

# ДИГИТАЛНИ ПЛАТФОРМИ ЗА БИОИНФОРМАТИКА

---

ПРОФ. БОРОВСКА

# ДИГИТАЛНИ ПЛАТФОРМИ

---

ДЕФИНИРАНЕ, БИЗНЕС МОДЕЛИ И  
КЛЮЧОВИ ХАРАКТЕРИСТИКИ



# 3

## ДИГИТАЛНИ ПЛАТФОРМИ

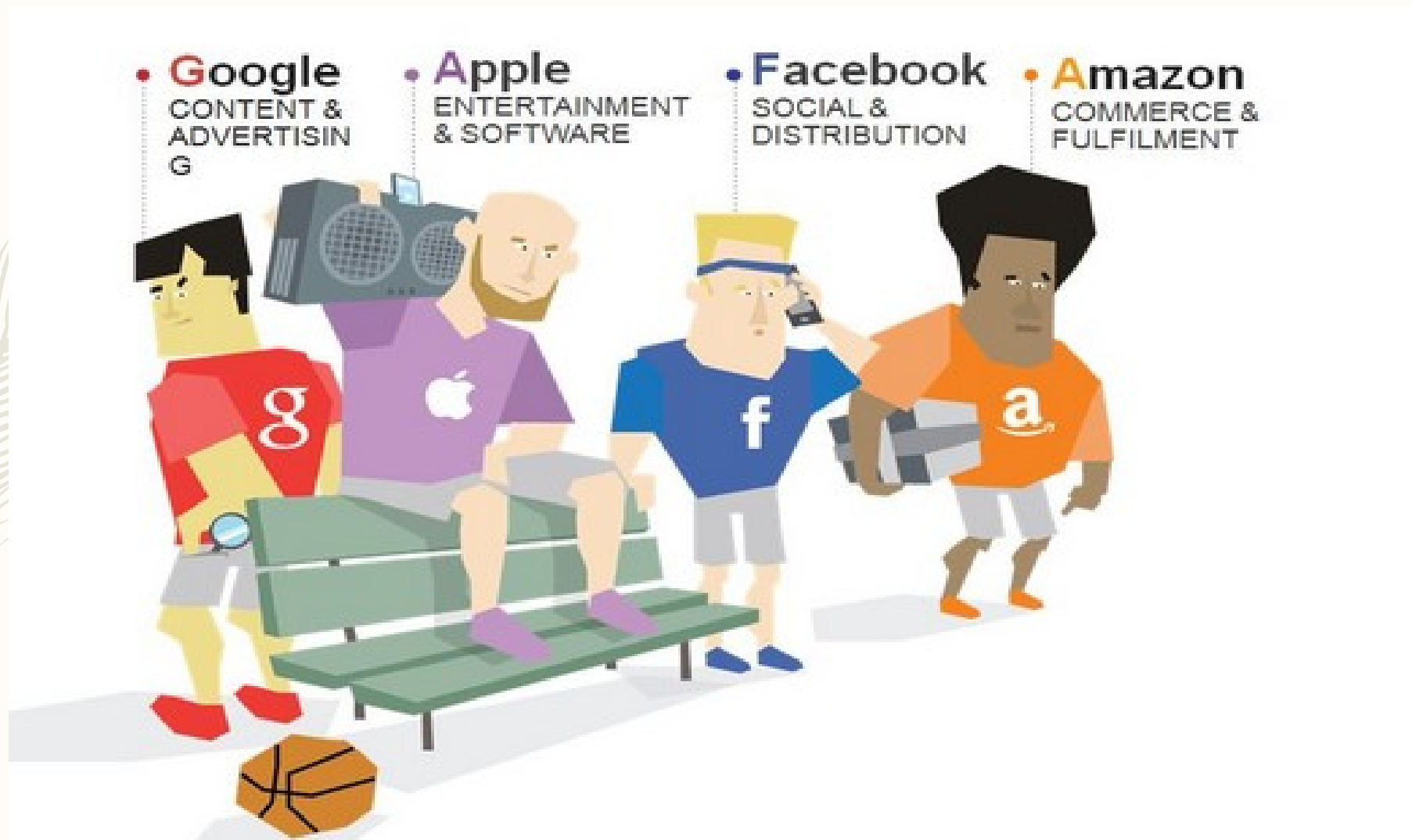
---

- Осигуряват среда за изпълнение на софтуер
- Може да бъде изчислителна инфраструктура (хардуер), ОС, уеб браузер и свързаните с него приложни програмни интерфейси (APIs), или друг базов софтуер, необходим за изпълнението на програмния код.
- Дигиталните платформи могат да имат различни нива на абстракция, обхващащи компютърна архитектура, ОС, или runtime библиотеки.
- Осигурява софтуерни инструменти и инфраструктура за развитие на софтуер, създаване на дигитално съдържание, видео, съответно.
- Съществуват и друг тип платформи като Alibaba и Uber, например, при които целта е да улеснят директния обмен

