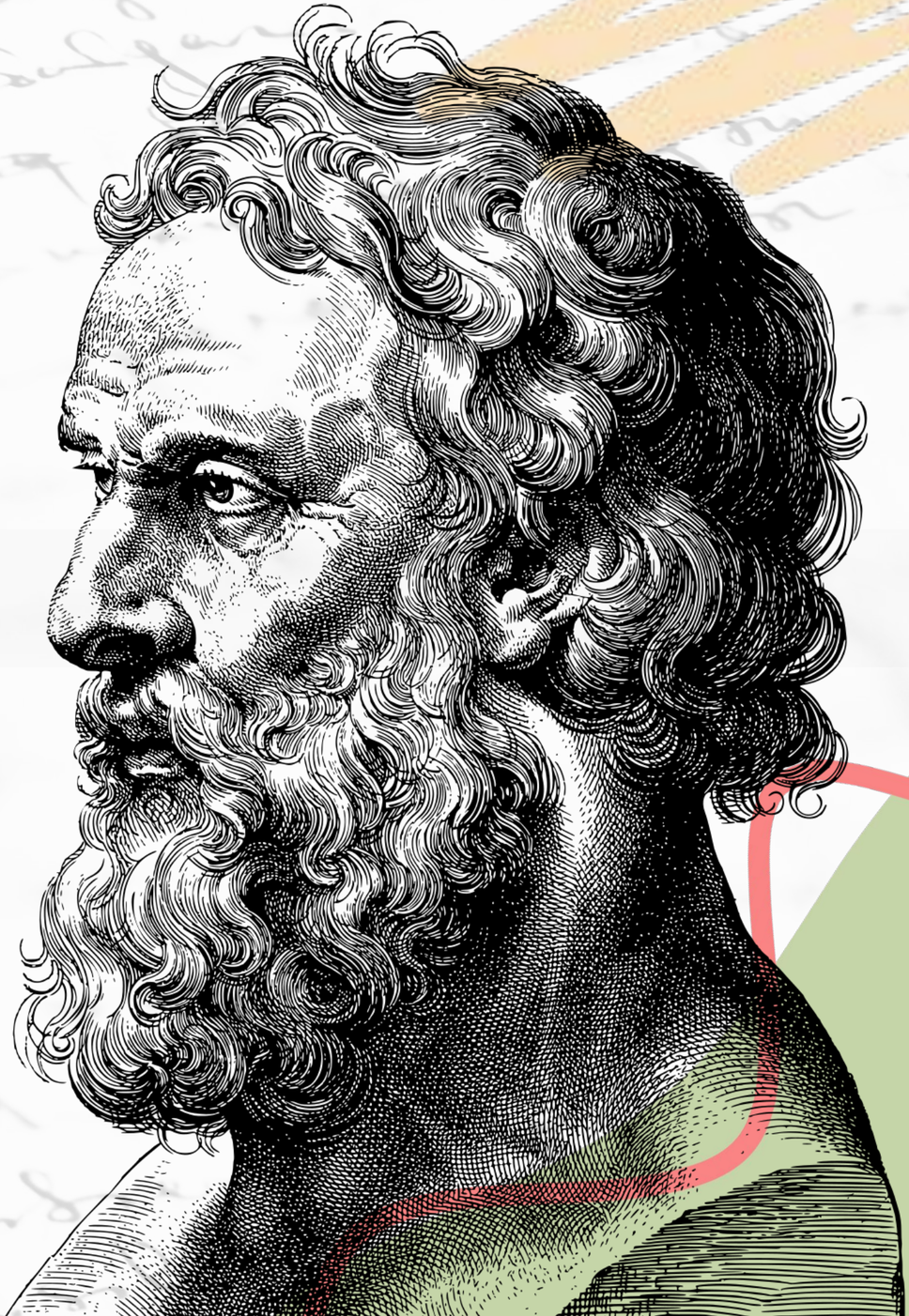
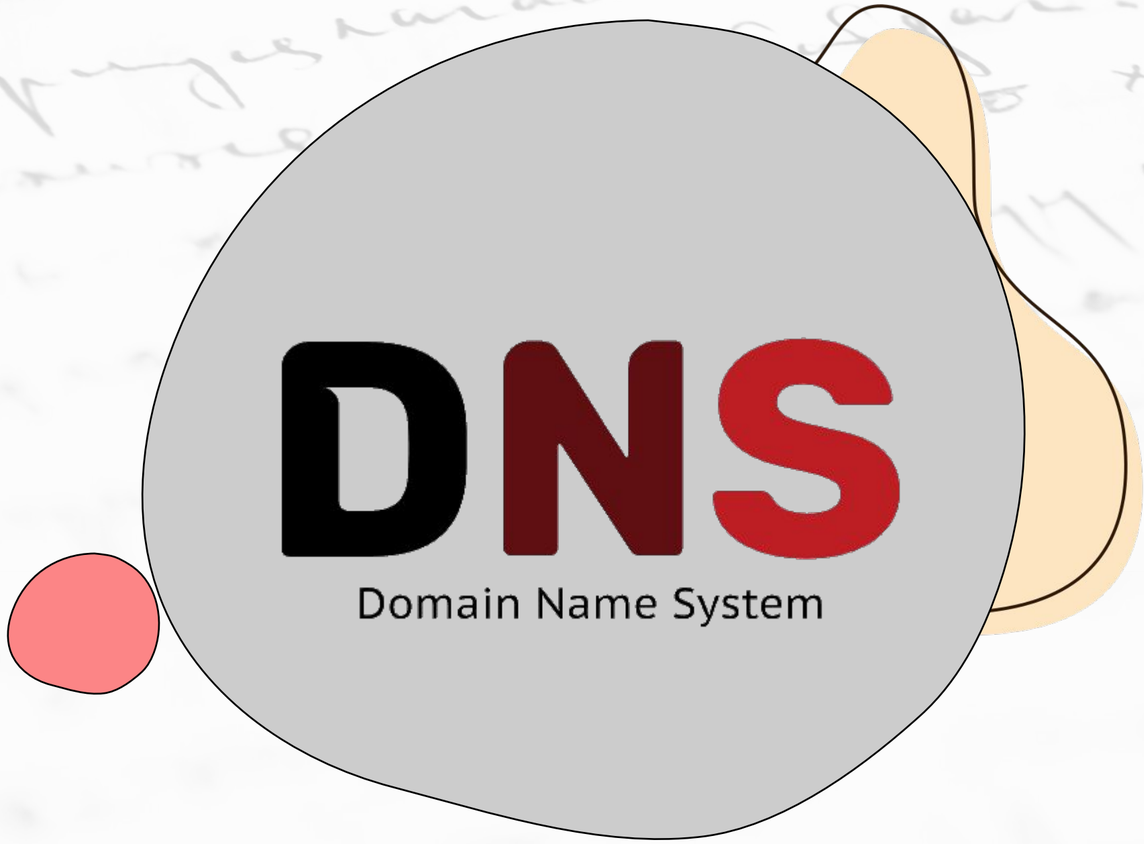
DNS

