DNS Summer Days 2013 チュートリアル 2013-07-19

DNS再入門

株式会社ハートビーツ　滝澤隆史

私は誰

- 氏名: 滝澤 隆史 @ttkzw
- 所属: 株式会社ハートビーツ
  - サーバの構築・運用や24時間365日の有人監視をやっている会社
  - いわゆるMSP（マネージド サービス プロバイダ）
- DNSとの関わり
  - システム管理者として1997年から2006年までネームサーバの運用
    - BIND4, BIND8, djbdns, BIND9
  - 現在は個人サーバでネームサーバを運用
    - NSD, Unbound
  - 日本Unboundユーザー会
    - Unbound/NSDの文書の翻訳
  - DNSは趣味です(ｷﾘｯ

アジェンダ

- イントロダクション
  - DNSの背景
  - DNSの概要
- 本編
  - ドメイン名
  - ドメイン名の管理
  - リソースレコード
  - マスターファイル
  - DNSメッセージ
  - リゾルバとネームサーバ

注意

- DNSの基本仕様を中心に話します。
- 拡張機能については原則として話しません。
  - EDNS0, DNSSECなど

DNSの背景

ホスト名とIPアドレスの関係。リゾルバの役割。

コンピュータ間で通信するには

- 互いに識別できるユニークな識別子が必要

ネットワーク アドレス

- コンピュータに割り当てられたユニークな番号
- ネットワーク上ではこのアドレスを使ってコンピュータ間の通信を行う

ネットワーク アドレスの例

- RFC 208 "ADDRESS TABLES"
  - ARPANET時代のネットワーク アドレス表

IMP NUMBER	SITE NAME	HOST NUMBER	HOST	NETWORK ADDRESS	SCHEDULED INSTALLATION
1	UCLA	0	SIGMA-7	1	
		1	IBM 360/91	65	
2	SRI	0	PDP-10 (NIC)	2	
		1	PDP-10 (AI)	66	
3	UCSB	0	IBM 360/75	3	
4	UTAH	0	PDP-10	4	
5	BBN	0	DDP-516	5)	See Note 1
		1	PDP-10 (A)	69)	
		2	PDP-10 (B)	133	

IPアドレス

- TCP/IPにおけるネットワーク アドレス
- IPアドレスの記述例
  - 192.0.2.1
  - 2001:db8:dead:beef:123:4567:89ab:cdef
- 数字の羅列であるため覚えにくい
  - 人に優しくない
  - 特にIPv6のアドレスなんて覚えられない。
  - →ホスト名を使う

ホスト名

- コンピュータを識別するための名前
- ホスト名の例
  - OFFICE-1
  - SRI-NIC
  - www.example.jp
- 通常はIPアドレスの代わりにホスト名を使ってコンピュータにアクセスする

名前解決

- 通常はIPアドレスの代わりにホスト名を使ってコンピュータにアクセスする
- 実際はネットワーク上のコンピュータにアクセスするにはIPアドレスを使う
- ホスト名をIPアドレスに変換する（名前を解決する）仕組みがあればよい
- →リゾルバ

