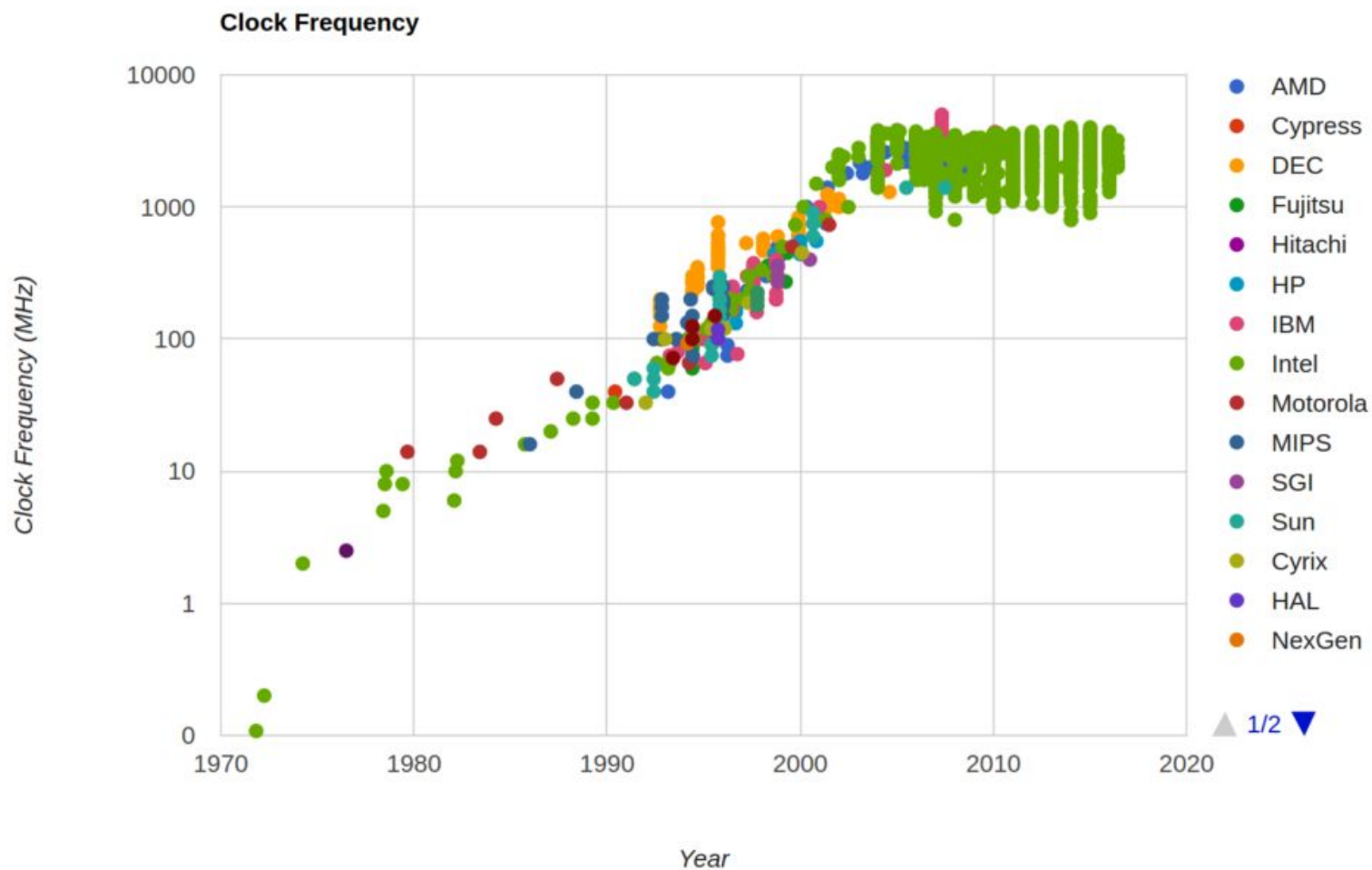


Parallel Programming