わかんねえ……





ドメインと IP アドレスの変換をするやつ

例：beyondjapan.com ＝ 104.21.91.53

例えるなら電話帳みたいなやつ（電話帳は名前と電話番号を変換する）

わかんねぇポイント

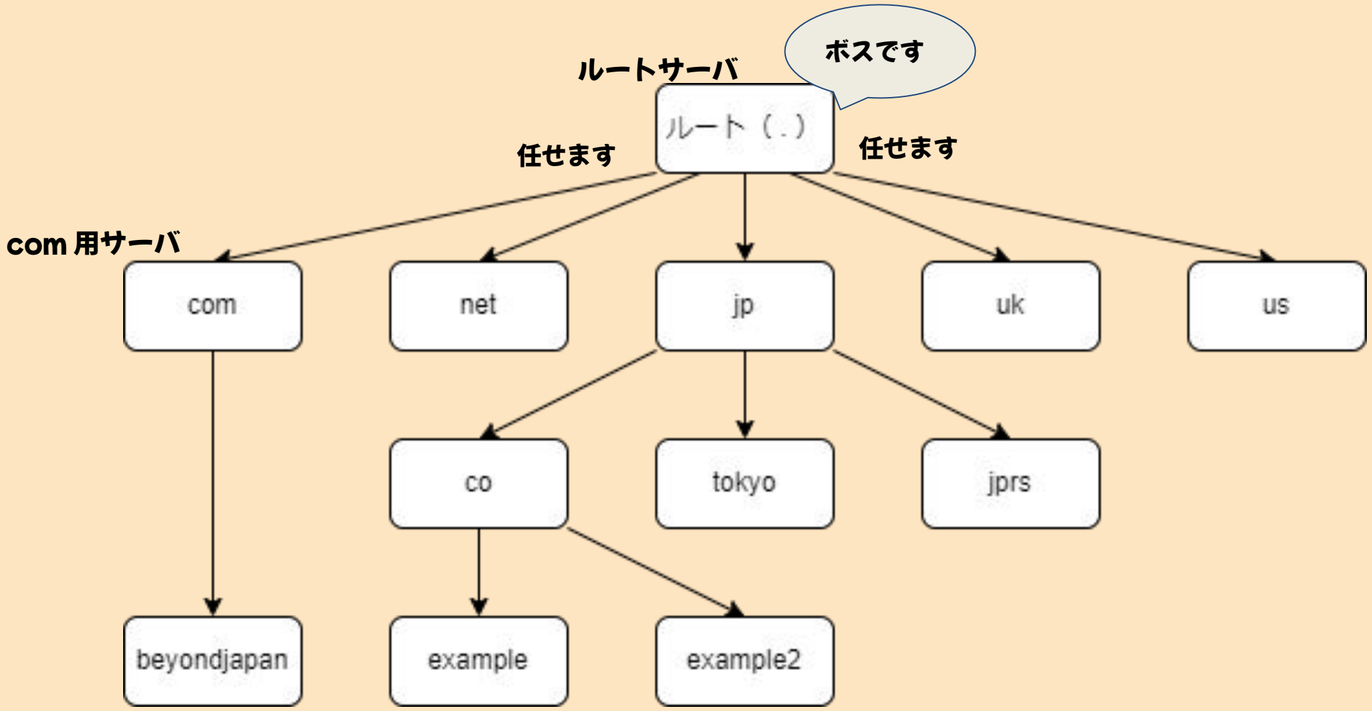
分散しすぎ

別名多すぎ

TTL
ややこしすぎ

分散しすぎ

DNS は分散型 DB みたいなもん



- それぞれのドメインを管理している場所が違う（情報が 1 か所に集中していない）
- 上から下に「このドメインについてはお前に任せた」と権限を渡している

```
$ dig beyondjapan.com

;; ANSWER SECTION:
beyondjapan.com.      300    IN    A     172.67.167.78
beyondjapan.com.      300    IN    A     104.21.91.53
```

ドメインの最後にルート（.）ついてる

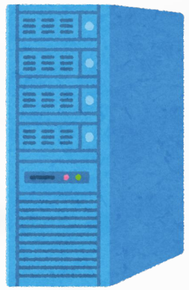
Q. ドメインの問い合わせをすると
どのように答えが得られるのか

ぼく

スタブリゾルバ

① beyondjapan.com くれ

⑧ 104.21.91.53 だってよ



フルサービスリゾルバ

②知ってる？

③今は com 用サーバに任せとんねん

④知っとるって聞いたで

⑤今は A に任せとんねん
そっちに聞いてや

⑥おしえてー

⑦104.21.91.53 やで

ルートサーバ

com 用サーバ

権威サーバ A

A. たらいまわし

ふたたびアクセスしたときの動き

ちょっとまえに聞いたとき
104.21.91.53って言ってたなぁ

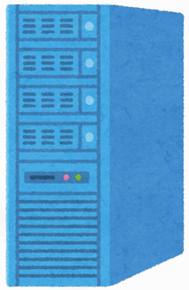
ぼく

スタブリゾルバ

① beyondjapan.com くれ

② 104.21.91.53 です!