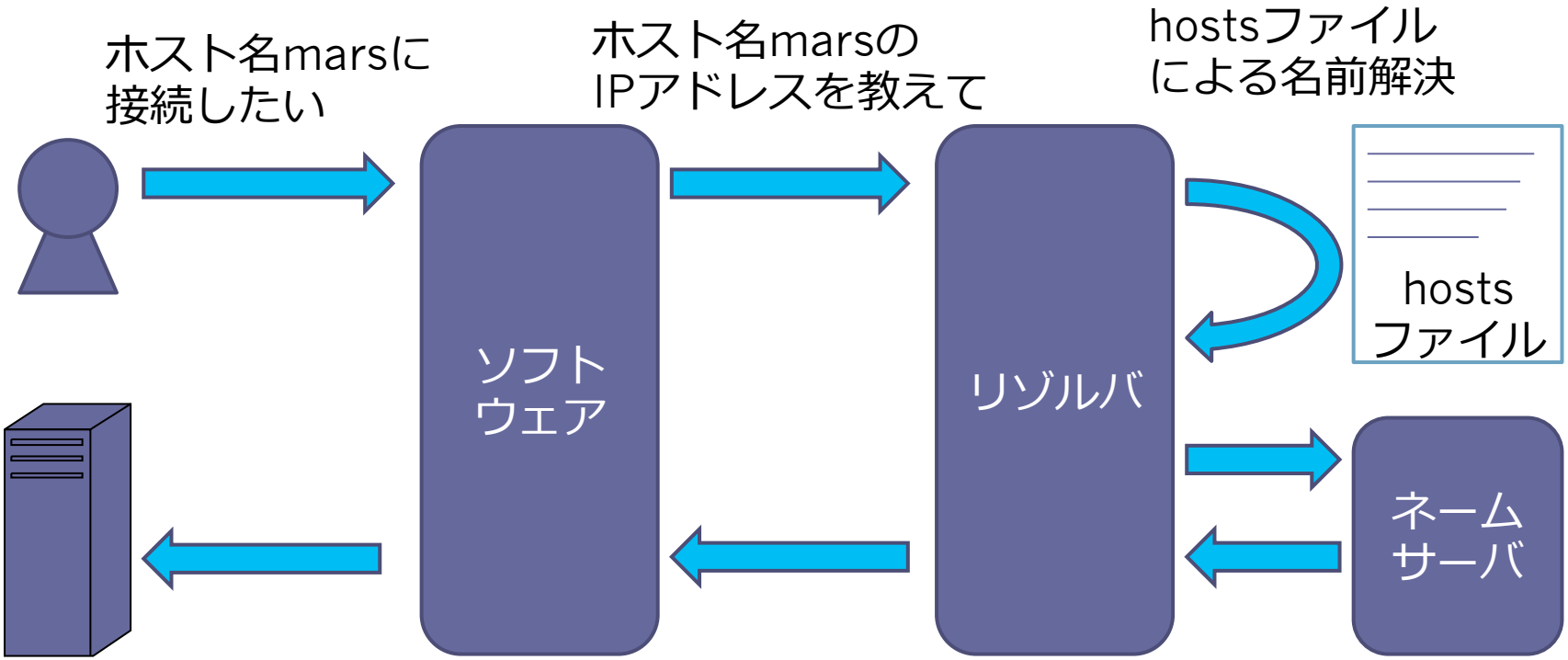
リゾルバ

- 名前を解決する仕組み
  - ホスト名に対するIPアドレスを見つける
- OSの機能あるいはライブラリやソフトウェアとして用意されている
- OS上で動作するソフトウェアの要求に応じてホスト名に対するIPアドレスを調べてくれる

リゾルバの検索方法

- hostsファイル
  - /etc/hosts
  - C:\Windows\System32\drivers\etc\hosts
- DNS
  - 今回のテーマ

リゾルバ


ホスト名marsに
接続したい
ホスト名marsの
IPアドレスを教えて
hostsファイル
による名前解決
ソフト
ウェア
リゾルバ
hosts
ファイル
ネーム
サーバ
IPアドレス192.0.2.4
に接続する
ホスト名marsの
IPアドレスは
192.0.2.4である
DNSによる
名前解決

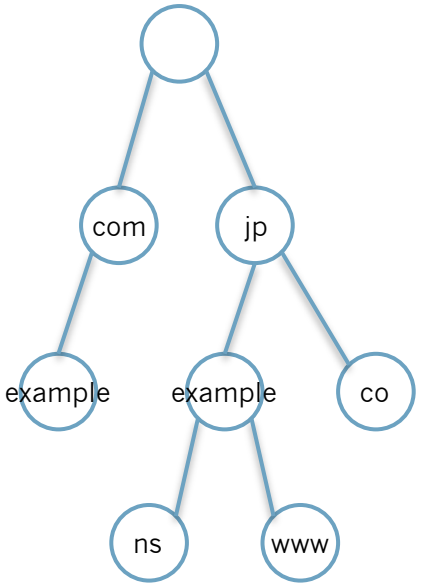
このセクションのまとめ

- リゾルバがホスト名に対応するIPアドレスを見つけてくれる（名前の解決）
  - hostsファイルやDNSを使う
- ネットワーク上のコンピュータにアクセスするために、覚えにくいIPアドレスの代わりにホスト名を使うことができる

DNSの概要

DNS（Domain Name System）とは

- ドメイン名を用いた階層構造を持つ
  - www.example.jpのような形式
- 分散型データベース
  - データベース サーバ（権威ネームサーバ）が分散して存在している



DNSで扱うデータ

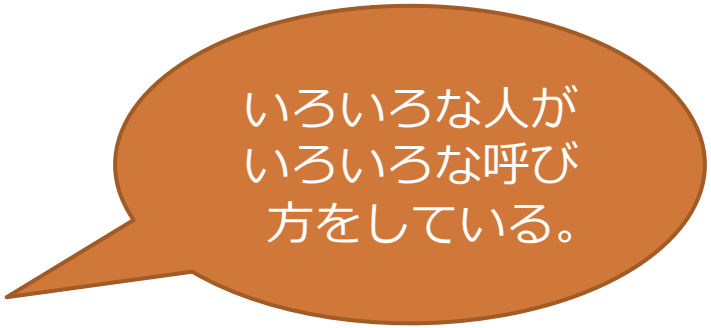
- コンピュータやネットワークに関するリソースの情報（ホスト名やIPアドレスなど）
  - ホスト名の名前解決ができる
  - ホスト名とIPアドレス以外のリソースも扱う
- リソースレコードという

