Apacheコミッターが見た、
Apache vs nginx

野村総合研究所 情報技術本部
オープンソースソリューション推進室
佐藤 高志

野村総合研究所のOpenStandia（オープンスタンディア）は、おかげさまで、2006年のサービス開始から2011年までの5年間で契約数累計が1,000件を突破いたしました！

株式会社 野村総合研究所 情報技術本部 オープンソースソリューション推進室
Mail：ossc@nri.co.jp　Web：http://openstandia.jp/

はじめに

自己紹介

Apacheとnginxの違い(1) アーキテクチャ編

Apacheとnginxの違い(2) コミュニティ編

Apacheとnginxの違い(3) 機能、その他編

さいごに

おことわり

- 今日は別にApacheが優れていると言いに来たわけではありません。むしろnginxのほうが…的な話ばっかりかも。
- nginxについては、Apacheに比べると知識はものすごく薄いので、変なことを言っていても優しくしてください。

- 正式名称はApache HTTP Serverですが、長いので今日はApacheと呼びます。
- 開発コミュニティを指す場合Apache Software Foundationと呼びます。

自己紹介

- **氏名：佐藤高志** @tksst1024
- **所属：株式会社野村総合研究所**
  情報技術本部
  オープンソースソリューション推進室
  オープンソースのサポートサービスやソリューションを提供。
  Apacheのサポートは私もやっています。
- Apache HTTP Server **コミッター**
  - **が、最近はあまり参加しているとは言えないか…**

自己紹介 コミッターとは

- オープンソース開発コミュニティにおいては通常、バージョン管理システム(Subversionなど)のリポジトリはインターネットで読み取りが公開されている
- 誰でも書き込みアクセス(=コミット)可能にしてしまうと大変なので、コミッターに制限されている
- コミッター以外は、ソースコードの貢献にはコミッターを通す必要がある
- Apache Software Foundationにおいては、各プロジェクト運営委員会(PMC)が、よくプロジェクトに貢献している人について、「コミッターになってもらったほうがプロジェクトにとって良い」と判断すると、コミッターへの勧誘メールが来る。

自己紹介

- コミッターではありますが、今まであまり大規模な新機能の開発などはしたことがありません。
- ほとんどがバグ修正や、新機能といっても数行の修正で済むようなものばかりです。
- それと、報告されたバグについてコメントしたり。
- そんな地味な活動でも、貢献が認められて、コミッターになれたりします。

ApacheとnginxのIの違い(1) アーキテクチャ編

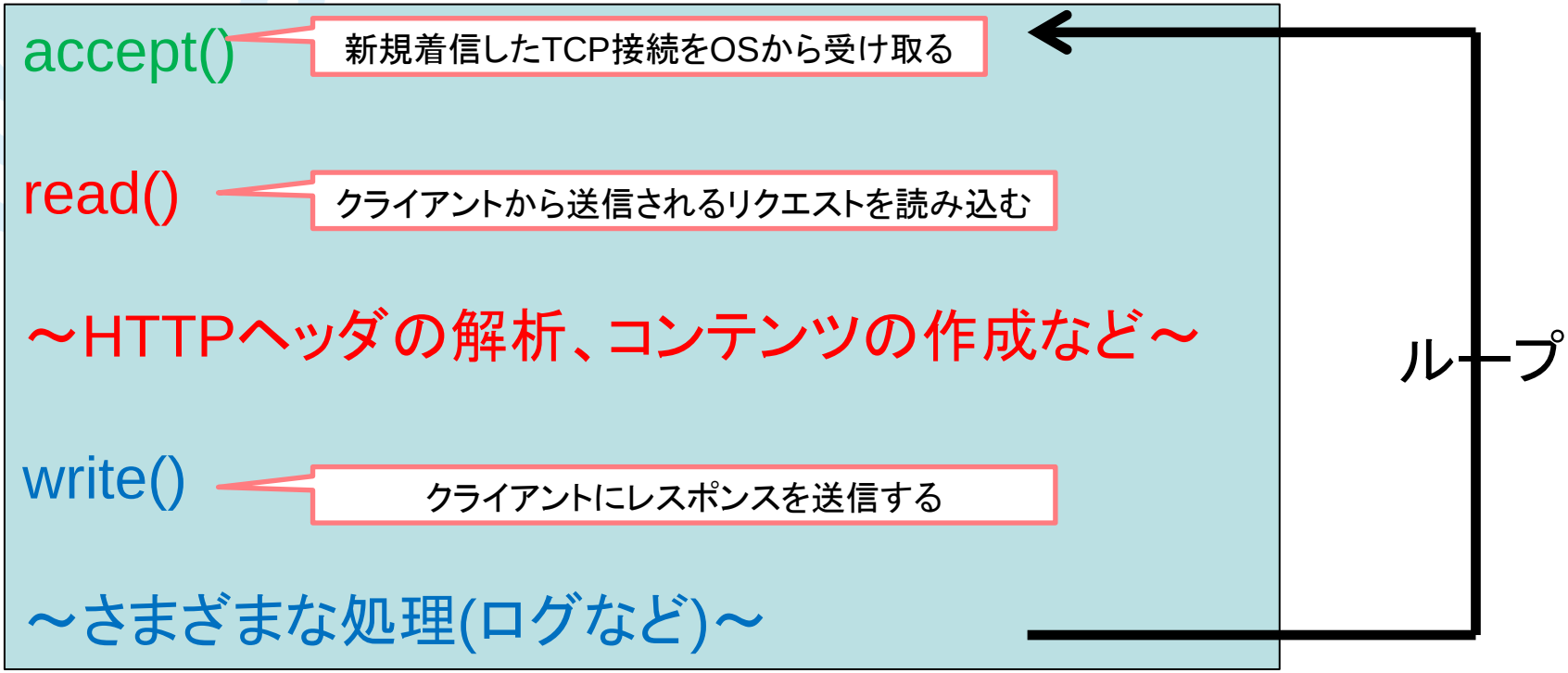
- Apacheとnginxの違いを、アーキテクチャの観点で話します。
- 対象：
  - HTTPの仕組みざっくりイメージできる人(テキストベースとか、リクエストしてレスポンスが返ってきているとか)
  - WebブラウザでURLにアクセスすると何か表示される、という理解だけだと厳しいかもしれません。
  - 例えば以下のキーワードの意味の違いを知っているような人は、ごめんなさい、簡単すぎると思います。
    - ✓非同期I/O、ノンブロッキングI/O、I/O 多重化

