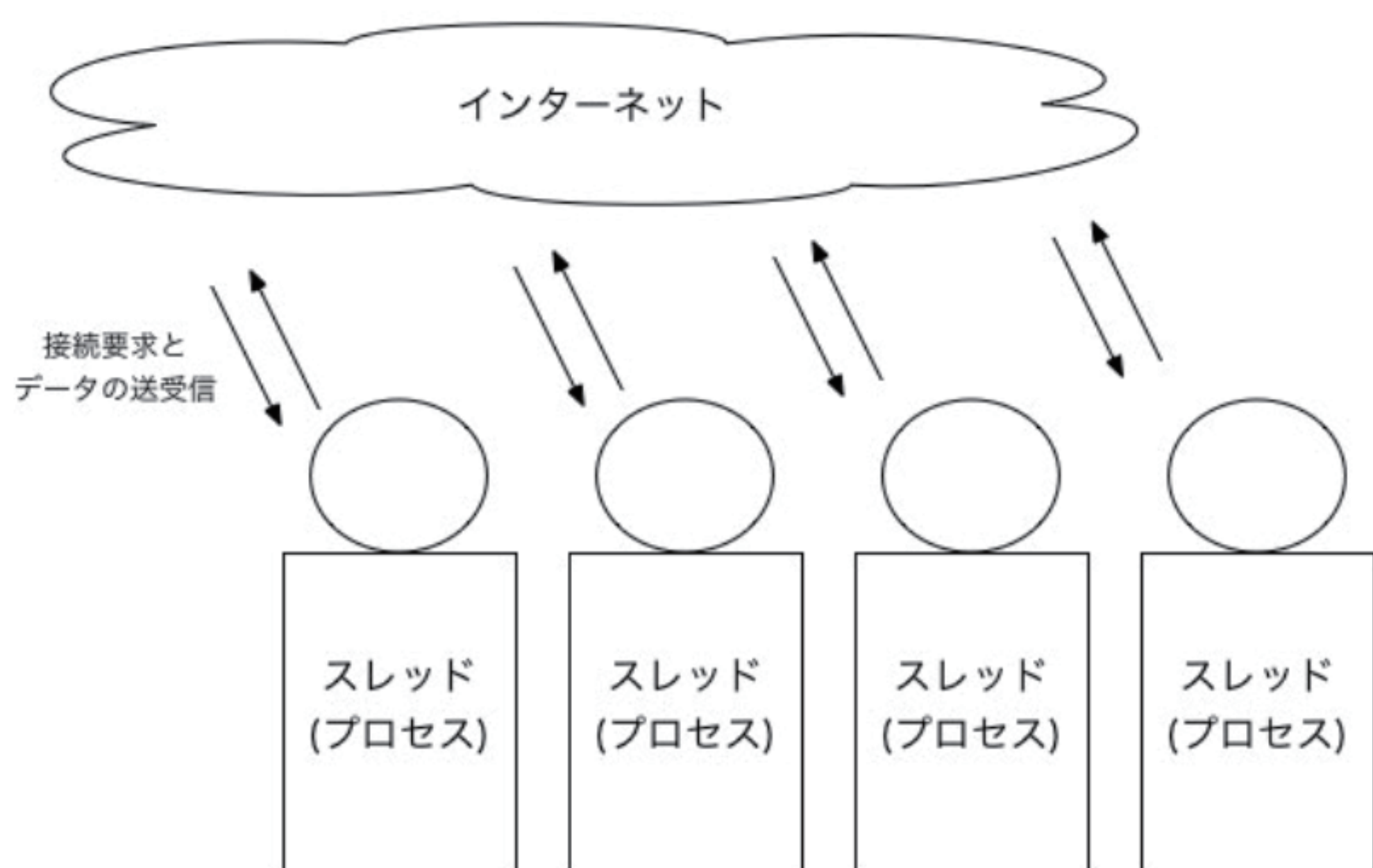


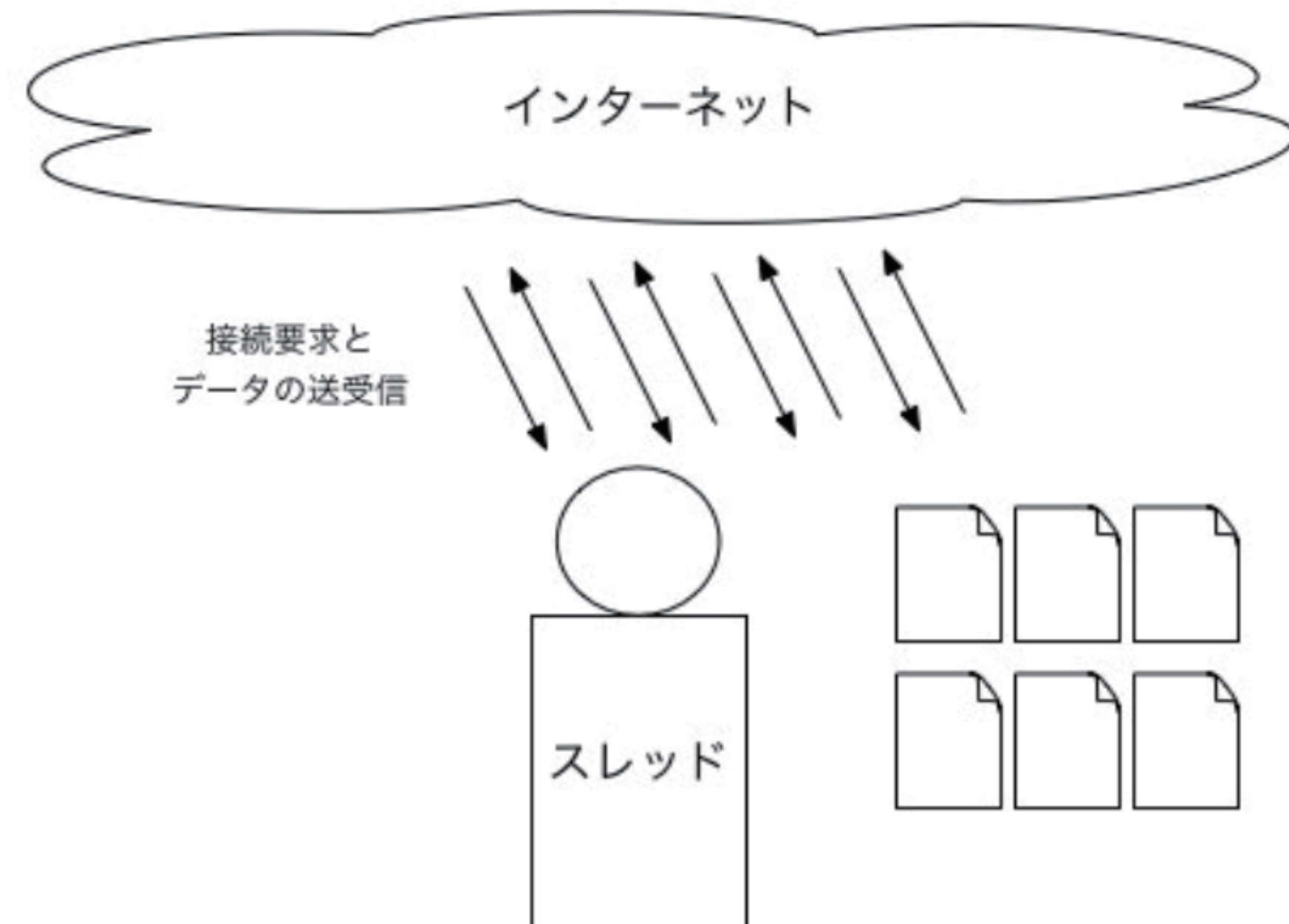
Node プログラミングモデル

Apache などの現状のウェブサーバーの問題点



- セッションごとに数 10MB 程度のスタックメモリを消費
- セッション数が極端に多くなると実メモリが不足
- 同時セッション最大数が大きく制約され、レイテンシが悪化

コールバック方式による解決



- 時間のかかる処理の前後で関数を分け、処理後にコールバック
- セッションあたり数~数百 KB 程度のメモリ消費の見込み
- 同時セッション最大数やレイテンシが大幅に改善される

C++ と Elixir で Node プログラミングモデルを実装

Zackernel の C++11 スレッドとのメモリ消費量の比較

```
volatile bool led1 = false;

void blinkLed1() {
  zLoop([&] {
    led1 = true;
    sleep(500, [&] {
      led1 = false;
      sleep(500, [&] {});
    });
  });
}
```

Zackernel のコード例

- macOS 上で評価
- mach/mach.h の task_basic_info 構造体の virtual_size で示される仮想メモリサイズを使用
- 繰返し測定して 3 回同じ測定値になった場合の値を採用
- 最小 2 乗法で傾きを求め 1 スレッドあたりの使用メモリを測定
- それぞれの相関係数は、Zackernel で 0.8396, C++ 11 スレッドで 0.99993

Num. of Tasks	Zackernel	C++11 Thread
1	0	536576
10	0	5365760
100	0	53657600
1000	2097152	537624576
2000	2097152	1094123520