DNSで扱うデータ

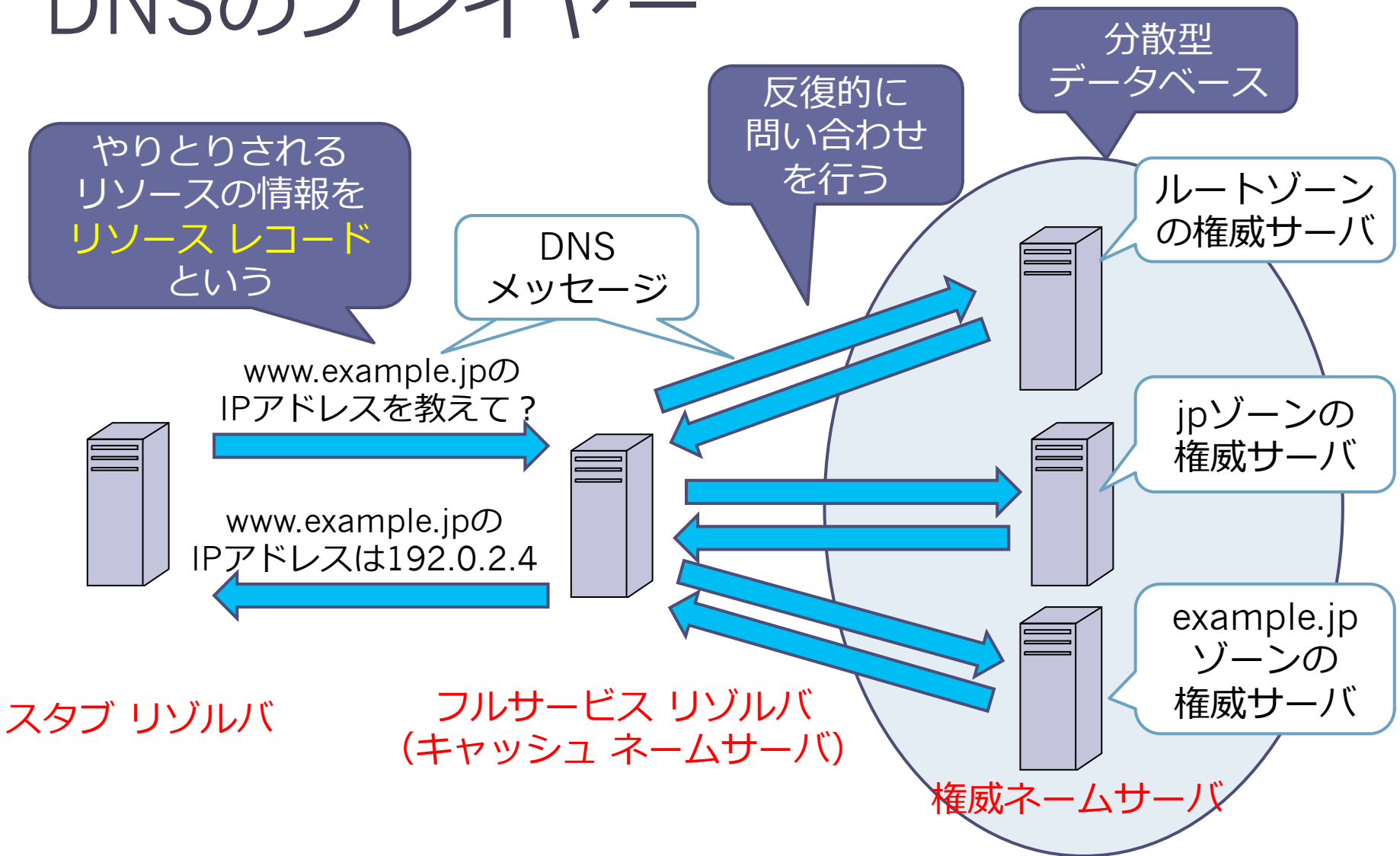
- 主要なリソースレコード
  - ゾーンの権威の開始を示すSOAレコード
  - 権威ネームサーバを示すNSレコード
  - IPアドレスを示すAレコード
  - IPアドレスに対する名前を示すPTRレコード
  - 別名に対する正式名を示すCNAMEレコード

DNSのプレイヤー

- スタブ リゾルバ
- フルサービス リゾルバ
  - キャッシュ ネームサーバ
  - キャッシュDNSサーバ
  - DNSキャッシュサーバ
  - キャッシュサーバ
- 権威ネームサーバ
  - 権威DNSサーバ
  - DNS権威サーバ
  - コンテンツサーバ
  - 権威サーバ