ApacheとnginxのApacheの違い(1) アーキテクチャ編 マルチプロセス・マルチスレッド

- Apacheはマルチプロセスまたはマルチスレッドのアーキテクチャです。(event MPMを除く)
  - 複数のアクセスに対応するために、プロセスまたはスレッドを使用しています。

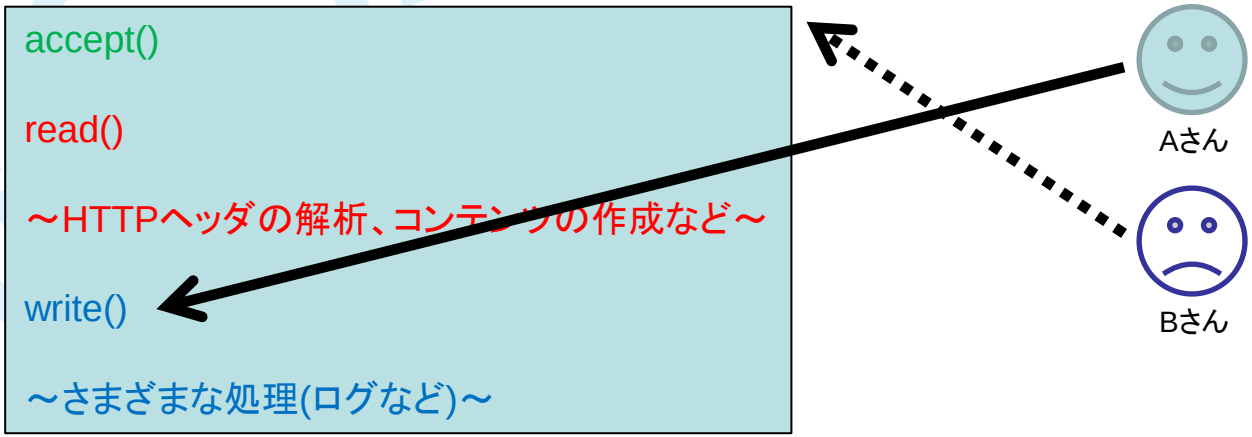
Apacheとnginxの違い(1) アーキテクチャ編
マルチプロセス・マルチスレッド