5000	4194304	N/A
10000	5242880	N/A
20000	7340032	N/A
50000	10485760	N/A

Zackernel

1 スレッドあたり 204 バイト

C++11 スレッド

1 スレッドあたり約 546KB

軽量コールバックスレッド (LCB) の Elixir プロセスとのメモリ消費量の比較

```
pid = ZeamCallback.Receptor.new
send(pid, {:spawn,
  fn(tid) ->
    IO.puts "foo #{tid}"
  end})
send(pid, {:spawn,
  fn(tid) ->
    IO.puts "bar #{tid}"
  end})
```

LCB のコード例

- macOS 上で評価
- LCB もしくは Elixir プロセスを一定数新規作成する前後の全体のメモリ量の変化を計測
- 繰返し測定して 3 回同じ測定値になった場合の値を採用
- 最小 2 乗法で傾きを求め 1 スレッドもしくは 1 プロセスあたりの使用メモリ量を測定
- それぞれの相関係数は、軽量コールバックスレッドで 0.99284, Elixir プロセスで 0.99998

Num. of Tasks	LCB	Elixir Process
100	76144	358848
1000	881656	2877360
2000	1496296	5636232
5000	5734120	14160616
10000	13004640	28402248

軽量コールバックスレッド

1 スレッドあたり 1332 バイト

Elixir プロセス

1 プロセスあたり 2835 バイト

将来課題

処理系レベルから設計を見直すことで、LCB を Zackernel 並みのメモリ消費量に抑える最適化を施せる余地があるかもしれない。今後、Elixir で Zackernel 並みにメモリ消費量を抑えるべく、新しい処理系 ZEAM を開発する。