DNSのプレイヤー


やりとりされる
リソースの情報を
リソース レコード
という
DNS
メッセージ
反復的に
問い合わせ
を行う
分散型
データベース
ルートゾーン
の権威サーバ
jpゾーンの
権威サーバ
example.jp
ゾーンの
権威サーバ
www.example.jpの
IPアドレスを教えて？
www.example.jpの
IPアドレスは192.0.2.4
スタブ リゾルバ
フルサービス リゾルバ
（キャッシュ ネームサーバ）
権威ネームサーバ

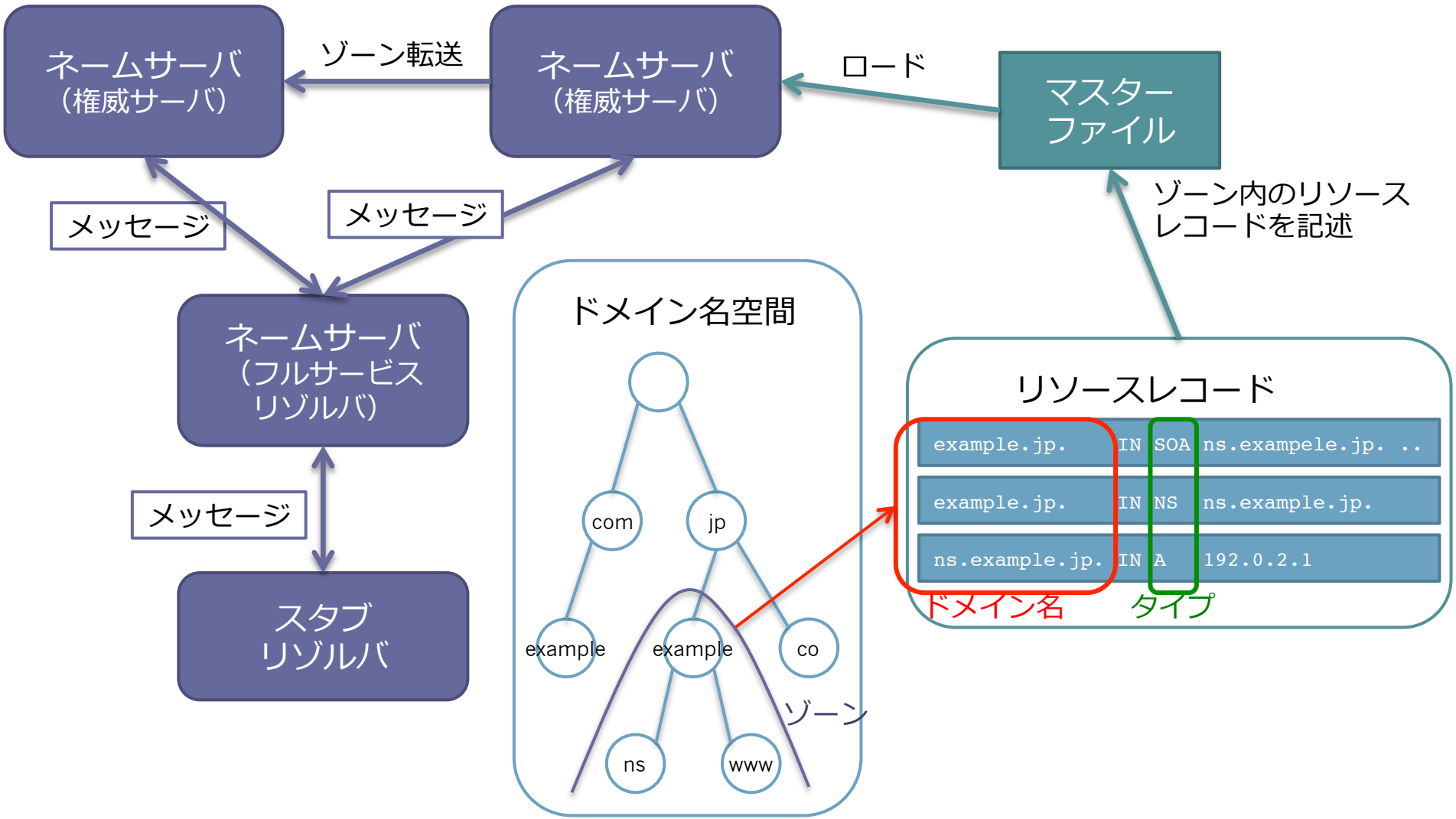
本日、学んでもらうことは

- DNSの名前空間として、ドメイン名の仕組み
- DNSで扱うデータとして、リソースレコード
- ゾーンのリソースレコードの集まりを記述するマスターファイル
- DNSの問い合わせや応答で運ばれるDNSメッセージ
- DNSをハンドリングするリゾルバとネームサーバ