- **超ざっくりなHTTPサーバ実装：**



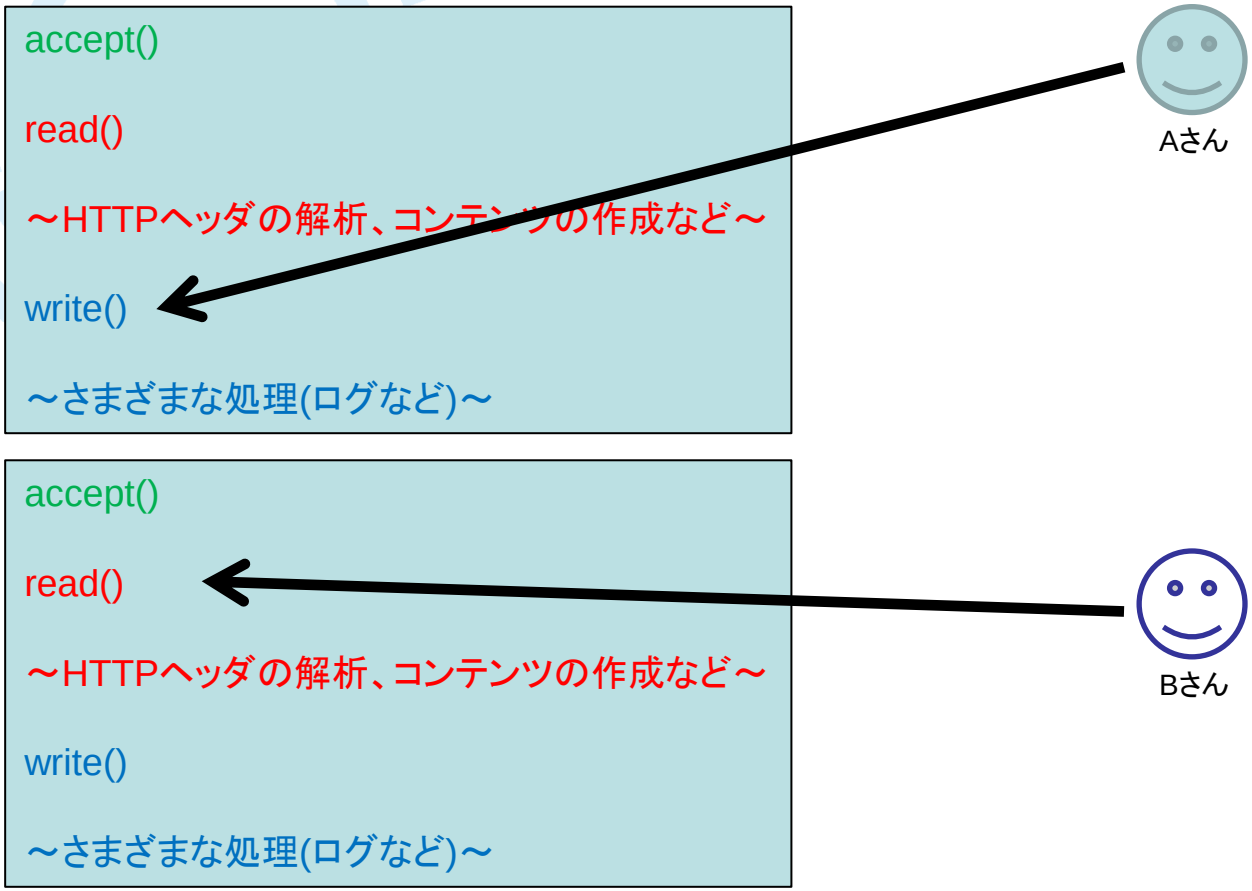
- **上記をお好きな言語で実装すれば、とりあえずHTTPをしゃべるサーバーになります。**
- accept(), read(), write()**はソケット(ネットワーク)に関するシステムコール(OS機能呼び出し)です。各言語に対応するメソッドなどがあるはずです。**

Apacheとnginxの違い(1) アーキテクチャ編
マルチプロセス・マルチスレッド



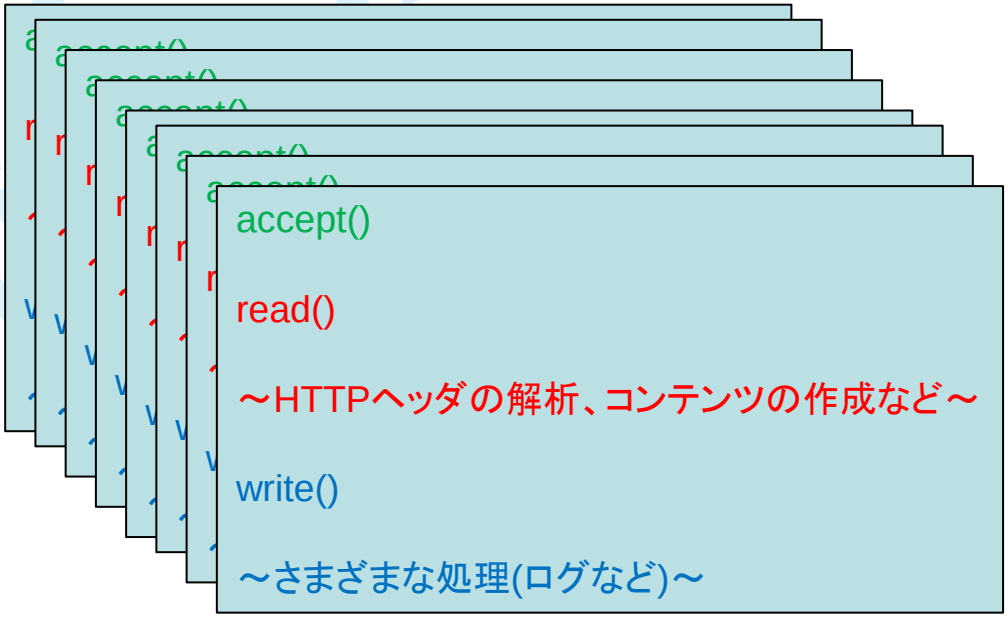
- TCP接続の着信を取り出して処理を開始するのは、accept()です。
- Bさんがブラウザで上記サーバーにアクセスしましたが、このサーバーは既にAさんが先にアクセスしており、ちょうどAさんにレスポンスを返しているところでした。Aさんの処理(上のブロック全部)が終わるまで、accept()が呼ばれないため、他の人は待つことになります。
- 0.1秒とかで終わればまだ良いですが、10秒とかかかっていたらBさんは不満ですよね。
- つまり、このサーバーは複数接続を同時に扱えません。