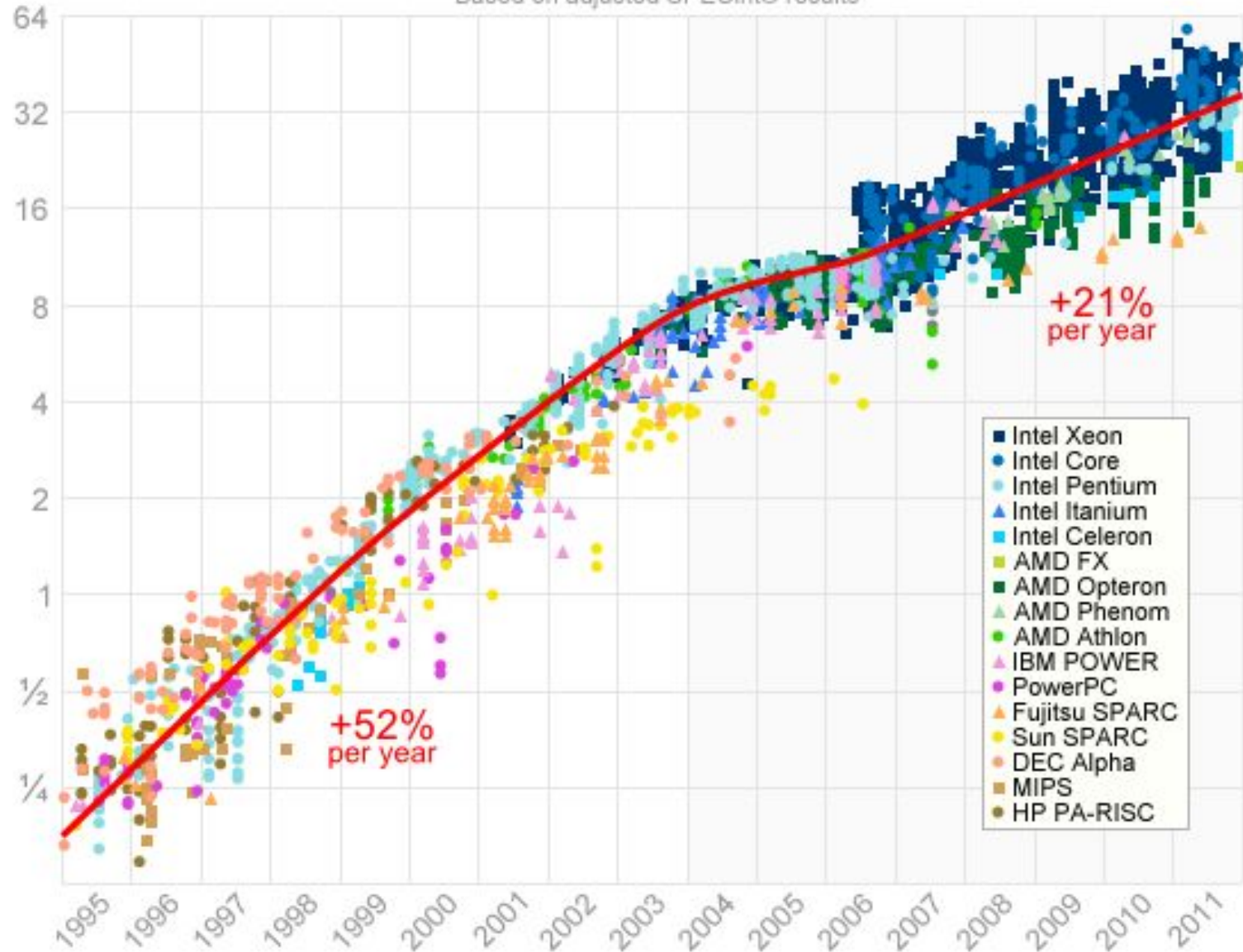
intro

Тактовая частота (clock frequency)