# ДИГИТАЛНИТЕ ПЛАТФОРМИ КАТО ИНОВАТИВНИ БИЗНЕС МОДЕЛИ

---

- **Платформи с търсачки** - Google search engine
- **Социални платформи** : Facebook, Twitter, Instagram, LinkedIn
- **Платформи за знания (Knowledge platforms)** – форуми от вида на StackOverflow
- **Платформи за научни изследвания** – биоинформатични платформи
- **Магазини за приложения (Application stores)** - Apple/Google Play
- **Пазари**: Amazon market place, shopping engines (бизнес модел е-търговия)
- **Медийни платформи**: Spotify, Deezer (бизнес модел „абонамент“ (“subscription”).
- **Партньорски платформи (Affiliate platforms)**: една от най-големите и утвърдени мрежи за партньорски маркетинг, създадена за постигане на интелигентен, скалируем и устойчив растеж
- **Платформи за набиране на тълпи (Crowd-sourcing platforms)** - Uber, BlablaCar, AirBnB – “pay as you go” бизнес модел
- **Платформи за хранилища (Repository platforms)** - GitHub – “freemium” бизнес модел : you pay only if you don’t want to share your code with the community.
- **Инфраструктурни платформи (Infrastructure platforms) (IaaS)**: AWS, Azure – “pay as you go” бизнес модел
- **Платформи за забавления** - Netflix- Internet entertainment service





## 6 Определение

---

*Дигиталната платформа е plug-and-play бизнес модел, който позволява на множество участници (производители и потребители) да се свързват с нея, да взаимодействат помежду си и да създават и обменят стойност*



# 7

## Ключови характеристики

---

- Основават се на технологично базиран бизнес модел.
- Улеснява обмена между множество групи - например крайни потребители и производители, които не е задължително да се познават.
- Предлагат стойност, пропорционална на размера на общността и мрежови ефекти.
- Доверителност: доверие с ясни общи условия по отношение на интелектуалната собственост и собствеността на данни.



# 8

## Ключови характеристики

---

- Има отворена свързаност: споделя данни с разработчици на трети страни за създаване на нови услуги и разширяване на екосистемата – посредством API.
- Платформата може да се скалира масово за справяне с милиони потребители без влошаване на производителността.
- Осигурява лесен достъп за потребителите, не е необходимо предварително или допълнително обучение, самообслужване.
- Платформата може да използва иновативни бизнес модели, базирани на непосредствената стойност.





# 9

## Ползи

---

- Генериране на приходи
- Намаляване на разходите
- Насърчаване на сътрудничеството и иновациите за нови продукти и услуги
- Скъсяване на времето за пускане на продукти на целевите пазари

# ПЛАТФОРМИ ЗА ХРАНИЛИЩА

---

**Repository platforms**



- Колаборативна платформа за развитие на софтуер
- GitHub, Inc. е глобална компания (САЩ), дъщерно дружество на Microsoft, което придобива компанията през 2018 г. за 7,5 милиарда щатски долара
- Предоставя хостинг услуги за контрол на версиите за разработка на софтуер, използвайки Git.
- Написана на **Ruby** - интерпретативен език за програмиране на високо ниво, създаден в средата на 1990-те години от Yukihiro Matsumoto "Matz" в Япония
- Ruby е от типа на езиците за динамично програмиране и използва garbage collection.
- Поддържа множество парадигми за програмиране обхващащи тези на процедурното, обектно-ориентираното и функционалното програмиране.

# 12 Git



- *Git е система за разпределено управление на версиите за проследяване на промените в сорс кода в процеса на разработката на софтуер*
- Проектиран е за координиране на съвместната работа на разработчиците на софтуера, но може да се използва за проследяване на промените в произволно множество файлове.
- Основните цели обхващат бързодействие, интегритет на данните, и поддръжка на разпределени нелинейни работни потоци (workflows)
- Git е създаден от Linus Torvalds през 2005г. за разработката на Linux kernel, съвместно с партньори, които са надграждали началния проект
- Всеки каталог на Git представлява напълно развито хранилище съдържащо цялата история на промените с пълна функционалност за проследяване на промените независими от достъпа до мрежата или от централен сървър
- Git е open-source software под GNU General Public License version 2.

# 13 Github - fork, pull request and merge

---

- GitHub представлява ***Git repository hosting service***, която добавя нови характеристики.
- Git е софтуерен инструмент базиран на командни линии (command line tool)
- GitHub осигурява Web-базиран графичен интерфейс.
- GitHub осигурява контрол на достъпа и функционалност за колаборация, като wikis и базови софтуерни инструменти за управление на задачите в рамките на всеки проект.
- Водещата функционалност на GitHub е разклоняване “forking” – копиране на хранилището от акаунта на един потребител в акаунта на друг потребител
- Осигурява достъп на потребител, който не участва в разработката на даден проект, но може да модифицира кода в неговия собствен акаунт.
- При желание да се споделят промени, потребителят изпраща нотификация “pull request” към собственика на проекта. При разрешение с натискането на бутон, промените в хранилището на заявителя се отразяват в хранилището (repo) на проекта.