Apacheとnginxの違い(1) アーキテクチャ編
マルチプロセス・マルチスレッド



- そこで、上記のブロックをもう1個用意しましょう。1個目がAさんの対応をしながら、もう1個がaccept()を呼び出し、Bさんの対応を開始します。

Apacheとnginxの違い(1) アーキテクチャ編
マルチプロセス・マルチスレッド



- プロセスやスレッドを使って複数の処理者を作り出すことで、同時アクセスに対応。
  - 処理者の数 = 同時アクセス可能数
- 実行単位としてプロセスを使うか、スレッドを使うかという違いはあるものの、処理者を複数走らせるという点で変わりはない
- ApacheのpreforkMPM、workerMPMは上記のように、ブロックを複数個(=MaxClients)持つことによって、同時アクセスに対応している。

Apacheとnginxの違い(1) アーキテクチャ編
マルチプロセス・マルチスレッド

- 利点
  - シンプルでわかりやすいアーキテクチャ
- 欠点
  - プロセスもスレッドも、それ自体のオーバーヘッドが大きい
    - ✓メモリ
    - ✓CPUのコンテキストスイッチ
  - ⇒大きな同時接続数に対応しようとすれば、それだけ上記負荷が増える

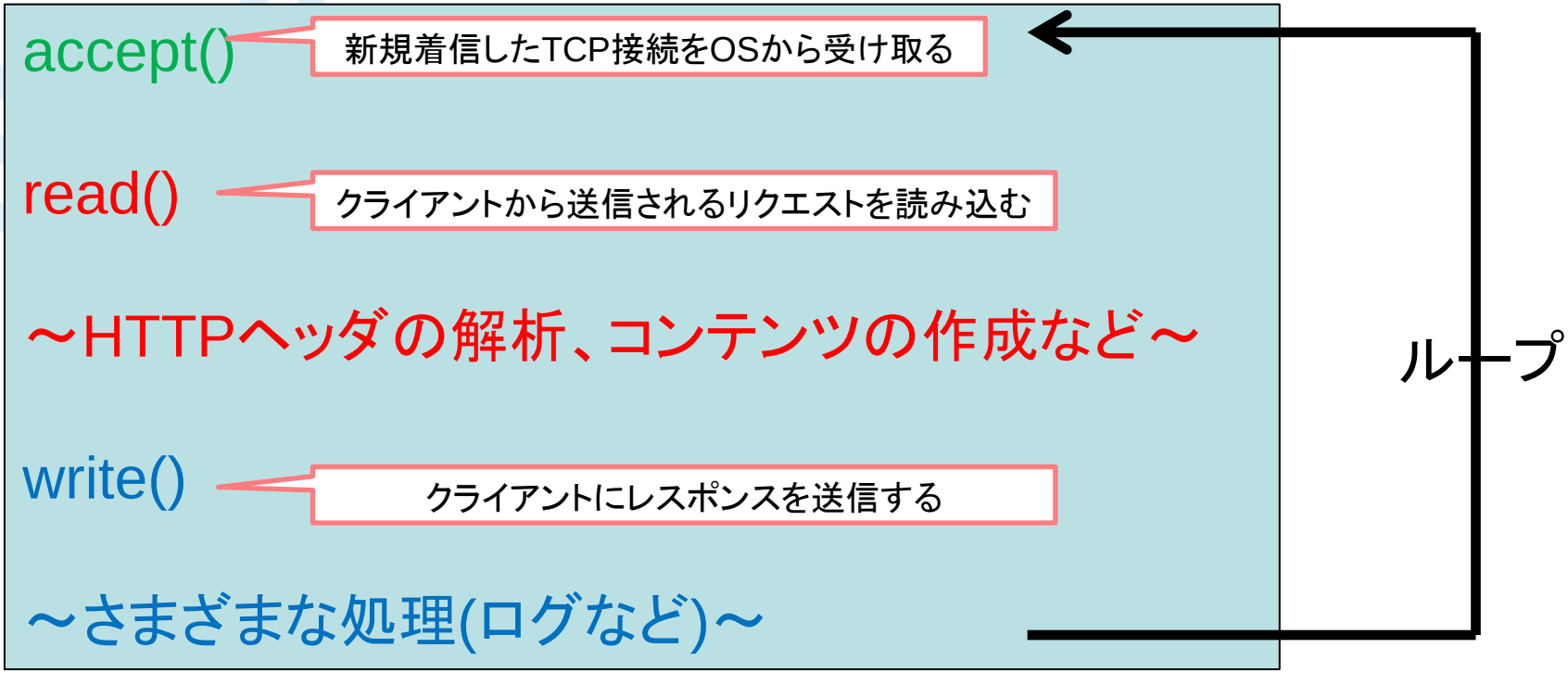
ApacheとnginxのIの違い(1) アーキテクチャ編
イベント駆動

- nginxは**イベント駆動**のアーキテクチャです。
  - 先ほど説明した「処理者」が1人で複数の接続を扱います。
  - これはどういうことでしょうか。

- なお、ApacheもEvent MPMもどちらかといえばイベント駆動です。
  - なぜ「どちらかといえば」なのかは後ほど。

ApacheとnginxのIの違い(1) アーキテクチャ編
イベント駆動

- 再び、超ざっくりなHTTPサーバ実装：



- 上記を複数のブロックに分割してみましょう。

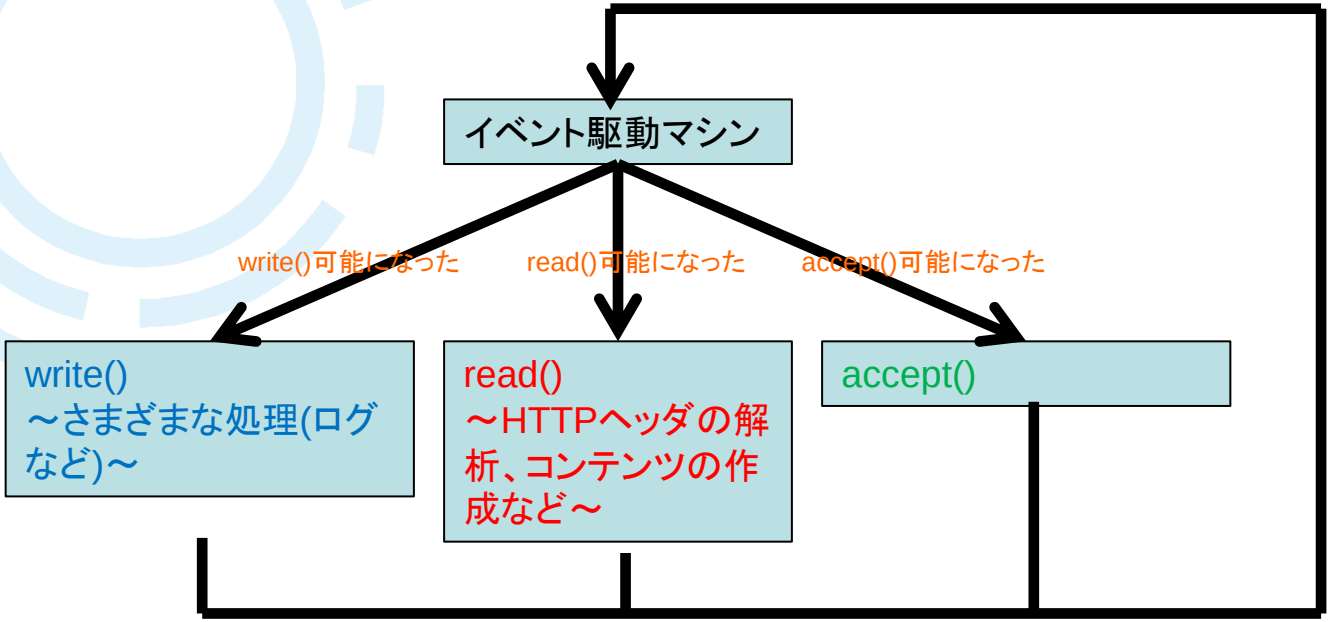
ApacheとnginxのIの違い(1) アーキテクチャ編
イベント駆動

- 下記のように分けてみます。

write() ～さまざまな処理(ログなど)～	read() ～HTTPヘッダの解析、コンテンツの作成など～	accept()

