упису и читању података из датотеке, уз уважавање логичке структуре слогова

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- **Накнадна разматрања**
- Ресурси

# Накнадна разматрања

(Agrawal et al., 2007; Arpaci-Dusseau & Arpaci-Dusseau, 2023)

- Обрасци у коришћењу датотечких система
  - петогодишње истраживање метаподатака датотечких система (Агравал, Болоски, Досер и Лорч; 2007)
    - период од 2000. до 2004. године
    - прикупљани подаци о локалним датотечким системима на стоним рачунарима с оперативним системом фамилије *Windows* у компанији *Microsoft*
    - преко 60000 датотечких система
    - заступљене врсте датотечких система
      - *FAT* (5%)
      - *FAT32* (15%)
      - *NTFS* (80%)

# Накнадна разматрања

(Agrawal et al., 2007; Arpaci-Dusseau & Arpaci-Dusseau, 2023)

- Обрасци у коришћењу датотечких система
  - петогодишње истраживање метаподатака датотечких система (Агравал, Болоски, Досер и Лорч; 2007)
    - нека запажања
      - бројеви датотека и директоријума по датотечком систему се повећавају
      - већина директоријума садржи мање од 20 ставки
      - већина датотека је величине која не прелази 4 KB
      - велики број малих датотека заузима мали део искоришћеног простора
        - у просечном датотечком систему у 2004. датотеке чија величина не прелази 4 KB заједно су заузимале 52 MB
      - просечна величина датотеке се повећава
        - са 108 KB на 189 KB

# Накнадна разматрања

(Agrawal et al., 2007; Arpaci-Dusseau & Arpaci-Dusseau, 2023)

- Обрасци у коришћењу датотечких система
  - петогодишње истраживање метаподатака датотечких система (Агравал, Болоски, Досер и Лорч; 2007)
    - нека запажања
      - већина података (бајтова) је у великим датотекама
        - посебно заступљени су видео садржај, базе података и датотеке типа *blob*
      - учешће датотека оперативног система у искоришћеном простору се повећава
      - просечна потрошња простора по датотечком систему се повећава
        - са 3 GB на 18GB
      - просечна искоришћеност расположивог простора је релативно стабилна
        - са 49% на 45%

# Садржај

- Увод
- Простор на екстерном меморијском уређају
- Каталог
- Физичка размена података
- Спрега између програма и датотека
- Системски позиви
- Методе приступа
- Накнадна разматрања
- **Ресурси**

# Ресурси

- Извори и литература
  - Pavle Mogin. Strukture podataka i organizacija datoteka. 3. izdanje. Računarski fakultet (Beograd, Srbija), CET (Beograd, Srbija). 2008.
    - Glava 3. Uloga operativnog sistema u upravljanju podacima
    - Glava 5. Fizičke strukture datoteka i postupci razmene podataka

# Ресурси

- Извори и литература
  - Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau. Operating Systems: Three Easy Pieces. Version 1.10. Arpaci-Dusseau Books. 2023.
    - 2. Introduction to Operating Systems
    - 19. Paging: Faster Translations (TLBs)
    - 37. Hard Disk Drives
    - 39. Interlude: Files and Directories
    - 40. File System Implementation
    - 41. Locality and The Fast File System
    - 50. The Andrew File System (AFS)



# Ресурси

- Извори и литература
  - Vincent van der Meer, Hugo Jonker, Jeroen van den Bos. A Contemporary Investigation of NTFS File Fragmentation. Forensic Science International: Digital Investigation. 2021; 38(Suppl.); 301125.
  - Nitin Agrawal, William J. Bolosky, John R. Douceur, Jacob R. Lorch. A Five-Year Study of File-System Metadata. Proceedings of the 5th USENIX Conference on File and Storage Technologies; 2007 Feb 13–16; San Jose, CA, USA. Berkeley, CA, USA: USENIX Association; 2007. 31–45.
  - Microsoft. Overview of FAT, HPFS, and NTFS file systems. [Internet]. 2025. URL: <https://learn.microsoft.com/en-us/troubleshoot/windows-client/backup-and-storage/fat-hpfs-and-ntfs-file-systems>

# Ресурси

- Извори и литература
  - The Linux Kernel Archives. About Linux Kernel. [Internet]. (n.d.). URL: <https://www.kernel.org/linux.html>
  - The Linux Kernel Documentation. Filesystems in the Linux Kernel. [Internet]. (n.d.). URL: <https://docs.kernel.org/filesystems/index.html>
  - man-pages. The Linux man-pages Project. [Internet]. (n.d.). URL: <https://www.kernel.org/doc/man-pages/>
  - Lars Wirzenius, Joanna Oja, Stephen Stafford, Alex Weeks. The Linux System Administrator's Guide. [Internet]. (n.d.). URL: <https://tldp.org/LDP/sag/html/index.html>

# Ресурси

- Извори и литература
  - The Austin Common Standards Revision Group. POSIX.1 FAQ. [Internet]. 2020. URL:  
[https://www.opengroup.org/austin/papers/posix\\_faq.html](https://www.opengroup.org/austin/papers/posix_faq.html)
  - The Austin Common Standards Revision Group. POSIX.1 Backgrounder. [Internet]. (n.d.). URL:  
<https://www.opengroup.org/austin/papers/backgrounder.html>

# Ресурси

- Извори и литература
  - OpenAFS. Welcome to the Home of OpenAFS. [Internet]. (n.d.). URL: <https://www.openafs.org/>
  - OpenAFS. OpenAFS Documentation. [Internet]. (n.d.). URL: <https://docs.openafs.org/index.html>
  - BTRFS Documentation. Welcome to BTRFS Documentation!. [Internet]. (n.d.). URL: <https://btrfs.readthedocs.io/en/latest/>