フルサービスリゾルバ



キャッシュします

DNS 問い合わせを追いかけたい

dig コマンドでいろいろ確認できる

大事なのは「@」と +norec オプション

ふつうに dig ると下から上へお伺いを立てる感じ
ブラウザからアクセスしたときもこうなる

スタブリゾルバ

① beyondjapan.com くれ

⑥ 104.21.1.53 だってよ

フルサービスリゾルバ

②キャッシュあるわ
com 用サーバに聞こ

③キャッシュあるわ
A サーバに聞こ

ルートサーバ

com 用サーバ

④おしえてー

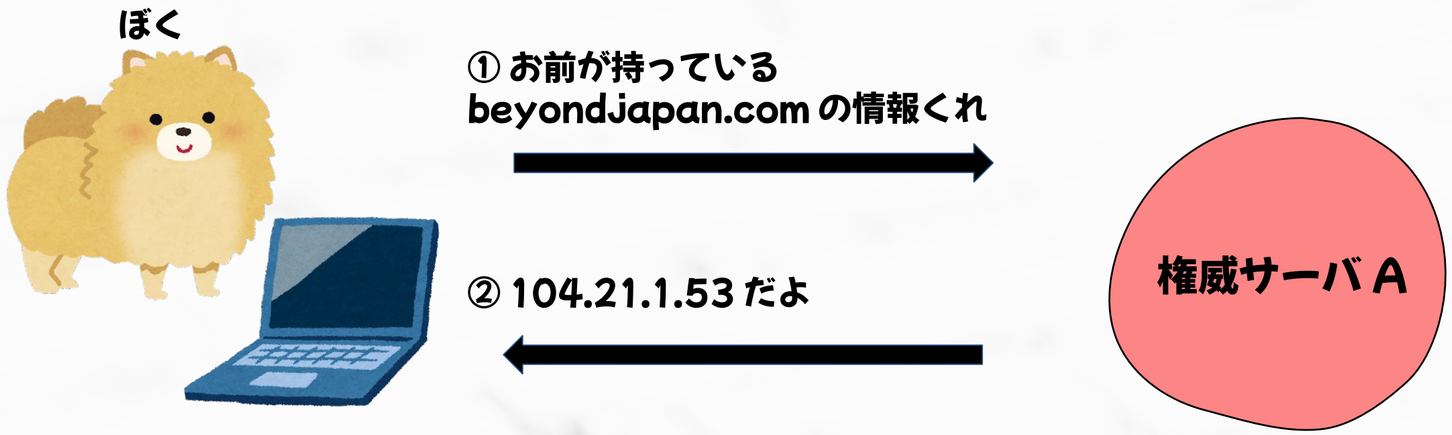
⑤104.21.91.53 やで

権威サーバ A

フルサービスリゾルバを通して
Answer Section がもらえるまで
繰り返し（再帰的に）探す

オプションをつけるとフルサービスリゾルバと同じ動きが出来る

dig @ネームサーバ beyondjapan.com +norec



直接そいつにだけ聞ける（キャッシュを見ない）

DNS 問い合わせを追いかけてみる②

01

ルートサーバに聞いてみる

※ AUTHORITY SECTION = 今はこいつに任せとんねんゾーン

```
$ dig @a.root-servers.net beyondjapan.com +norec
.
.
省略
.
.
;; AUTHORITY SECTION:
com.            172800  IN      NS      e.gtld-servers.net.
com.            172800  IN      NS      b.gtld-servers.net.
com.            172800  IN      NS      j.gtld-servers.net.
com.            172800  IN      NS      m.gtld-servers.net.
com.            172800  IN      NS      i.gtld-servers.net.
com.            172800  IN      NS      f.gtld-servers.net.
com.            172800  IN      NS      a.gtld-servers.net.
com.            172800  IN      NS      g.gtld-servers.net.
com.            172800  IN      NS      h.gtld-servers.net.
com.            172800  IN      NS      l.gtld-servers.net.
com.            172800  IN      NS      k.gtld-servers.net.
com.            172800  IN      NS      c.gtld-servers.net.
com.            172800  IN      NS      d.gtld-servers.net.
```

02

任されてるサーバに聞いてみる

```
$ dig @a.gtld-servers.net beyondjapan.com +norec
.
.
省略
.
.
;; AUTHORITY SECTION:
beyondjapan.com.    172800  IN      NS      ingrid.ns.cloudflare.com.
beyondjapan.com.    172800  IN      NS      damien.ns.cloudflare.com.
```