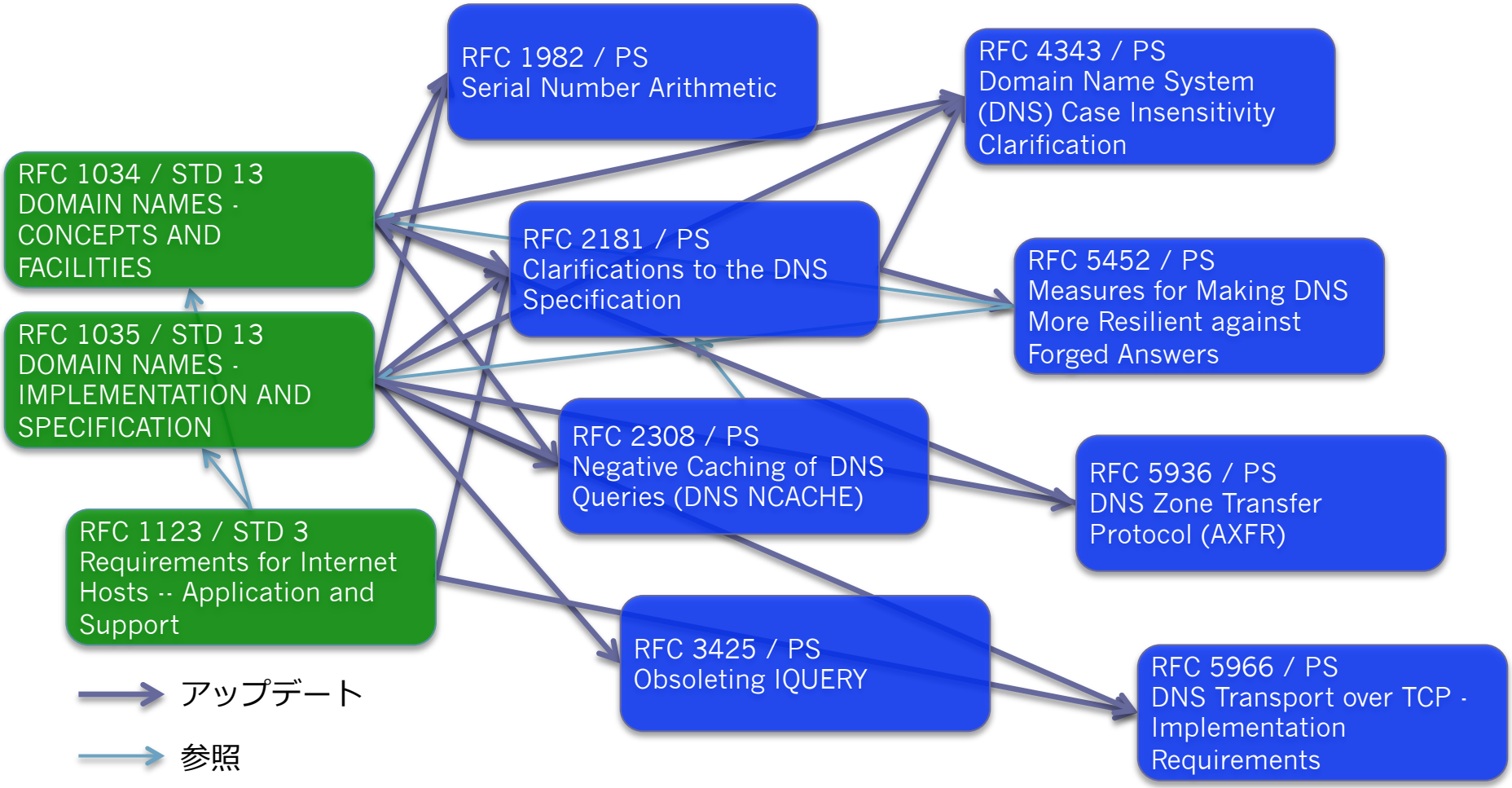
本編

ここからが本番

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
メッセージ
メッセージ
ゾーン内のリソース
レコードを記述
ネームサーバ
（フルサービス
リゾルバ）
メッセージ
スタブ
リゾルバ
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ

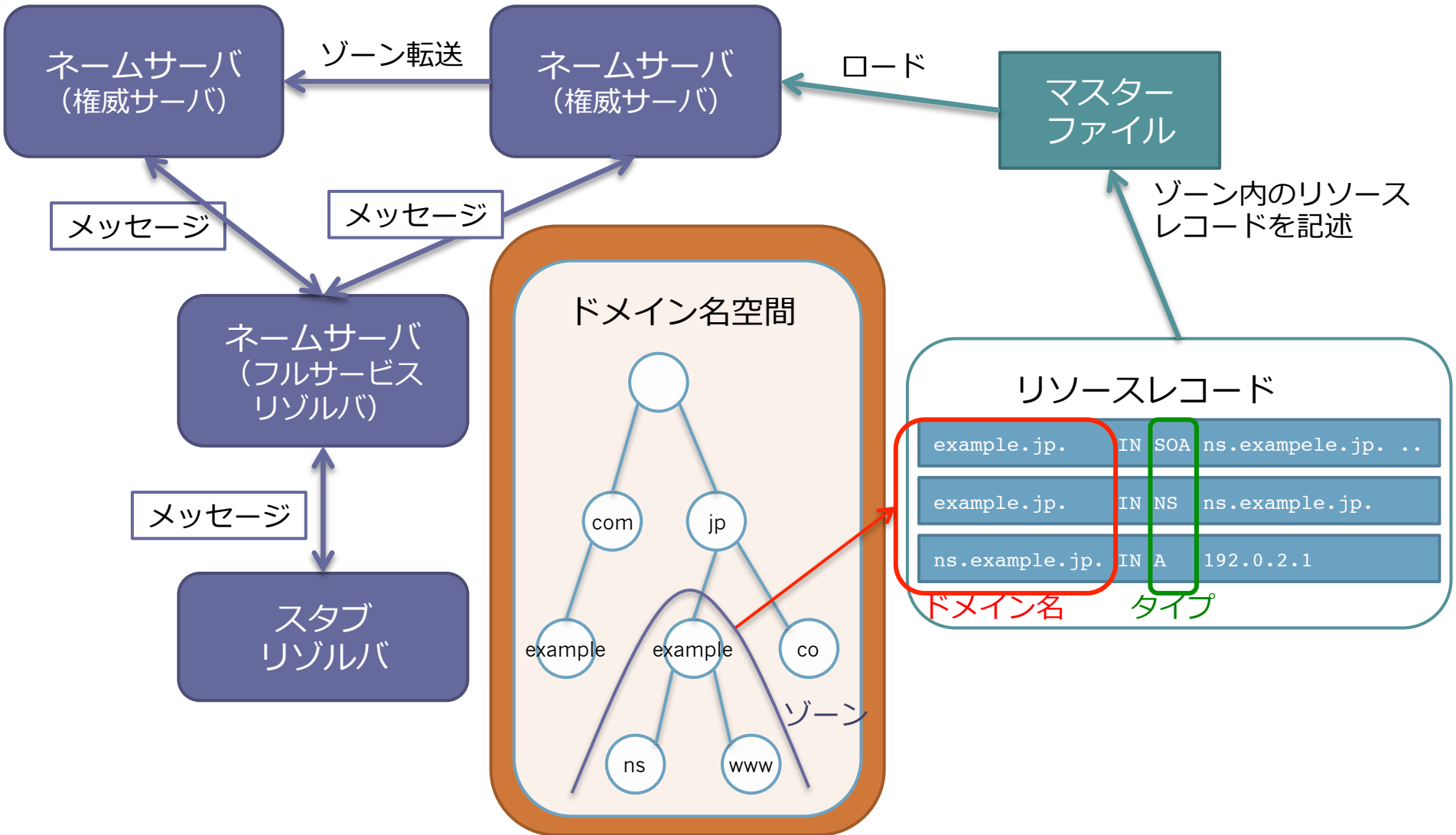
参考資料: DNSの基本仕様のRFC



ドメイン名

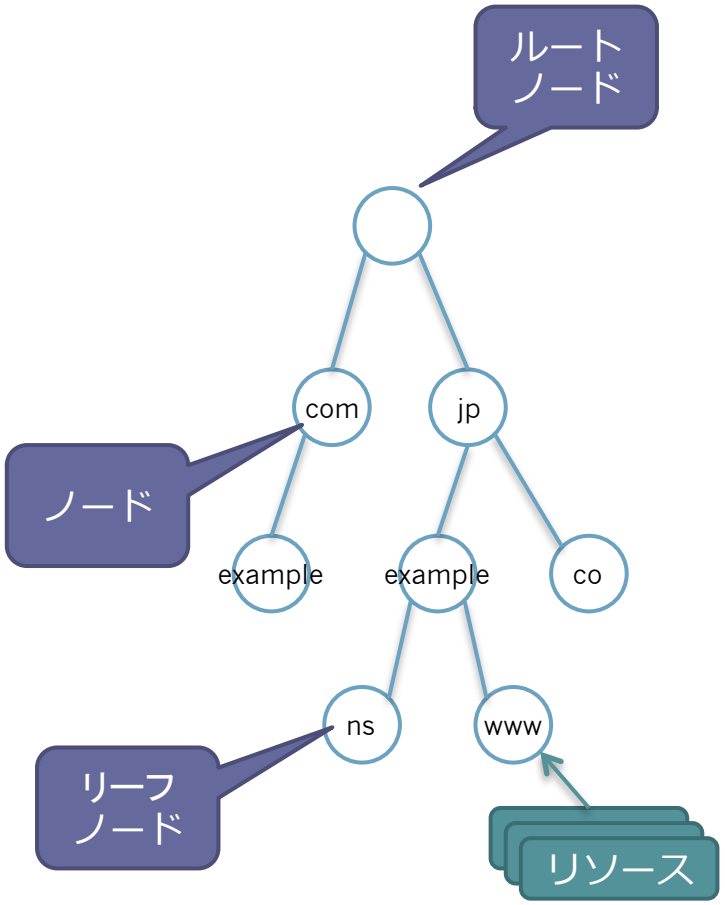
ドメイン名についてもう少し詳しく見ていこう

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
メッセージ
メッセージ
ゾーン内のリソース
レコードを記述
ネームサーバ
（フルサービス
リゾルバ）
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ
メッセージ
スタブ
リゾルバ