# 14 Atlassian Bitbucket

- 
- <https://www.atlassian.com/software/jira>
  - ви позволява да хоствате частни, локални Git хранилища с функции за екипно сътрудничество в стил BitBucket / GitHub.
  - Компанията също така продава софтуени инструменти за колаборация за разработчици като бърг тракер Jira и wiki Confluence
  - Knowledge management
  - Development workflow
  - Continuous integration and deployment
  - Confluence - team workspace for knowledge and collaboration

# 15

## Bitbucket

<https://bitbucket.org>

---

- Bitbucket осигурява пространство и функционалност за управление на кода в Git, планиране на софтуерните проекти, колаборативно развитие на кода, тестване и внедряване на софтуера
- Създаване на качествен код с преглед на кода
- Одобряване на прегледа на кода посредством „*pull requests*“
- Създаване на контролен списък (*check list*) при обединяването на секции код при специфицирани лица за одобрението им
- Провеждане на дискусии посредством вградени (*inline*) коментари
- Интеграция с Trello (<https://trello.com/>) – дъските (*boards*), списъците и картите (*cards*) на Trello осигуряват функционалност за приоритизиране на проектите по гъвкав и ефективен начин

# 16 Bitbucket

<https://bitbucket.org/>

---

- **Bitbucket hosting options** – cloud, server, data center
- **branch** - независима линия на развитие на кода.
- **Fork** - Вместо да се използва едно хранилище от страна на сървъра, за да действа като „централна“ база на кода, forking осигурява на всеки програмист хранилище от страна на сървъра - всеки сътрудник има не едно, а две хранилища на Git: частно локално хранилище и публично (споделено) хранилище.
- **HEAD** – начинът на Git за рефериране на текущия snapshot. Командата на Git актуализира HEAD да указва специфициран клон или добавяне на код (commit)
- **Master** - клон за развитие по подразбиране (default development branch) – при създаване на git хранилище, се създава и активира клон с име "master.

# 17

## Bitbucket

<https://bitbucket.org/>

---

- **Bitbucket Pipelines** – интегрирана услуга, вградена в Bitbucket. Осигурява функционалност за автоматично изграждане (build), тестване и внедряване на кода, на основата на конфигурационен файл в хранилището
- **Bitbucket Pipes** - къси секции код се прибавят към Pipelines за специфицирана функционалност.
- **Pipes** – осигуряват функционалност за лесно и ефективно изграждане и изпълнение на автоматизирани работни потоци (CI/CD workflows)
- **CI/CD** - Continuous Integration, Continuous Delivery, and Continuous Deployment, (GitLab CI/CD).

# 18

## Bitbucket

<https://bitbucket.org/>

---

- **Project** – контейнер на хранилища (container for repositories).
- **Pull requests** – характеристики, улесняващи колаборацията на разработчиците на софтуер посредством user-friendly web интерфейс за обсъждане на предлаганите промени преди интеграцията им в официалния проект.
- **Working Tree** – дървото на актуалните проверени и одобрени файлове, обхваща съдържанието на HEAD commit's tree и всички промени направени локално преди да бъдат добавени (committed).
- **Bitbucket + HubSpot Integrations** – свързване с apps