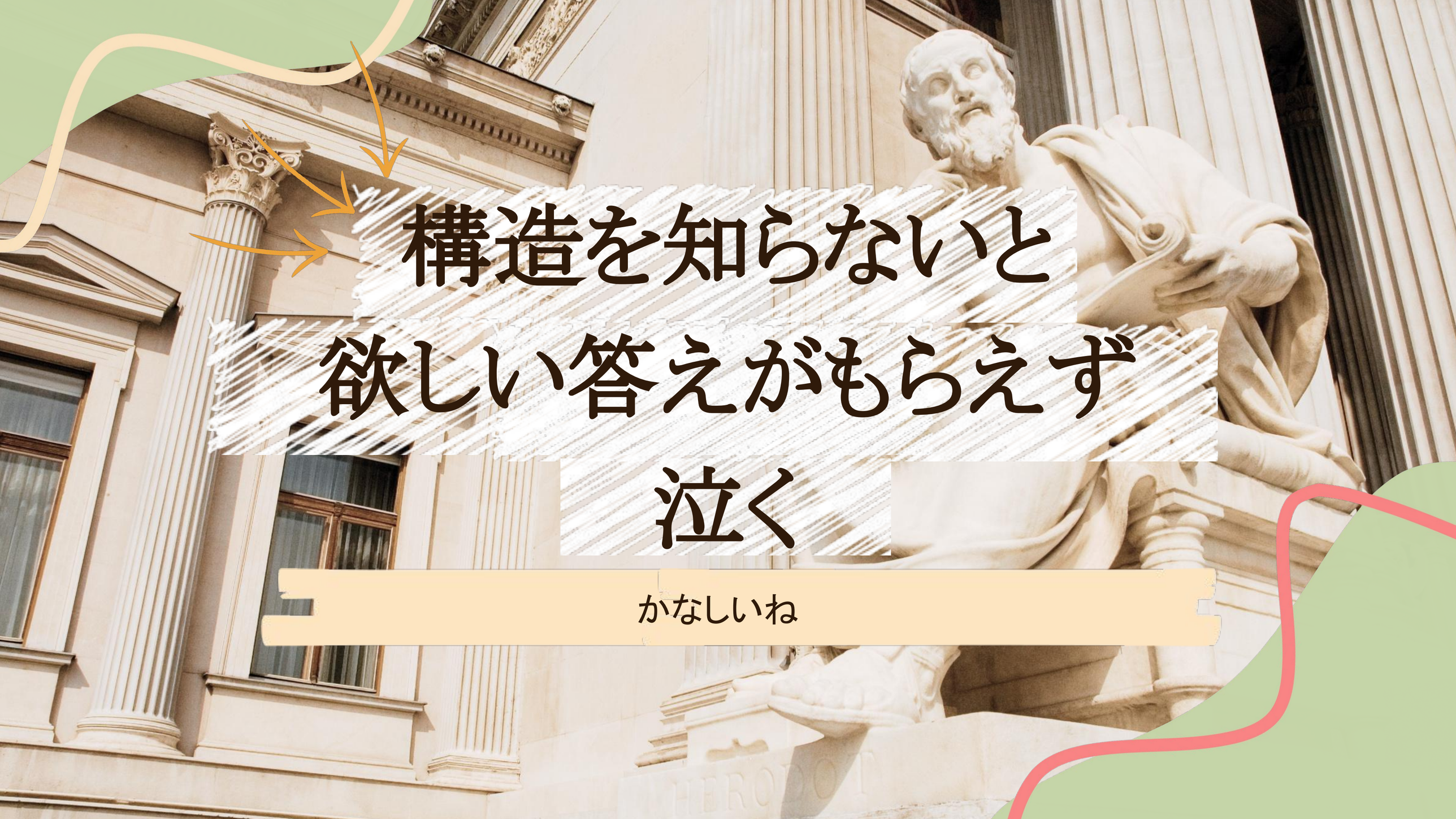
03

さらに聞いてみる

※ ANSWER SECTION = ほんまの答えゾーン

```
$ dig @ingrid.ns.cloudflare.com beyondjapan.com +norec
.
.
省略
.
.
;; ANSWER SECTION:
beyondjapan.com.    300     IN      A       172.67.167.78
beyondjapan.com.    300     IN      A       104.21.91.53
```

階層構造がよくわかる



構造を知らないと欲しい答えがもらえず泣く

かなしいね

別名多すぎ

DNS の話するとき
いろんな名称が出てきます

スタブリゾルバ

フルサービスリゾルバ

権威サーバ

ネームサーバ

他にも キャッシュサーバ
コンテンツサーバ とか

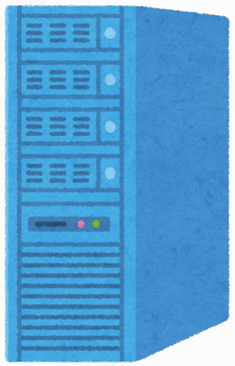
どれが誰だよ

登場人物

ぼく

スタブリゾルバ

フルサービスリゾルバ



ルートサーバ

