

# **Kolaborasi lintas platform dan bahasa pemrograman**

# Bahasa pemrograman

- Merupakan notasi untuk mendeskripsikan komputasi kepada orang dan mesin\*
- Saat ini, ada ribuan bahasa pemrograman \*
  - Puluhan diantaranya umum digunakan \*\*

\* Compiler Principles, Techniques, & Tools (2<sup>nd</sup> edition), hal. 13

\*\* Merujuk pada indeks TIOBE

# Bahasa pemrograman: Kebutuhan

- Ketika diharuskan, tidak tahu pilihan bahasa lain, tidak *ingin* tahu pilihan bahasa lain
- Diharuskan?
  - Mata kuliah
  - Peluang karir atau tuntutan pekerjaan
  - Bergabung ke proyek software tertentu
  - Ingin membangun software tertentu
- Tidak tahu pilihan bahasa lain
  - Wajar, ketika baru memulai pemrograman
- Tidak ingin tahu pilihan bahasa lain
  - Lebih banyak konsekuensi yang harus dihadapi

Dikutip dari Wikipedia, sejak tahun 1950, terdapat rata-rata 50 bahasa pemrograman baru setiap dekade. Pada tahun 2010-an saja, sudah terdapat lebih dari 20 bahasa baru [https://en.wikipedia.org/wiki/Timeline\\_of\\_programming\\_languages](https://en.wikipedia.org/wiki/Timeline_of_programming_languages)

# Bahasa pemrograman: preferensi dan tantangan

- Seiring waktu, seorang programmer mungkin lebih produktif dengan bahasa pemrograman tertentu
- Akan tetapi:
  - Teknologi dan platform baru mungkin berkembang, dimana bahasa pemrograman lain *mungkin* lebih cocok
  - Tim kerja membesar, dan terdapat programmer lain yang merasa lebih produktif atau cocok dengan bahasa pemrograman lain

Beberapa bahasa pemrograman kadang tidak terhindarkan: keterbatasan teknis, latar belakang anggota tim, efisiensi, dan lainnya  
(Beberapa bahasa pemrograman untuk membangun satu aplikasi, Noprianto, 2018)

# Kolaborasi lintas bahasa pemrograman

- Library
- Virtual machine
- File / Proses

# Kolaborasi lintas bahasa pemrograman: Library (1)

- Pada sistem operasi
  - Ekstensi nama file: .a, .so, .dylib, .dll, .lib
- Contoh:
  - Fungsionalitas tertentu dalam bahasa C
  - Dipanggil lewat bahasa Python

# Kolaborasi lintas bahasa pemrograman: Library (2)

```
library.c ✕  
1  
2 int test(int x) {  
3     return x;  
4 }  
5
```

```
gcc -c -fpic library.c  
gcc -shared -o liblibrary.so library.o
```

```
test.py ✕  
1 from ctypes import *  
2  
3 lib = CDLL('./liblibrary.so')  
4 res = lib.test(1000)  
5 print res
```

```
$ python test.py  
1000
```

# Kolaborasi lintas bahasa pemrograman: Virtual machine

- Virtual machine yang dijalankan pada sistem operasi
- Fungsionalitas tertentu dalam suatu bahasa, dan fungsionalitas lain dalam bahasa-bahasa lainnya
- Dimungkinkan selama compiler atau assembler suatu bahasa dapat menghasilkan kode yang dimengerti oleh virtual machine tersebut
  - Interoperabilitas antar bahasa seringkali tidak sesederhana ini, karena setiap bahasa dapat memiliki karakteristik yang berbeda-beda
  - Akan tetapi, hal ini umum ditemukan dalam virtual machine populer



# Kolaborasi lintas bahasa pemrograman: File / Proses

- File untuk data-interchange seperti JSON dan XML
  - Pustaka tersedia untuk berbagai bahasa pemrograman
- Serialisasi
  - Untuk yang kompatibel
- Named pipe
  - Untuk sistem operasi yang mendukung
- Jalankan program
  - Dan dapatkan outputnya

# Kolaborasi lintas platform

- Platform: dapat diartikan sebagai lingkungan dimana software berjalan
- Dalam presentasi ini, konteks platform adalah:
  - Beda arsitektur komputer
  - Beda sistem operasi
  - Antara desktop, web, dan mobile
  - Menggunakan beberapa bahasa pemrograman

# Kolaborasi lintas platform: Beda arsitektur komputer

- Walaupun umum sebuah sistem operasi mendukung beberapa arsitektur komputer → ada kemungkinan kita bekerja dengan arsitektur tertentu yang kompatibel dengan sistem operasi khusus
  - Dengan demikian, pustaka/library di level sistem operasi tidak bisa digunakan → sistem operasinya berbeda  
(Bahkan, walaupun untuk sistem operasi yang sama, karena bedanya arsitektur komputer, kompilasi ulang umumnya perlu dilakukan)

# Kolaborasi lintas platform: Beda sistem operasi

- Sistem operasi yang berbeda:
  - Pustaka/library di level sistem operasi tidak bisa digunakan
- Virtual machine tertentu yang berjalan pada berbagai sistem operasi dapat digunakan: misal Java Runtime Environment
  - Kode ditulis sekali, lalu dikompilasi, dan selanjutnya jalan dimanapun Java Runtime Environment yang kompatibel, tersedia (Write Once, Run Anywhere)

# Kolaborasi lintas platform: Desktop, web, mobile, ...

- Untuk bahasa pemrograman tertentu, seperti Java, platform bisa sangat beragam
- Java berjalan di:
  - High-performance computing
  - Server
  - Web
  - Desktop
  - Mobile
  - Embedded
  - IoT
  - ...