# НАУЧНИ РАБОТНИ ПОТОЦИ

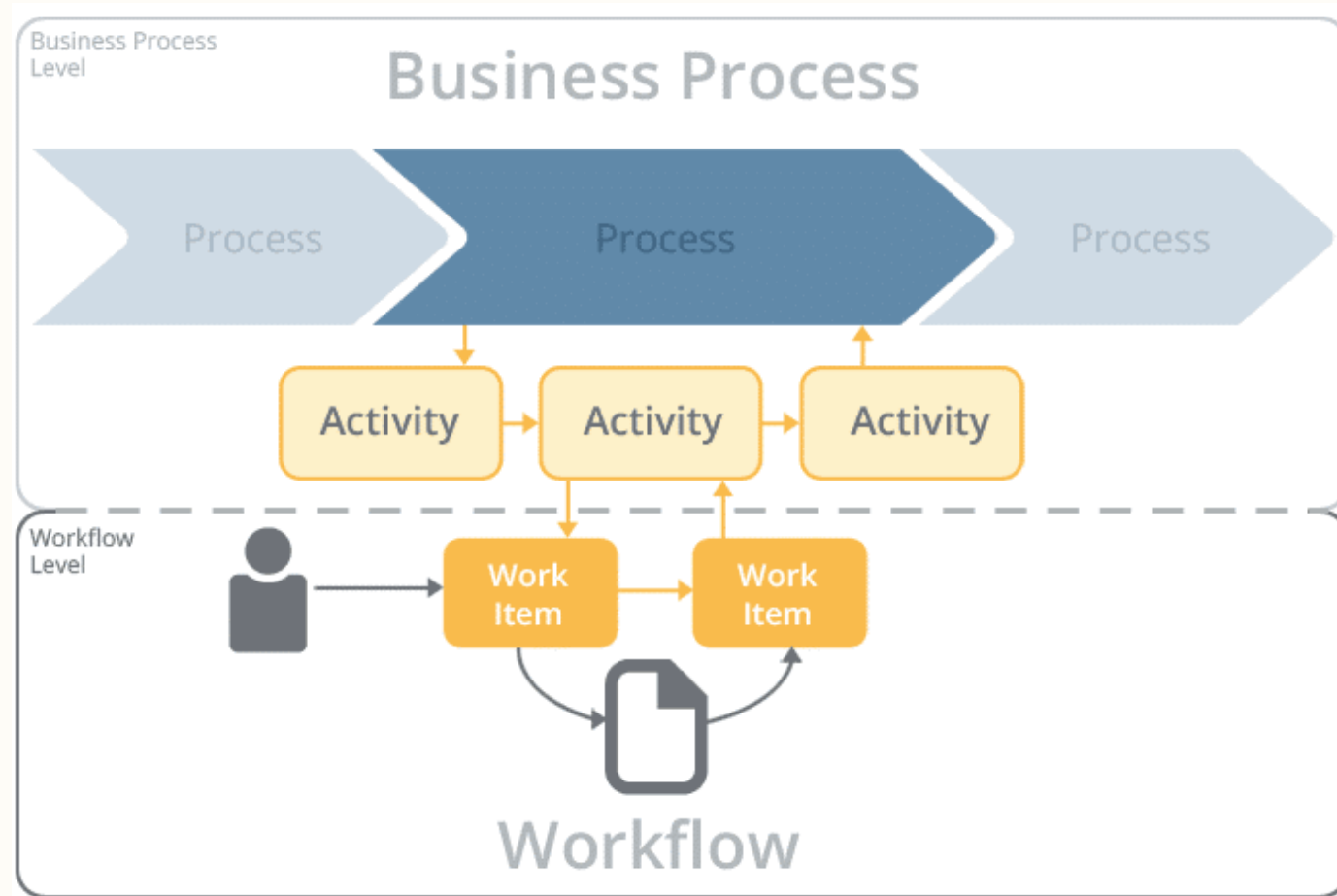
---

**SCIENTIFIC WORKFLOWS**

# 20 Работни потоци (workflows)

## ДЕФИНИЦИЯ:

Последователността от индустриални, административни или други процеси, през които част от работата преминава от започването ѝ до завършването ѝ



- Най-простият компютъризиран научен работен поток обхваща скриптове, които извикват данни, изпълняват програми и извикват евентуално други входни данни, генерирайки резултати, които могат да включват визуализации и аналитични резултати.
- Научните работни потоци могат да бъдат имплементирани с R или MATLAB, като се използва скриптов език като Python или Perl с интерфейс на команден ред или по-скоро с използване на уеб приложения с отворен код като Jupyter Notebook.
- Има много мотиви за разграничаване на научните работни потоци от традиционните работни потоци в бизнес процесите.

- изключително полезни за стимулиране на е-науката и иновациите
- осигуряване на лесна за използване среда за учени от различни области, за да създадат свои собствени работни потоци.
- осигуряване на интерактивни софтуерни инструменти за учените, които им позволяват да изпълняват самостоятелно работните си потоци и да преглеждат резултатите си в реално време.
- опростяване на процеса на споделяне и повторно използване на работните процеси
- позволява на учените да проследяват произхода на резултатите от изпълнението на работния процес и стъпките за създаване на работния процес.

- Методът на работните потоци осигурява създаването на цялостно дигитално решение, подпомагащо проектирането и провеждането на експерименти, като се използва верига от софтуерни инструменти (toolschain).
- Работният поток се дефинира като модел, определящ последователното изпълнение на процесите или потока от задачи, който се координира и планира въз основа на систематичен план.
- Научните работни процеси осигуряват метод за дефиниране на целите на експеримента на високо ниво, моделиран от работния поток на научните задачи.
- Различни видове задачи могат да се изпълняват в рамките на работния поток, особено когато изходите от една задача се използват като вход за следващата задача (конвейер).



- Научните работните потоци значително подобряват качеството на анализа на данните, особено когато данните се получават от множество източници и се генерират от изчисления на разпределени ресурси и / или различни софтуерни инструменти за анализ.
- Този напредък в системния анализ на научната информация, осигурен от работните потоци, предопредели нарастващата необходимост от автоматизирани приложения, управлявани от данни, които също събират и управляват произхода на данните и процесите с малко взаимодействие на потребителите и с приемливи допълнителни системни разходи.
- Парадигмите за научните изследвания представлява непрекъснато развиващ се процес, при който учените наблюдават явления, събират данни, разработват хипотези, извършват тестове, отричат или променят хипотези, повтарят процеса с различни данни.

# 25

## Scientific workflows

---

- Научните изследвания предполагат итеративен процес на експериментиране и получаване на резултати, изследвайки значимостта на различни фактори.
- *Научно-изследователския процес изисква ефективно използване и натрупване на експериментални данни, както и възможността за повторно възпроизвеждане на експериментите за валидиране на резултатите.*
- Научните работни потоци са мощно средство и решаващ фактор за развитието не само на е-науката, а също така и на дигиталната научно- изследователска инфраструктура.
- *Научните работни потоци са средство, чрез което учените могат да планират експериментите си, да моделират процеси или явления, да изпълняват серия от взаимно свързани експерименти, да отстраняват грешки, пренастроят и рестартират изчислителните конвейери за анализ и визуализация.*

# 26

## Scientific workflows

---

- *Задължително изискване при съвременните научни изследвания е да може да се установи произхода на даден резултат, като се дефинират входните набори данни, експерименталните методи и използваните софтуерни инструменти, параметрите, настройките и т.н.*
- Подобна информация е полезна за изследователя да интерпретира резултатите от работния поток, както и за други изследователски екипи да могат евентуално да възпроизведат експеримента и да верифицират експерименталните резултати.

# Система с научни работни потоци

---

- Важна среда за *in silico* експериментиране, като освен механизми и средства за създаване и изпълнение на работни потоци трябва да осигурява средства за дефиниране на входно/ изходните потоци данни между отделните фази на работните потоци и координацията на функционирането на изчислителния конвейер.
- Тези изисквания предполагат необходимост от запазване на информация за произхода и извличането на данни и процеси в работния поток, за обвързване на резултатите с персонализирана и изпълнима версия на работния поток, както и за проследяване на развитието на изпълнението на работния поток.
- Целта е да се осигури специфицирана логическа последователност от *in silico* експерименти със строго дефинирана функционалност и входно/изходни интерфейси на отделните фази.