com 用サーバ

権威サーバ A

登場人物

たけだ
＝
ぼく

やばーい

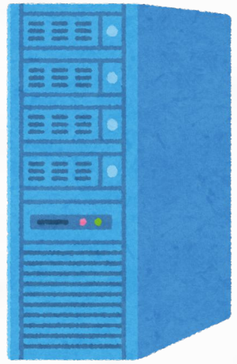
リゾルバ
＝
スタブリゾルバ

名前「解決」
するので

PC の OS か
ブラウザ の機能

要求を送るだけ

キャッシュサーバ
＝
リゾルバ
＝
フルサービスリゾルバ



TTL のぶん
レコードキャッシュ
するので

公開 DNS サーバとか
ISP の DNS サーバとか

再起検索で完全に
DNS 解決できる

「名前」解決
するので

＝ ネームサーバ ＝
＝
DNS サーバ

リゾルバに渡す
コンテンツを保有
しているので

コンテンツサーバ

＝ ルートサーバ

＝ com 用サーバ

＝ 権威サーバ A

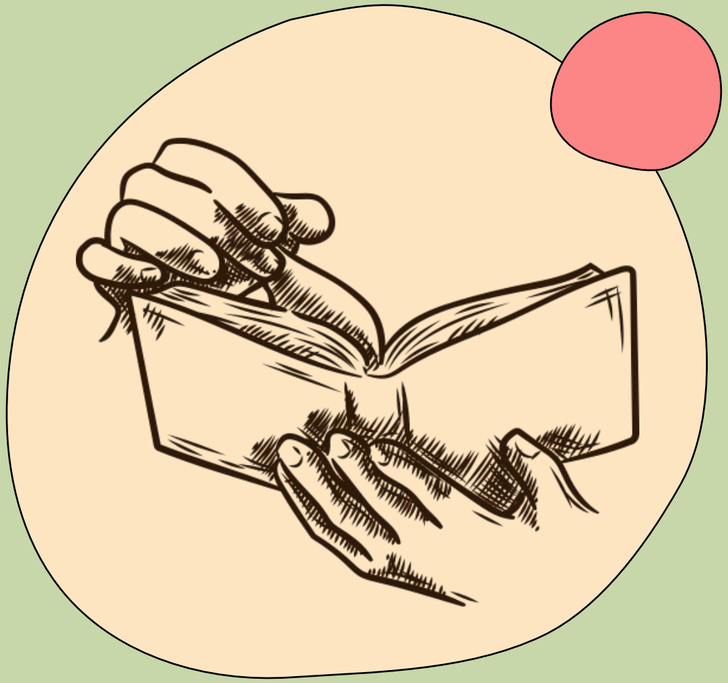
勉強するとき
どれがなにかわからんくて
泣く

別のもんに同じ名前つけなさんな

ちなみに

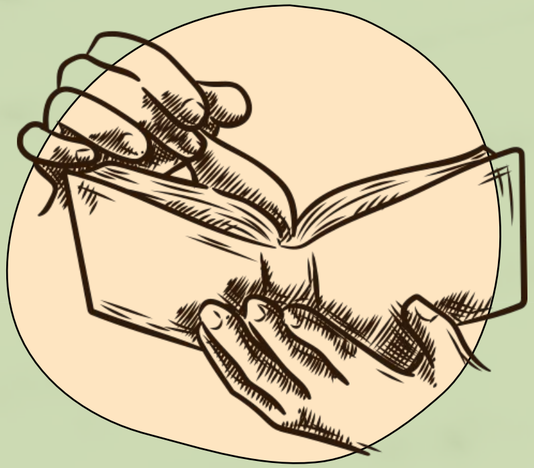
Route53 でゾーン作成すると勝手に NS レコードが4つできる
（SOA レコードも出来る）