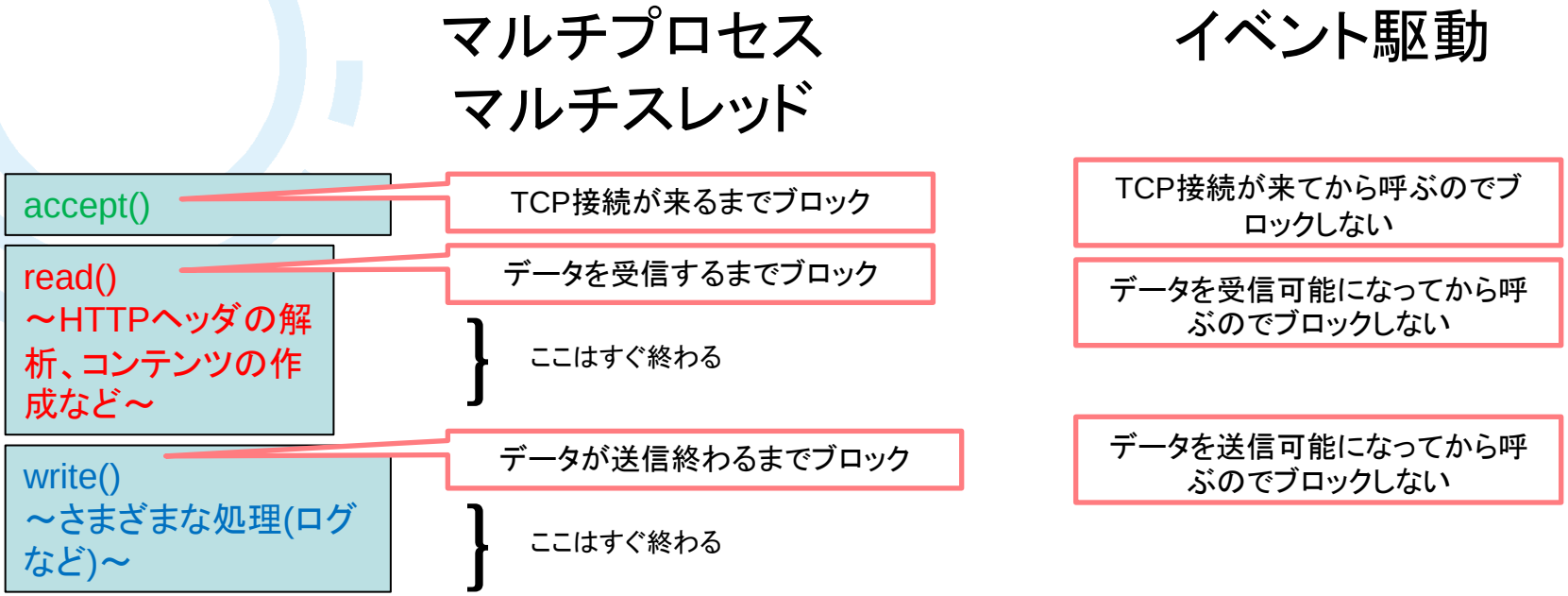
- どれもブロックの最初には、ネットワークI/Oを行なうシステムコールがあります。
- 単に分けただけでは、プログラムとして動作しません。

ApacheとnginxのApacheとnginxの違い(1) アーキテクチャ編
イベント駆動



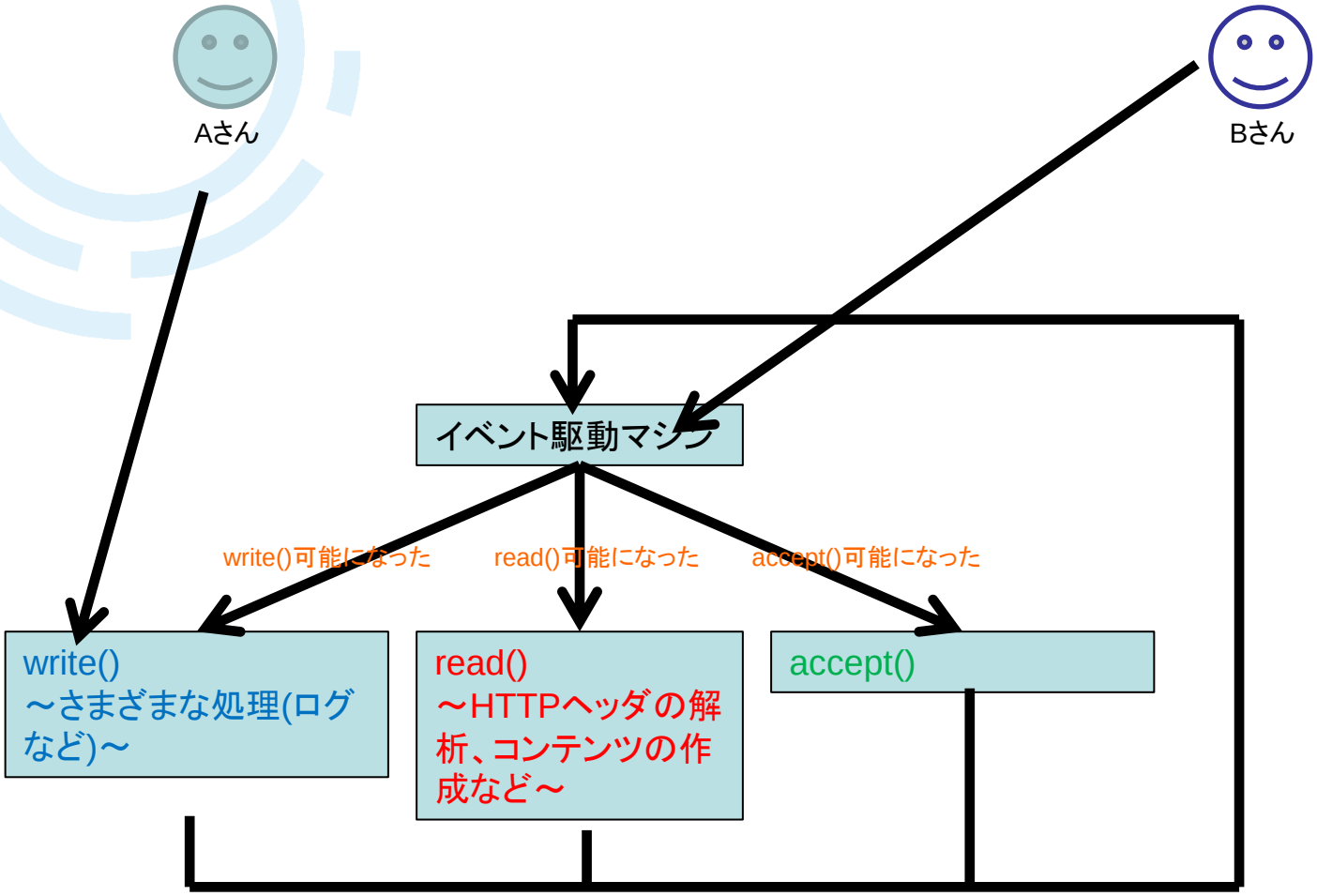
- TCP接続の状態を調べて、可能になった処理を呼び出す「イベント駆動マシン」を用意します。
- 各処理の後、イベント駆動マシンに戻ります。
- 事前に調べるので、各ブロックのread(), write(), accept()で処理は止まりません(ブロックしない)。
- イベント駆動マシンは同時に複数個のTCP接続を監視することができます。
  - 複数接続を同時に扱える！