## (Workflow Management System - WMS)

---

- Представяват софтуер, който осигурява инфраструктура за конфигуриране, изпълнение и мониторинг на научните работни потоци.
- Осигуряват средата, в която се дефинират и изпълняват *in silico* експериментите.
- Важна функция на WMS е координацията на индивидуалните компоненти, изграждащи работния поток, който често се нарича „**оркестриране**“.
- Научните изследвания стават все по-интензивни и все повече разчитат на използването на компютри, необходимо е големи обеми от експериментални данни да се записват по-бързо и с по-голяма точност.
- Тази тенденция предизвиква значително увеличаване на сложността на научния софтуер за симулация.



# 29 Системи за управление на работните потоци

## (Workflow Management System - WMS)

---

- Много софтуерни инструменти изпълняват само малка добре дефинирана задача, като по този начин се налага няколко от тях да бъдат съединени в рамките на конвейер за моделиране на цялостен експеримент.
- Допълнителни трудности възникват от необходимостта за справяне с проблема за *оперативната съвместимост* - несъвместимите формати на данни, които различни услуги произвеждат или консумират.
- Научните работни потоци и WMS решават тези проблеми като предоставят лесен за използване начин за специфициране на задачите, които трябва да бъдат изпълнени по време на даден *in silico* експеримент.
- Комбинирането на няколко софтуерни инструмента в рамките на един изследователски анализ се осъществява от системите за управление на работните потоци.



# 30

## *Системи за управление на работните потоци*

### (Workflow Management System - WMS)

---

- Съвременните системи с работни потоци представляват платформи с front end система за визуално програмиране, позволяваща на потребителите лесно да конструират своите приложения като визуална графика, свързвайки графично фазите на работния поток, а също така са разработени инструменти за използване на платформено-независим начин.
- Ключов компонент в технологията на работните потоци е софтуерното приложение, което управлява процесите – т.нар. „двигател“ (engine).
- „Двигателят“ на работните потоци имплементира основна техника за създаване на софтуер за разпределение на задачи, като например управление на процеси, при който двигателят разпределя задачите на различни изпълними модули, поддържайки взаимодействието между участниците, и в общия случай използва сървър на база данни.

# 31

## *Системи за управление на работните потоци*

### (Workflow Management System - WMS)

---

- „Двигателят“ на работните потоци управлява потоците от данни, задачи и събития.
- Функционалността му обхваща управлението и мониторинга на дейностите в рамките на работния поток като например, обработката на файл с данни, или определянето на прехода към нова фаза на обработката съобразно специфицираната последователност от дейности.
- Основните отговорности могат да бъдат обобщение както следва:
  1. Верификация на статуса на текущата задача;
  2. Оторизация на потребителите;
  3. Изпълнение на условен скрипт.

- 
- *Conda е крос-платформа с отворен код, езиково-агностичен мениджър на софтуерни пакети и система за управление на програмната среда.*
  - *Езиково-агностичното програмиране или скриптове* (наричани още езиково неутрални, независими от езика или многоезични) е парадигма за разработка на софтуер, при която определен език е избран като подходящ за определена задача (като се вземат предвид всички фактори, включително екосистемата , набори от умения на програмисти, производителност и т.н.), а не само поради уменията и опита на екипа за разработка.
  - Например, езиково Java екип за езиково - агностична разработка на софтуер може да избере да използва Ruby или Perl разработка на даден модул, при който Ruby или Perl биха били по-подходящи от Java.

- 
- „*Cross-Language*“ в програмирането и скриптирането се използва за развитие на софтуер, при който два или повече езика трябва да бъдат използвани в програмния код заедно с основния език за програмиране, избран за кодиране на софтуера.
  - Независимо дали това означава включване на скрипт като източник, който да се използва при необходимост, изпълним код с виртуални машини, независим от езика, като JVM, или модели на обекти като COM, целта е да си сътрудничат помежду си или да се избират езици, които работят добре заедно.





- 
- Функционалност - управление на софтуерни пакети, зависимости и програмна среда за всеки език - Python, R, Ruby, Scala, Java, JavaScript, C / C ++, FORTRAN и други.
  - Conda лесно създава, съхранява, зарежда и превключва между среди
  - Софтуерният пакет conda package и environment manager са включени във всички версии на Anaconda и Miniconda
  - Conda е включен и в Anaconda Enterprise, осигуряващ on-site enterprise package and environment management for Python, R, Node.js, Java and other application stacks.