こんなん ⇒ ns-〇〇.awsdns-〇〇.com
これがネームサーバ



TTL
ややこしすぎ

TTL = Time To Live = たいむとぅらいぶ

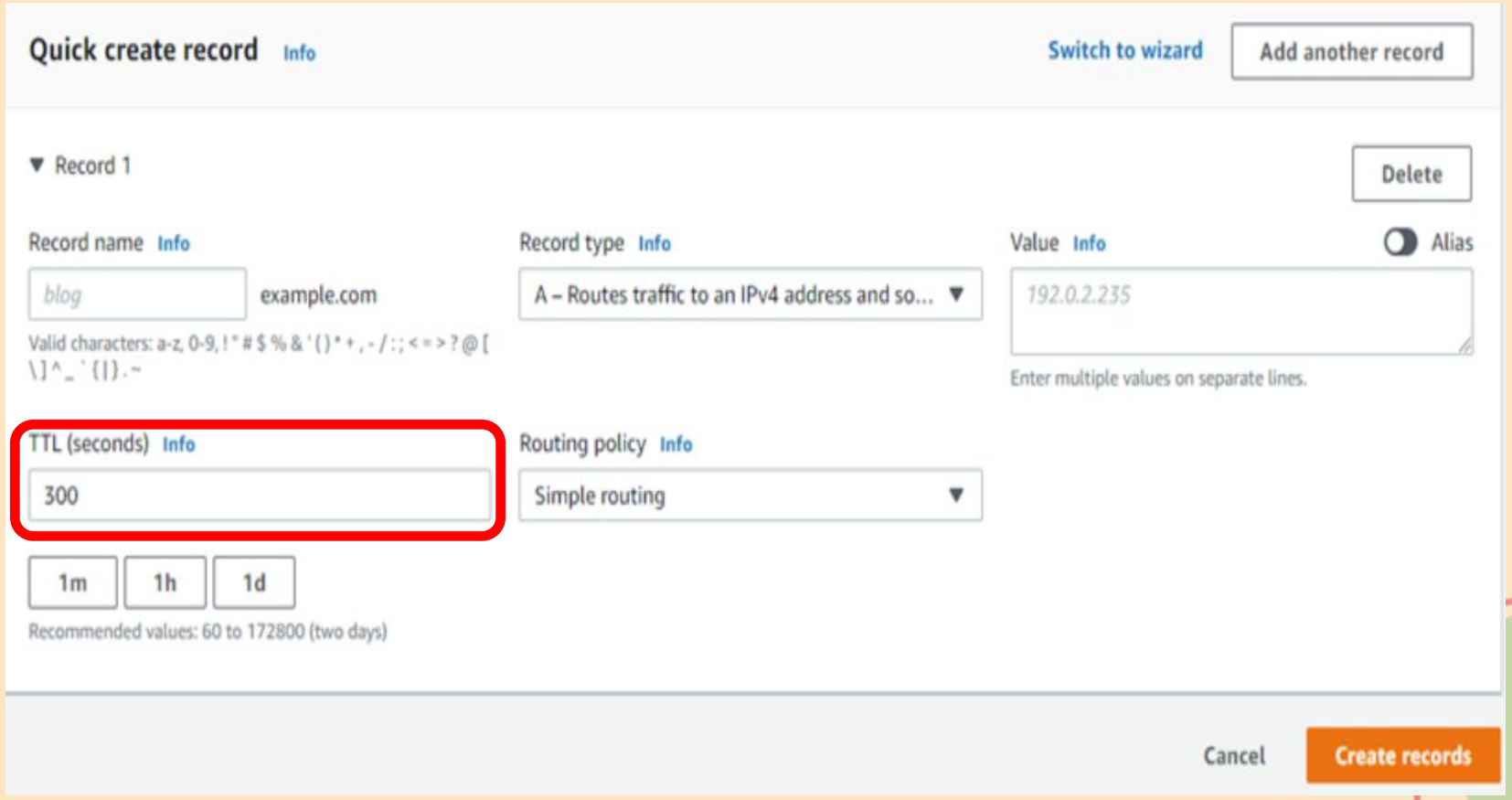


DNS レコードをキャッシュに保持してもいい時間

つまり

「キャッシュサーバへの指示」

AWS Route53



- 権威サーバに関しては自分で決められる

DNS 問い合わせを追いかけたときのやつ

01

ルートサーバ

TTL＝172800 秒＝2日

```
$ dig @a.root-servers.net beyondjapan.com +norec
.
.
省略
.
.
;; AUTHORITY SECTION:
com.            172800  IN      NS      e.gtld-servers.net.
com.            172800  IN      NS      b.gtld-servers.net.
com.            172800  IN      NS      j.gtld-servers.net.
com.            172800  IN      NS      m.gtld-servers.net.
com.            172800  IN      NS      i.gtld-servers.net.
com.            172800  IN      NS      f.gtld-servers.net.
com.            172800  IN      NS      a.gtld-servers.net.
com.            172800  IN      NS      g.gtld-servers.net.
com.            172800  IN      NS      h.gtld-servers.net.
com.            172800  IN      NS      l.gtld-servers.net.
com.            172800  IN      NS      k.gtld-servers.net.
com.            172800  IN      NS      c.gtld-servers.net.
com.            172800  IN      NS      d.gtld-servers.net.
```