Apacheとnginxの違い(1) アーキテクチャ編
イベント駆動



- 各ブロックの処理の間は他の接続を処理できない
- しかし、上記の通り、各ブロックの処理はイベント駆動の場合すぐに終わるため問題ない

ApacheとnginxのApacheとnginxの違い(1) アーキテクチャ編
イベント駆動



ApacheとnginxのApacheとnginxの違い(1) アーキテクチャ編
イベント駆動

利点

- 同時接続数が増えてもプロセスやスレッドは増えないため、オーバーヘッドも増えない
  - ✓⇒性能が高い！

欠点

- 処理の流れを淡々と並べるマルチプロセス・マルチスレッドに比べて、イベントをばらばらと書くため、ソースコードが複雑でわかりにくいという意見がある

Apacheとnginxの違い(1) アーキテクチャ編
イベント駆動

- **実は説明不十分なところ、というか嘘があります。**
  - 「ここはすぐ終わる」としたところは本当？

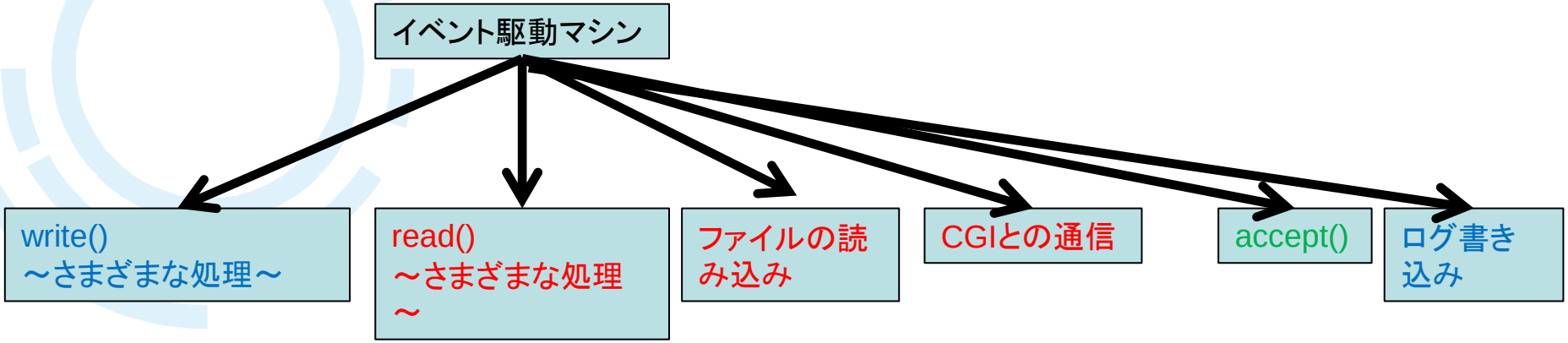
read()
～HTTPヘッダの解析、コンテンツの作成など～

} ここはすぐ終わる？

- **静的ファイルならディスクから読まなきゃ。プロキシの場合はバックエンドサーバーと通信しなきゃ。CGIとかPHPとかJavaのアプリケーションの場合、DBやファイルアクセスがあるかも知れない。**