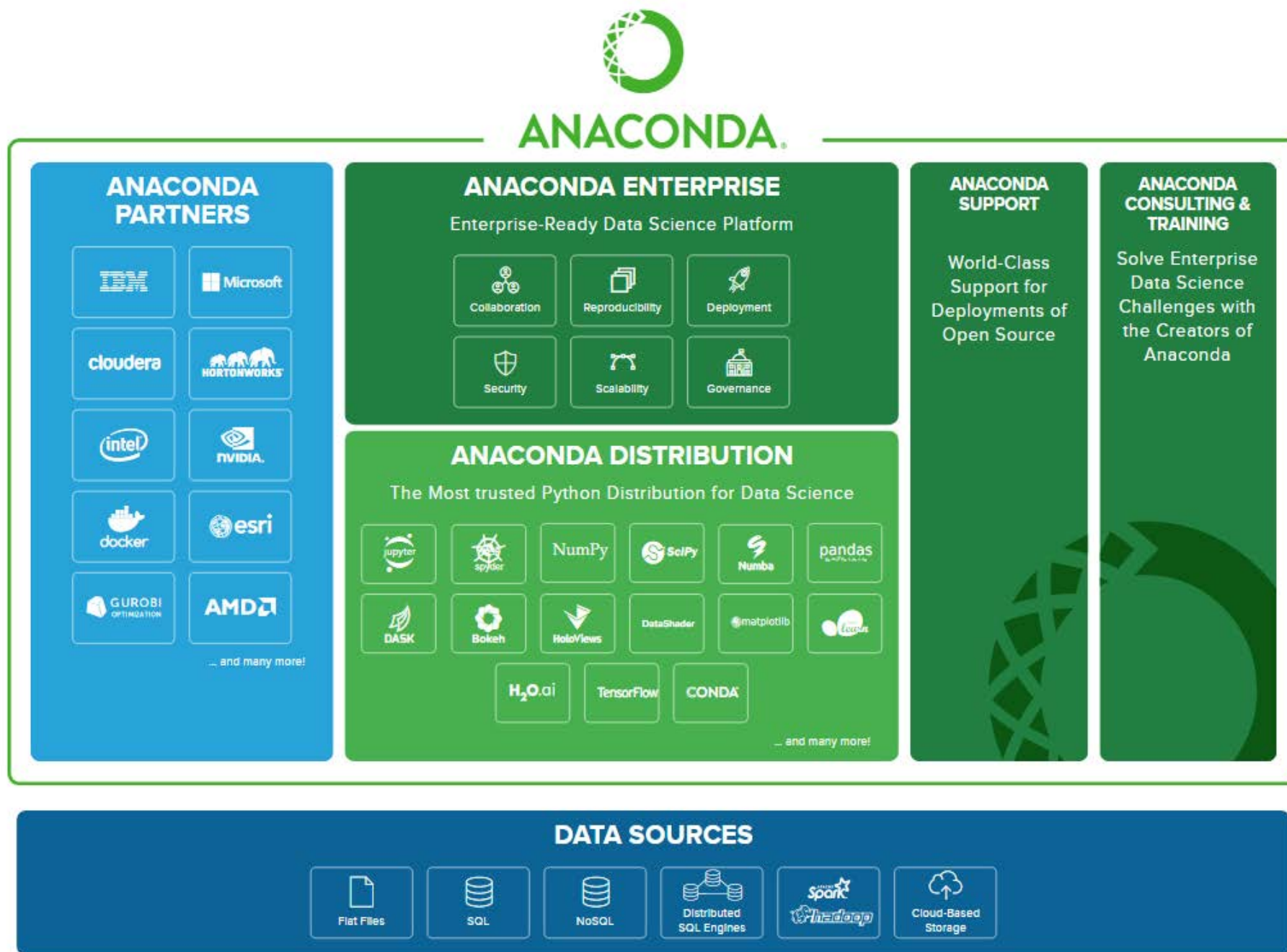
# BIOCONDA<sup>®</sup>

- 
- Bioconda е канал (channel) за conda package manager за софтуери за биоинформатика
  - <https://bioconda.github.io/>
  - <https://anaconda.org/bioconda/>
  - Обхваща:
    1. a repository of recipes hosted on GitHub
    2. a build system turning these recipes into conda packages
    3. a repository of packages containing over 7000 bioinformatics packages ready to use with
    4. over 850 contributors and 570 members who add, modify, update and maintain the recipes