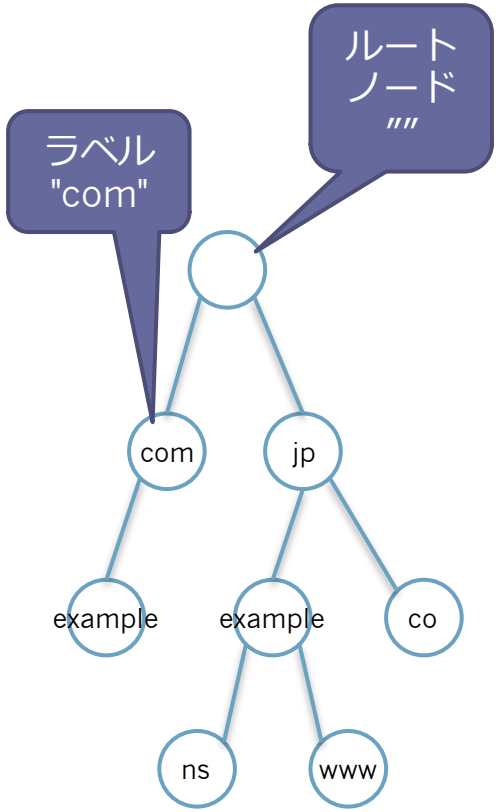
ドメイン名空間の構造

- ドメイン名空間はツリー構造
- 各ノードとリーフはリソースの集まりに対応している
- 内部ノードとリーフノードを区別しない。両方とも「ノード」と呼ぶ。



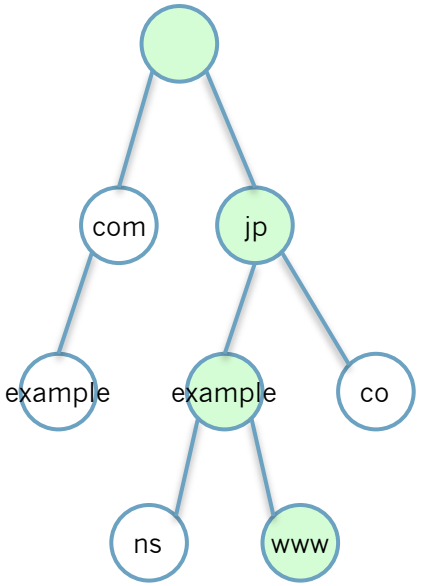
ラベル

- 各ノードはラベルを持つ。
- ルートのためにnullラベル（長さ0）が予約されている。



ドメイン名

- ノードのドメイン名はそのノードからルートノードまでのパス上のラベルのリスト
  - 例) "www", "example", "jp", ""