- **「すぐ終わる」が崩れると、それだけ同時アクセスを阻害します。**

Apacheとnginxの違い(1) アーキテクチャ編
イベント駆動



- nginxはあらゆる時間のかかる(ブロックが起こる)処理をイベント駆動しています。
- ApacheのEventMPMは残念ながら、イベント駆動化されているのはクライアントとのネットワークI/Oだけです。そのため、上記自体を複数持つことで対応しています(マルチプロセス・マルチスレッドとイベント駆動のハイブリッド)

- ⇒nginxのほうが優れていると考えられる

ApacheとnginxのIの違い(1) アーキテクチャ編
落ち穂拾い

- 「イベント駆動は処理者一人」と言ったが、処理者は1つのCPUしか使用できないため、実際はCPU数に合わせてイベント駆動のスレッドを複数用意することが多い。
- イベント駆動型において「read()可能かどうか確認してからread()する」というような説明をしました。これはI/O多重化と呼びます。イベント駆動型を実装するにはこれ以外に非同期I/Oなどもありますが、今回は詳しく説明しません。nginxは非同期I/Oも使用しています。

ApacheとnginxのWebサーバとしての違い(2) コミュニティ編

- ApacheとnginxはともにオープンソースのWebサーバです。
- しかし、その開発体制は大きく異なります。

Apacheとnginxの違い(2) コミュニティ編

- Apache
  - **開発元：** Apache Software Foundation ⇒**コミュニティ**

- **営利企業ではなくコミュニティにおいて、開発意欲のある人がその人の意思で開発に参加している。**
- **コミッターやメンバー(＝貢献度の高いコミッター)になる方法も明記されている。**
- **誰でも参加できる公開メーリングリストで開発に関する議論を行なうのが原則。あなたも明日から意見を投稿できる。**

ApacheとnginxのNginx違い(2) コミュニティ編

- nginx
  - 開発元：Nginx, Inc. ⇒営利企業

- 営利企業による開発。
- 私が調べた限り、一般の開発者はコミット権を持たず、全てNginx, Inc.の社員による開発のように見える。
- コミッターは全員社員のように見える。少なくとも、コミッターになる方法は説明されていなさそう。
- もちろん、一般からの参加を拒んでいるわけではない。メーリングリスト・バグ管理システムは公開されており、パッチも受け付けている。ドキュメントについては完全にコミュニティベース。
- ものすごく開発に貢献すれば、会社から声がかかったりするのだろうか？
- 商用サポート購入者のみが導入できる機能が提供されている。

ApacheとnginxのApacheの違い(2) コミュニティ編

- どちらもオープンソースであり、一般からの参加も受け付けているので、十分オープンだと思います
- ただ、私は何となくApacheのようなコミュニティ主体のほうが好きです。

ApacheとnginxのApacheの違い(3) 機能、その他編

機能

- 両者ともおおむね同じような機能をもっている
- しかしnginxのほうが後発な事もあるのか、Apacheにない視点での機能がある。例えば、動画ストリーミング機能が充実している。
- 他には例えば、Apacheの標準にはない、SPDY対応がnginxにはある。
- Apacheはマニアックなところまでをモジュール化をしている印象
  - ✓例えば認証情報のデータストアやロードバランシングアルゴリズムがモジュール化されている
- nginxは無停止で設定変更やバージョンアップができる機能がある。
  - ✓設定変更であればApacheでもgraceful restartで対応出来るか…

ApacheとnginxのNの違い(3) 機能、その他編

- ソースコードの読みやすさ(完全に個人の感想です)
  - 初めて読むならば、正直nginxのほうが読みやすいのではないかと…
  - Apacheは非常に長かったり、ネストしすぎな関数がちらほらあったりしますし、API仕様が不明確だったり、構造体を直接書き換えさせることがしばしばあります。

さいごに

- なんだか、nginxのほうが優れているとばかり言ったような気がします。。。
- nginxは後発な事もあって、やはりよく考えられて作られています。
- Apacheは最初にマルチプロセス型として作られたので、各開発者の努力を見ていると、イベント駆動型への転換はとても大変なことに思えます。
- nginxのシェアが今後どんどん増えていき、いつかApacheを逆転する日が来る可能性は否定できないと思います。
- 一方Apacheもバージョン3.0構想がちょいちょい開発メーリングリストで出てきていたりします。まだまだApacheもがんばれると思います。

- **OpenStandiaは、「攻めのIT」を支援します。**
- **オープンソースのことなら、なんでもご相談ください！**

オープンソースまるごと

OpenStandia™

Open Source Technology

お問い合わせは、NRIオープンソースソリューション推進室へ

ossc@nri.co.jp

http://openstandia.jp/

NRI 未来創発

Dream up the future.