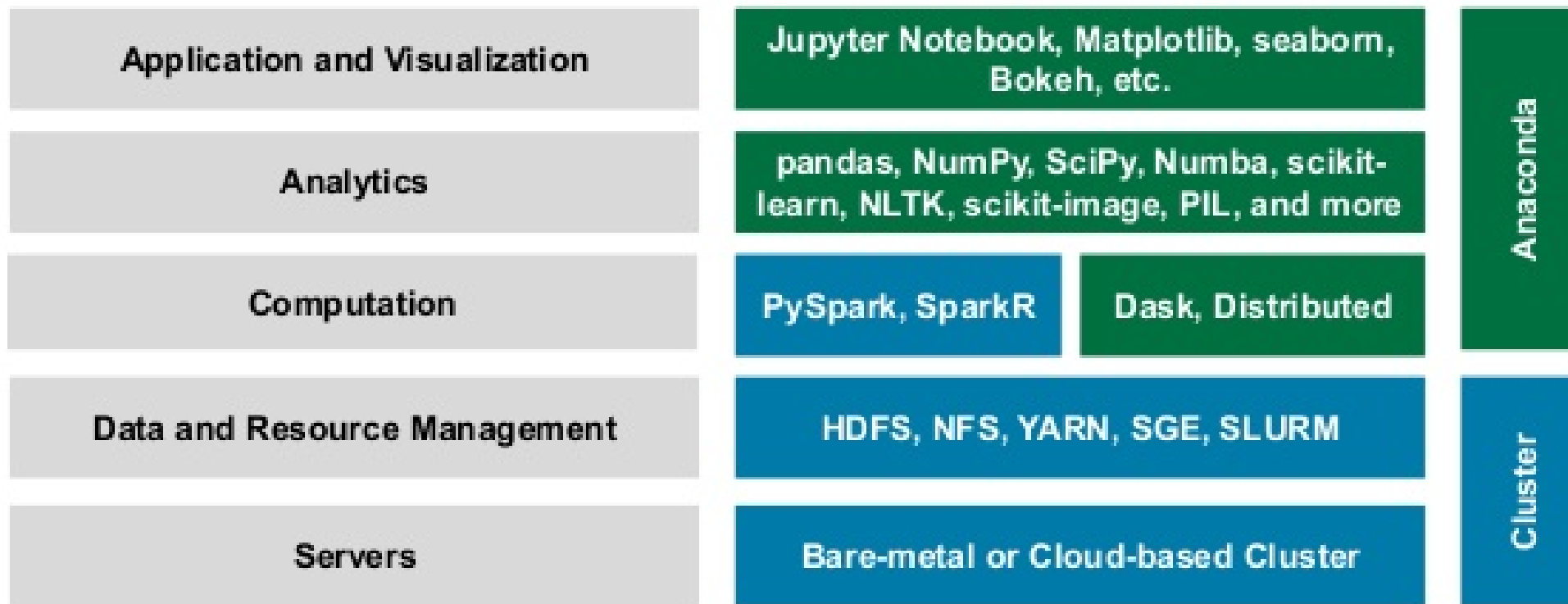
36



ПРОФ. БОРОВСКА



# Anaconda - Scaled Out Open Data Science





Open Data Science Deep Learning, Machine Learning, AI Interactive Dashboards & Apps

Big Data Data Science Collaboration Data Science Governance Data Science Deployment Self-Service Analytics

### OPEN DATA SCIENCE



### DATA



### COMPUTATION



## RESULTS

- ✓ **Deploying Data Science models and visualizations easily**
- ✓ **Building interactive visualizations and web applications**
- ✓ **Collaboration and sharing of results easily**
- ✓ **Scaling up & out Data Science to meet performance goals**
- ✓ **Reproducibility and governance for Data Science assets**

- Биоинформатичните платформи са специализирани за научни работни потоци за целите на биоинформатиката.
- Понастоящем, съществува широк спектър от биоинформатични платформи
- фокусът е върху 3 платформи – Unipro UGENE, Taverna и Galaxy



- представлява мултиплатформен софтуер с отворен код, чието основно предназначение е да подпомага молекулярните биолози в управлението на техните *in silico* експерименти, както и при визуализацията и анализа на експерименталните резултати.
- UGENE интегрира широк спектър биоинформатични софтуерни инструменти в рамките на общ потребителски интерфейс.
- Поддържат се множество биологични формати на данни и се осигурява извличането на данни от отдалечени източници.
- Съдържа модули за визуализация на биологични обекти като анотирани секвенции на геноми, асемблирани NGS данни, множествоно подравняване на биологични секвенции, конструиране на филогенетични дървета и тримерни структури на протеини.



- UGENE предлага среда за визуализация при създаването на многократно изпълними работни потоци, които могат да бъдат изпълнявани както на локални, така и на отдалечени високо производителни изчислителни инфраструктури.
- UGENE е написан на езика за програмиране C++ в работната рамка Qt.
- Вградената plugin системи и структурираният приложен програмен интерфейс UGENE API осигурява възможности за разширяването на системата от софтуерни инструменти с нови функционалности.
- Workflow Designer на UGENE е софтуерен инструмент за визуализация при изграждането на сложни аналитични конвейери. UGENE е самостоятелно приложение и не изисква инсталирането на допълнителни приложения.

Unipro UGENE - Integrated Bioinformatics Software

Not secure | ugene.net

Unipro UGENE

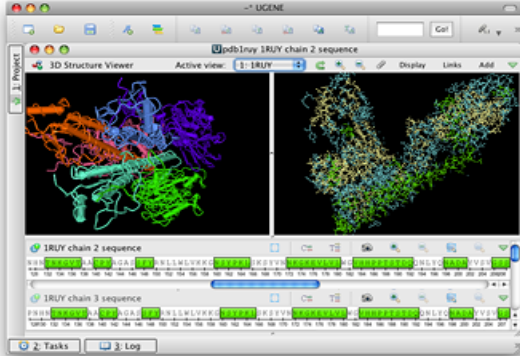
HomeDocumentationLearnCommunityDownloadsSupportContactsDonate

Unipro UGENE  
v. 33 September, 2019

UGENE is free open-source cross-platform  
bioinformatics software

It works perfectly on Windows, macOS and Linux  
and requires only a few clicks to install

Download PDFDownload UGENE



News

September 30th, 2019  
UGENE 33 has been released

June 17th, 2019  
UGENE's 11th anniversary

Podcast

Ep54: What's new in UGENE 1.31 & 1.32 – Part 1. Metagenomic workflows

Ep53: What's new in UGENE 1.28, 1.29 & 1.30

Board

>Feature Requests: Donate a little bit

>Feature Requests: Re: DNA-Star file conversion

Cite Us

Okonechnikov K, Golosova O, Fursov M, the UGENE team. **Unipro UGENE: a unified bioinformatics toolkit**. *Bioinformatics* 2012 28: 1166-1167. doi:10.1093/bioinformatics/bts091

Support and Services

Feel free to contact us if you need [technical support](#).