02

TTL＝172800 秒＝2日

```
$ dig @a.gtld-servers.net beyondjapan.com +norec
.
.
省略
.
.
;; AUTHORITY SECTION:
beyondjapan.com.    172800  IN      NS      ingrid.ns.cloudflare.com.
beyondjapan.com.    172800  IN      NS      damien.ns.cloudflare.com.
```

03

権威サーバ

TTL＝300 秒＝5分

```
$ dig @ingrid.ns.cloudflare.com beyondjapan.com +norec
.
.
省略
.
.
;; ANSWER SECTION:
beyondjapan.com.    300     IN      A       172.67.167.78
beyondjapan.com.    300     IN      A       104.21.91.53
```

TTLの観点から見た名前解決の流れ
※キャッシュ済の例

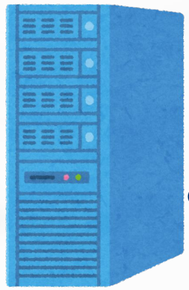
ぼく

スタブリゾルバ

① beyondjapan.com くれ

② 104.21.1.53 だってよ

フルサービスリゾルバ



NS レコード参照

172800 秒間
com サーバをキャッシュ

ルートサーバ

NS レコード参照

172800 秒間
A サーバをキャッシュ

com 用サーバ

A レコード参照

300 秒間
「104.21.1.53」
キャッシュ

権威サーバ A

フルサービスリゾルバくんは TTL 分の
秒数が経過するまではキャッシュを見続ける

たとえば 300 秒経過後

ぼく

スタブリゾルバ

① beyondjapan.com くれ

④ 104.21.1.53 だってよ

フルサービスリゾルバ

172800 秒
経ってないので
NS レコードについては
キャッシュ参照

ルートサーバ

172800 秒
経ってないので
NS レコードについては
キャッシュ参照

com 用サーバ

② おしえてー

③ 104.21.91.53 やで

権威サーバ A

TTL 分 経過したあとに問い合わせがあったら
改めて現状を確認しに行く
（そして再度キャッシュ）

もし権威サーバの「レコード」を変更したら

このようなAレコードを変更した場合

```
beyondjapan.com.      300    IN     A      104.21.91.53
```