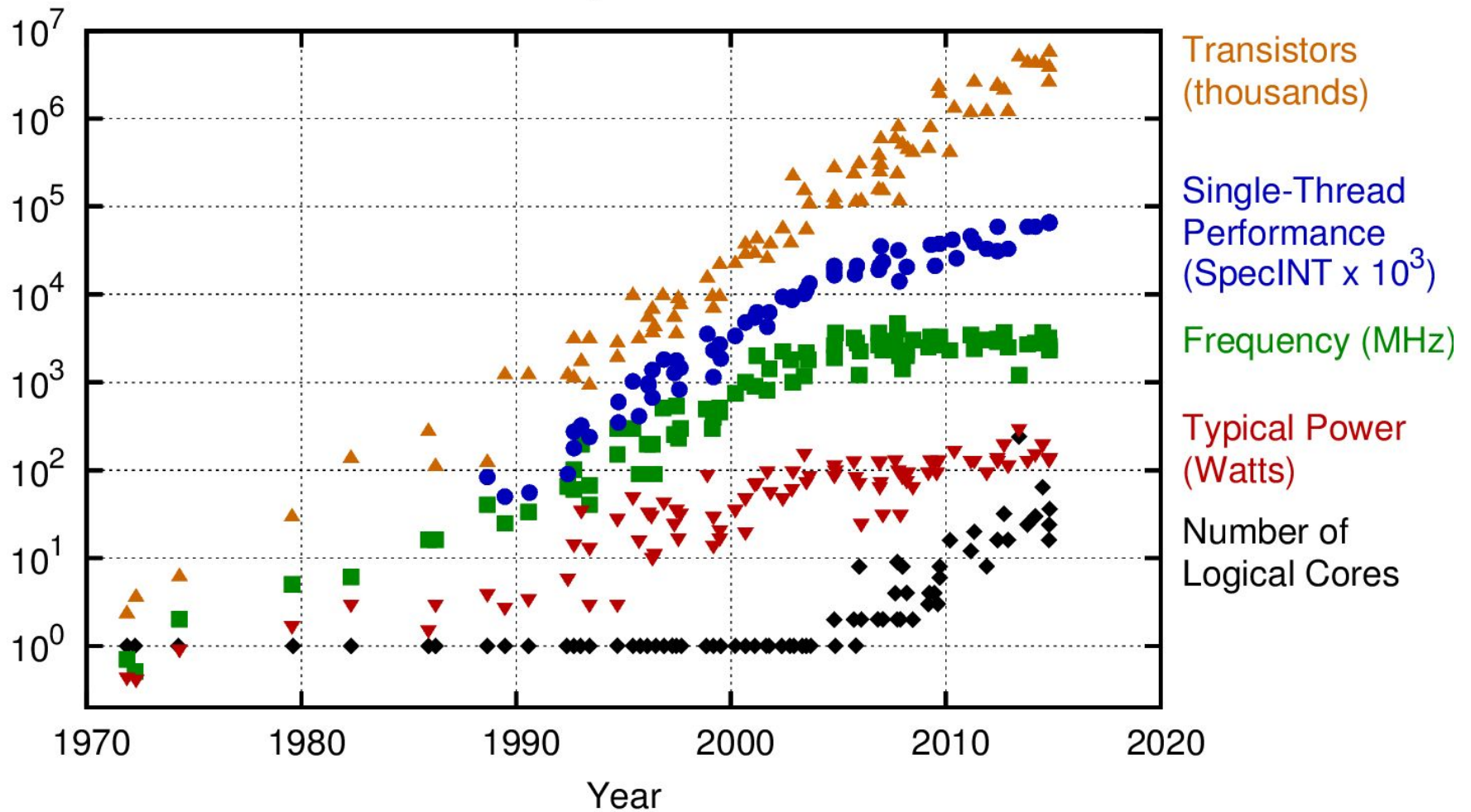


Single-Threaded Integer Performance

Based on adjusted SPECint® results



40 Years of Microprocessor Trend Data



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2015 by K. Rupp

SSE

Streaming SIMD Extensions

SIMD расширение добавляет восемь(или 16, если процессор 64-битный) 128-битных регистров XMM (XMM0-XMM7)