Free Resources

[Browse the documentation](#)

Windows taskbar with icons for File Explorer, Microsoft Store, Visual Studio, Word, Chrome, Mail, Edge, and PowerPoint. System tray shows network, volume, and language (ENG) settings, along with the time 10:02 and date 17.11.2019 r.

## Система за биоинформатични работни потоци **Apache Taverna**

---

- система за управление на работни потоци с отворен код и независима от домейните.
- Представлява набор от софтуерни инструменти, използвани за проектиране и изпълнение на научни работни потоци за *in silico* експериментиране.
- *Taverna* е проект с отворен код от 2003 г. с участието на различни академични и индустриални партньори.
- От 2014г. *Таверна* става проект в инкубатора на *Apache Software Foundation* и променя името си на *Apache Taverna* (инкубационен - *incubating*).

# Система за биоинформатични работни потоци **Apache Taverna**

---

- Проектът разработва *Apache Taverna 3.x* като лицензът е променен от *LGPL 2.1* на *Apache License 2.0*
- *Apache Incubator* е порталът за проекти с отворен код, предназначени да станат пълноценни проекти на *Apache Software Foundation*.
- Проектът за инкубатора е създаден през октомври 2002 г., за да осигури подготовката на външни проекти за присъединяването им към проектите и базите данни на софтуерната фондация *Apache*.
- Всички дарения на код от външни организации и съществуващи външни проекти, които желаят да се преместят в *Apache*, трябва да влязат през Инкубатора.

## Система за биоинформатични работни потоци **Apache Taverna**

---

- Проектът за инкубатора на *Apache* от една страна служи като временен проект за контейнери, докато инкубационният проект бъде приет и се превърне в проект на най-високо ниво на софтуерната фондация Apache или стане подпроект на подходящ проект, като например Jakarta Project или Apache XML.
- От друга страна, проектът за инкубатора документира как работи Фондацията и как да се направят проектите в нейните рамки, което обхваща документирането, ролите и политиките в рамките на софтуерната фондация Apache и нейните членове.

- уеб базирана платформа с отворен код за интензивни биомедицински научни изследвания.
- Представява платформа за управление на научни работни потоци, а също така осигурява средства за интегриране на биологични данни.
- Създадена е от Galaxy Community (първа версия през 2005г.) и обхваща следните 3 проекта: Federated Galaxy Roadmap, Gen3 Galaxy integration, Remote Data Galaxy.
- Платформата е Linux базирана и е написана на езиците Python и JavaScript.



- Galaxy първоначално е предназначена за анализ на биологични данни, по-специално геномика.
- Наборът от налични инструменти се разширява значително с годините и Galaxy вече се използва и за експресия на гени, асемблиране на геноми, протеомика, епигеномика, транскриптомика и множество други дисциплини в науките за живота.
- Самата платформа всъщност е агностичен домейн и може да се приложи на теория към всяка научна област.
- Например, създадени са Galaxy сървъри за анализ на изображения, изчислителна химия и дизайн на лекарства, космология, климатично моделиране, социални науки, и лингвистика.

Galaxy

usegalaxy.org

Galaxy

Analyze Data Workflow Visualize Shared Data Help Login or Register

Using 0%

Tools

search tools

Get Data

Collection Operations

Expression Tools

GENERAL TEXT TOOLS

Text Manipulation

Filter and Sort

Join, Subtract and Group

Datamash

GENOMIC FILE MANIPULATION

FASTA/FASTQ

FASTQ Quality Control

SAM/BAM

BED


VCF/BCF

Nanopore

Galaxy is an open source, web-based platform for data intensive biomedical research. If you are new to Galaxy start here or consult our help resources. You can install your own Galaxy by following the tutorial and choose from thousands of tools from the Tool Shed.

Galaxy 101

an introduction to Galaxy tutorial



Galaxy Training Network

Tweets by @galaxyproject

Galaxy Project Retweeted

Peter van Heusden

@pvanheus

Thanks for the excellent pictures @SamarElsheikh. Today we will discuss running your own Galaxy server and the way forward for effective @galaxyproject use on the African continent!

<https://twitter.com/SamarElsheikh/status/1195235305714044928>

Nov 15, 2019

Galaxy Project Retweeted

Galaxy Training Network

Embed

View on Twitter

History

search datasets

Unnamed history

(empty)

This history is empty. You can load your own data or get data from an external source

Windows Taskbar

05:49

18.11.2019 г.

- Galaxy поддържа прозрачността в научни изследвания, като дава възможност на изследователите да споделят някои от техните Galaxy обекти публично или с конкретни лица.
- Споделените елементи могат да бъдат разгледани подробно, да бъдат преизпълнени по желание и копирани и модифицирани, за да се тестват хипотези.
- Обектите, които се поддържат от Galaxy са история, работни потоци, набори данни и страници.
- Историите, работните потоци и наборите данни могат да съдържат анотации на изследователя.
- *Galaxy Pages* осигуряват средства за създаването на виртуални научни доклади, описващи целия експеримент.



**БЪДЕТЕ ЗДРАВИ!!!**