Mengenal dan Menggunakan Bahasa Pemrograman  
Java®, Noprianto, 2018, ISBN: 978-602-52770-0-9  
Download gratis: <https://stabilstandar.com/books>

# Kolaborasi lintas platform: Bahasa pemrograman berbeda

- Dari pengalaman, ada saja anggota tim kerja yang datang dengan latar belakang bahasa pemrograman dan platform yang berbeda

*“Lintas platform dan bahasa pemrograman –  
seberapapun usaha standarisasi dilakukan –  
terkadang tidaklah terhindarkan”*

# Protokol komunikasi

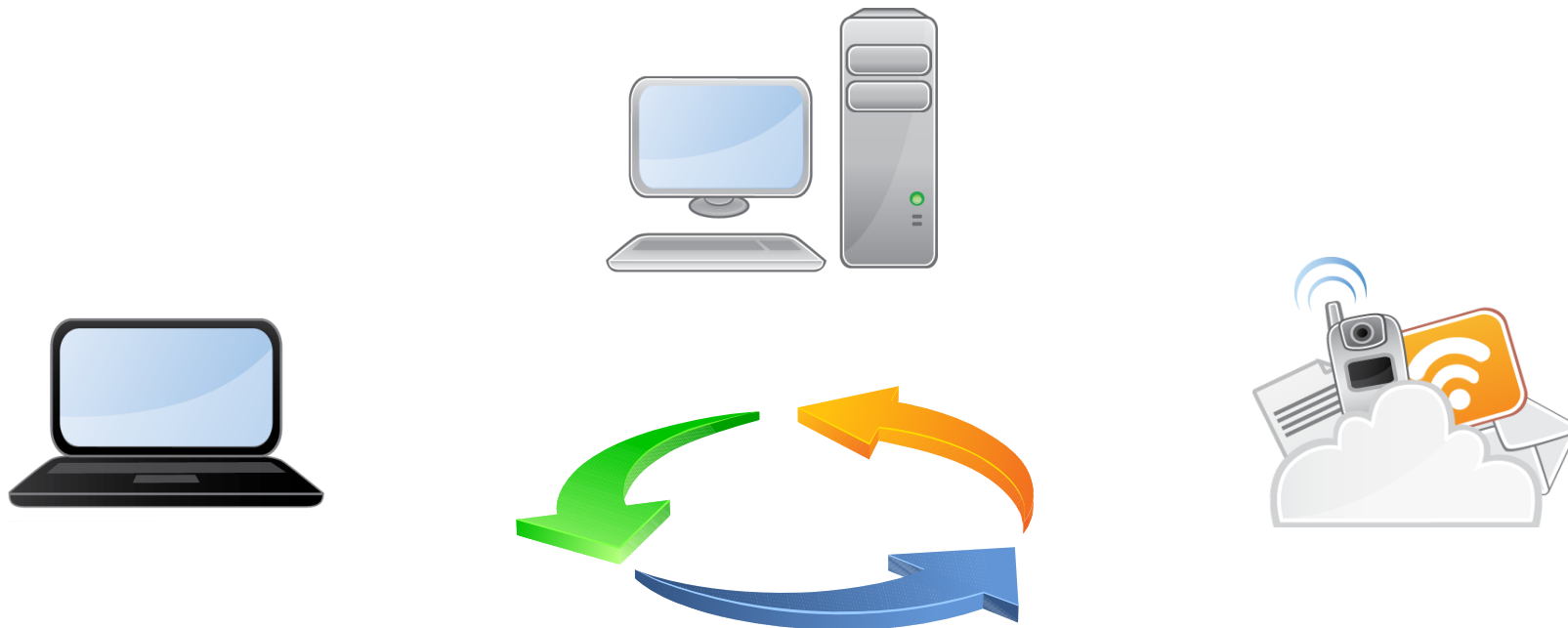
- Apapun platformnya, dikembangkan dengan bahasa pemrograman apapun, kolaborasi dimungkinkan selama protokol komunikasi tertentu digunakan
- Protokol apa yang: relatif ringan, tersedia meluas, didukung berbagai bahasa pemrograman, lintas platform, relatif sederhana, digunakan di industri, dan telah terbukti?

# HTTP

- RFC: HTTP/1.0 RFC1945, HTTP/1.1 RFC2616, HTTP/2 RFC7540
- Digunakan di World-Wide Web sejak 1990
- Tersedia meluas dan digunakan di industri




# Komunikasi lintas platform



Komunikasi  
lintas platform  
dengan HTTP

# Format pertukaran data

- Protokol komunikasi 
- Lalu, bagaimana dengan format pertukaran data?
- Format apa yang: relatif ringan dan tersedia pustaka untuk berbagai bahasa pemrograman?

# JSON

- RFC4627 dan ECMA-404 standard
- Relatif ringan / berukuran kecil
- Tersedia pustaka untuk banyak bahasa pemrograman

# Terima kasih

- Diskusi