SSE	выполнять 70 новых инструкций, которые и работают на этих 128 битах,
SSE 2	2 64-битных числа в регистре. Включает в себя набор инструкций, который производит операции со скалярными и упакованными типами данных
SSE 3	Добавлено 13 дополнительных инструкций которые расширяют математические возможности
SSE 4	Добавлены 54 инструкции (из них ускорение видео и операции со строками), но с оговорками. В SSE 4.1 (появилась в Intel Penryn) содержится урезанный набор из 47 инструкций, а в SSE 4.2 – полный, из 54 (+7, добавлены в архитектуре Nehalem).

```
float a[4] = { 300.0, 4.0, 4.0, 12.0 };
```

```
float b[4] = { 1.5, 2.5, 3.5, 4.5 };
```

```
__asm {
```

```
    movups xmm0, a ; // поместить из a в регистр xmm0
```

```
    movups xmm1, b ; // поместить из b в регистр xmm1
```

```
    mulps xmm0, xmm1 ; // перемножить пакеты плавающих точек
```

```
    movups a, xmm0 ; // выгрузить результаты из регистра xmm0 по адресам a
```

```
};
```

Закон Амдала

(Джин Амдал сформулировал закон в 1967 году)

Какое **теоретически** потенциальное ускорение
мы можем получить от распараллеливание
нашей программы

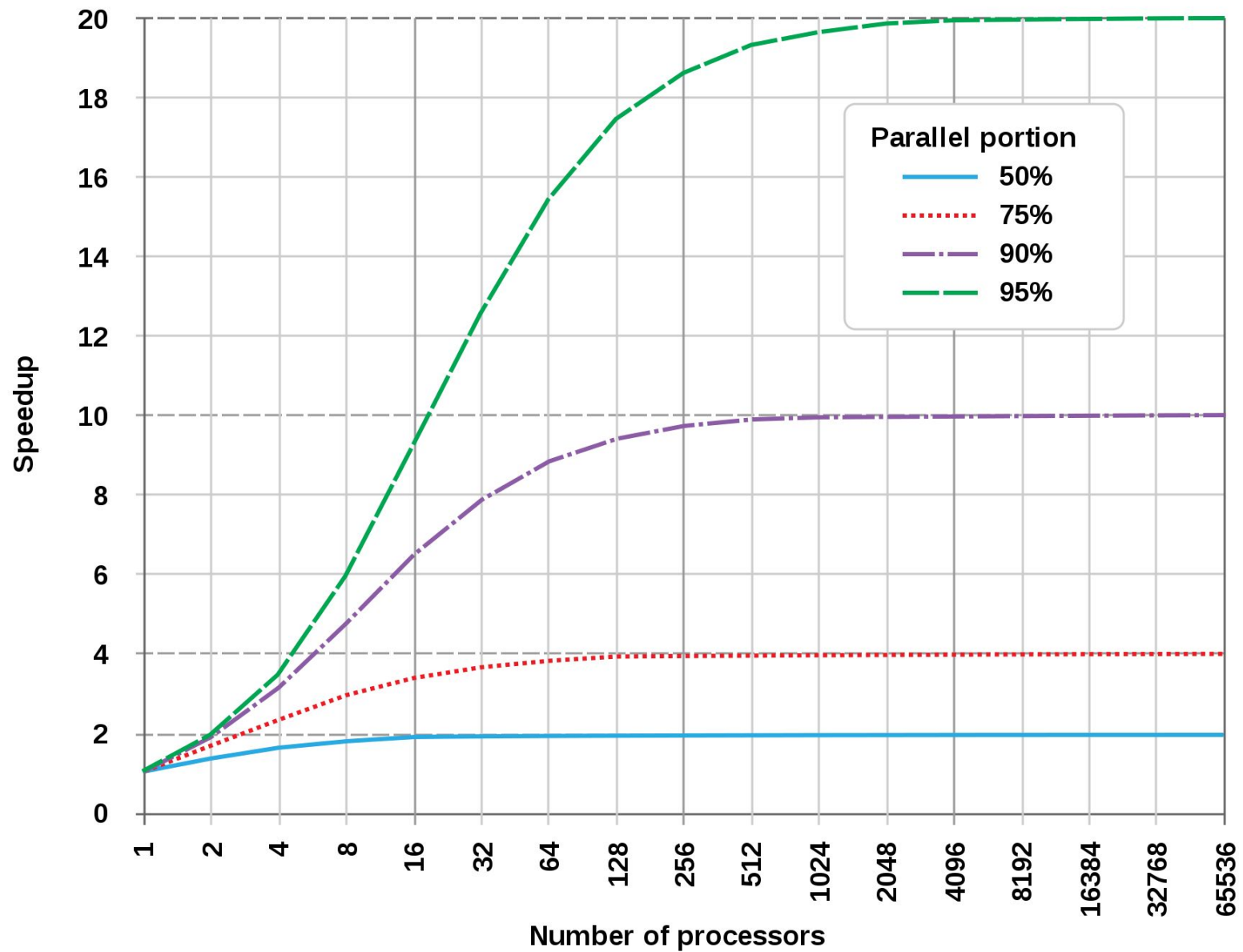
$$S_p = \frac{1}{\alpha + \frac{1 - \alpha}{p}}$$