すでにフルサービスリゾルバにキャッシュされているなら

検索しても最大300秒は変更前のAレコード(キャッシュ)が返る

変更後、権威サーバに反映されているかすぐ確認したい……??
なら直接権威サーバに聞こう！

$ dig @権威サーバ beyondjapan.com +norec

$ dig beyondjapan.com　なんてしたらキャッシュが返るよ

もし「権威サーバ自体」を変更したら

com 用サーバ

これからは
こっちに頼むわ

権威サーバ A

beyondjapan.com は
104.21.1.53 だ！

まだ 172800 秒
経ってないなぁ
キャッシュで！

フルサービスリゾルバ

権威サーバ B

beyondjapan.com は
142.250.207.110 だ！

beyondjapan.com くれ

104.21.1.53 ！
（サーバ A の情報）

権威サーバ A

ぼく

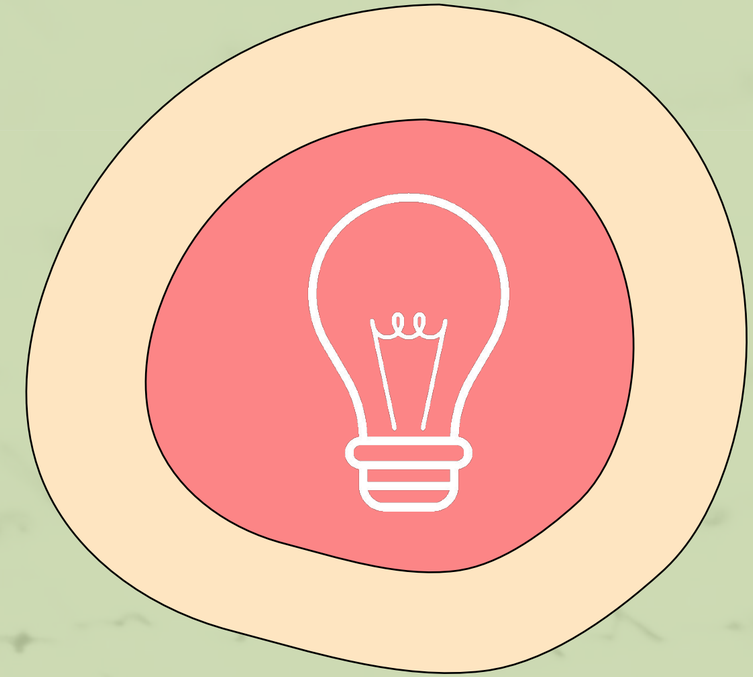
スタブリゾルバ

作業前に「変更後どの TTL の反映を待つ必要があるか」理解しよう

TTLどれくらいか把握してないと変更時に泣く

そのとき使われる「浸透」という言い訳

おまけ



古い BIND の持つ問題

01

古い BIND を搭載した
DNS サーバだと
NS レコード TTL 復活現象が
起こる

幽霊ドメイン名脆弱性

02

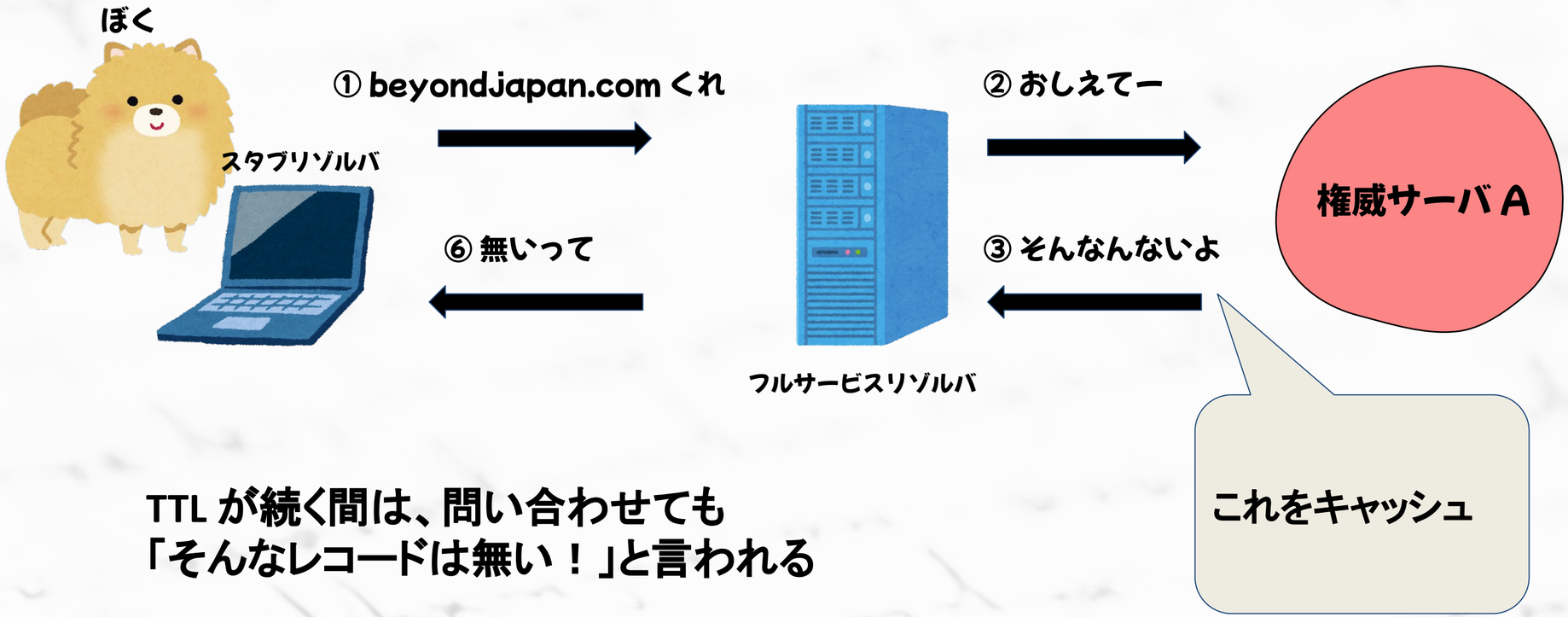
キャッシュが消える前に
対象のドメインを検索すると
古い IP と古い NS レコードを
受け取って
キャッシュを更新してしまう

03

現在は「NS レコードキャッシュ更新時に既にキャッシュ済の NS レコードの TTL 以上の値に更新しない」などの修正が入った

ネガティブキャッシュに気をつけて

まだ A レコード登録してないうちにブラウザでコンテンツ確認しちゃうポメ



TTL が続く間は、問い合わせても
「そんなレコードは無い！」と言われる

ネガティブキャッシュは SOA レコードに依存

SOA レコードの最後の項目か SOA 自身の TTL の
どちらか短い方が適用される（下の例では 900 s）

```
dig hogepome.com
;
;
省略
;
;
;; QUESTION SECTION:
;hogepome.com.                  IN      A

;; AUTHORITY SECTION:
com.                    900     IN      SOA     a.gtld-servers.net. nstld.verisign-grs.com. 1674107425 1800 900 604800 86400
```

ネガティブキャッシュ 食らわずに権威サーバへの反映を確認するには？

やっぱこれっすわ

$ dig @権威サーバ beyondjapan.com +norec

これだけ覚えて帰ってください

DNS

むじぃ

おわり