α – доля последовательного кода
 p – количество

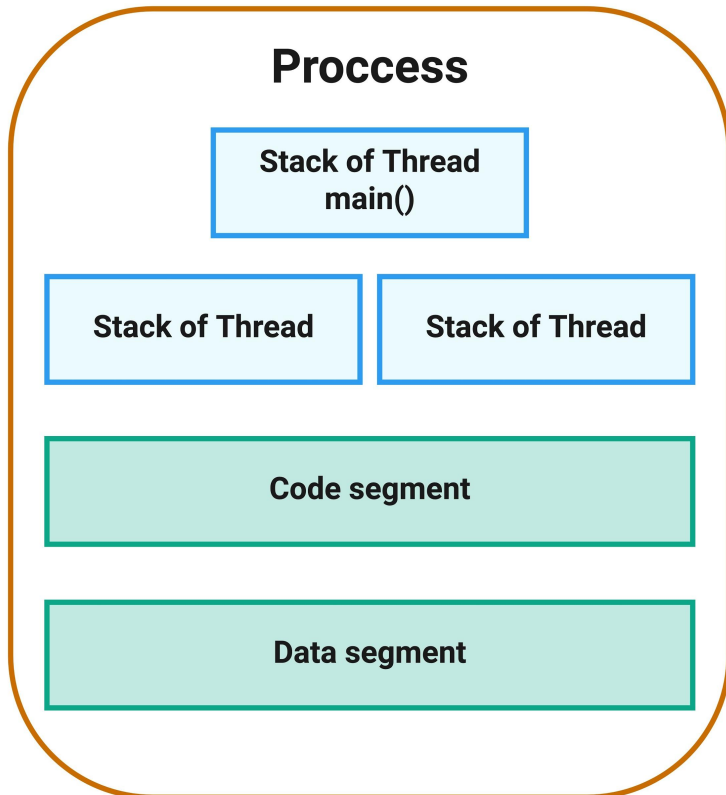
Закон Амдала иллюстрация

α/p	10	100	1000
0	10	100	1000
10 %	5.263	9.174	9.910
25 %	3.077	3.883	3.988
40 %	2.174	2.463	2.496

Amdahl's Law



Процессы и потоки



- В начале выполнения процесс — 1 поток
- Потоки могут создавать новые потоки в пределах одного процесса
- Все потоки имеют общие сегменты кода и данных
- Каждый поток имеет свой стек выполнения
- Каждый поток имеет свой идентификатор
- Единица планирования ОС - поток