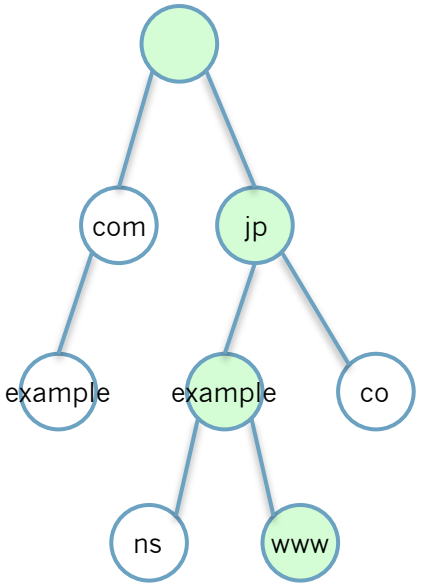


ドメイン名の内部表現

- ラベルはオクテットの長さと文字列で表される。
  - "www"の内部表現を16進数で表すと
    3 77 77 77

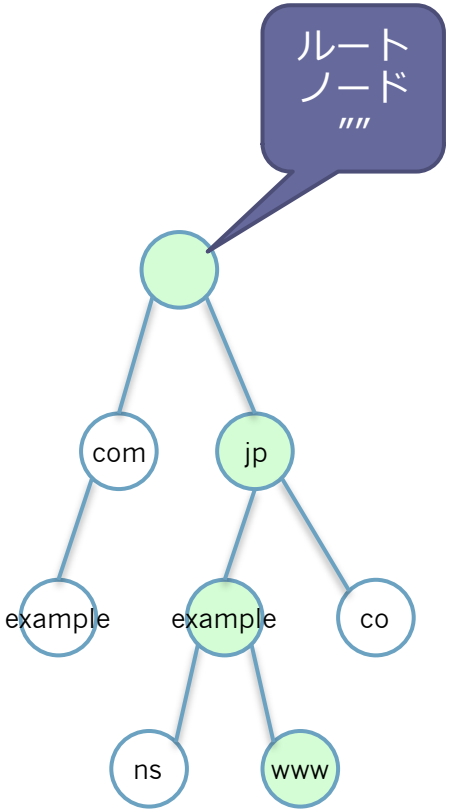
ドメイン名の内部表現

- ドメイン名はラベルをつなげたもの
- すべてのドメイン名はルートで終わり、ルートのラベルはnull文字であるため、内部表現はドメイン名の終わりに0バイトの長さを使う。
  - www.example.jp.の内部表現
    3 77 77 77 7 65 78 61 6d 70 6c 65 2 6a 70 0



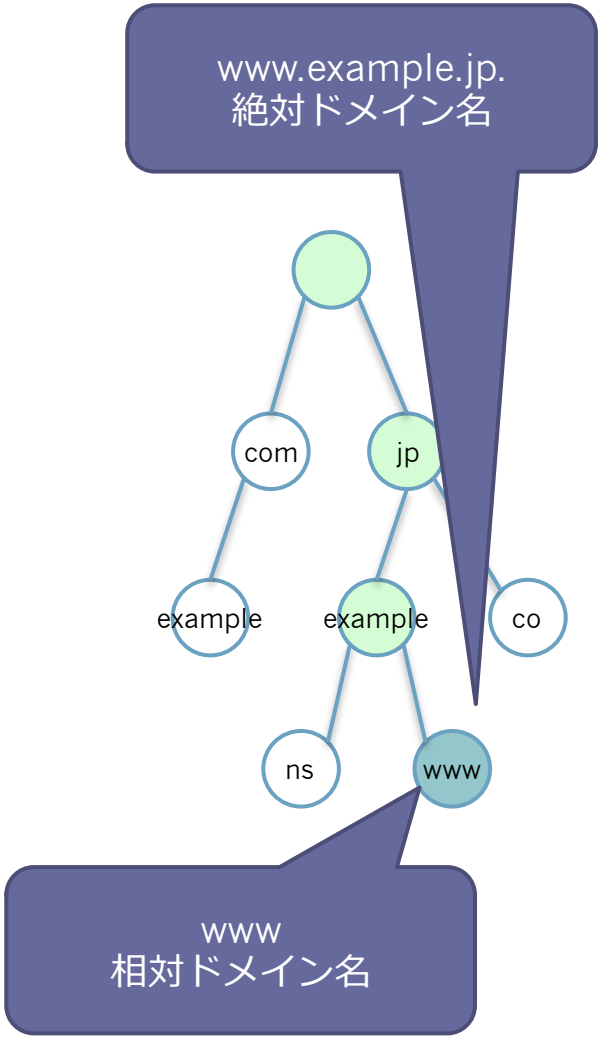
ドメイン名のテキスト表現

- ラベルの長さを省き、ラベルを"."で分ける。
  - 例）www.example.jp.
- ドメイン名はルート（空の）ラベルで終わるため、ドットで終わる形式になる。
  - 例）www.example.jp."空のラベル（非表示）"



絶対ドメイン名と相対ドメイン名

- 絶対ドメイン名
  - "www.example.jp."のようにドットで終わるドメイン名
- 相対ドメイン名
  - 親のドメイン名に対して相対的に表したドメイン名
  - "www" は親ドメイン名 "example.jp."に対する相対ドメイン名



絶対ドメイン名

- DNS関連の設定では原則として絶対ドメイン名を使う
  - マスターファイル
    ```
    example.jp. IN NS ns.example.jp.
    ```
  - 相対ドメイン名の方がわかりやすい場合もある
    ```
    $ORIGIN example.jp.
    www           IN A 192.0.2.4
    ```
- 最後に"."が付いていないと相対ドメイン名として扱われてしまう
  - 最後に"."を付け忘れると
    ```
    example.jp. IN NS ns.example.jp
    ```
  - 次のように解釈される
    ```
    example.jp. IN NS ns.example.jp.example.jp
    ```

絶対ドメイン名

- digなどのDNS関連のツールの場合も絶対ドメイン名を使う。
  - `dig www.example.jp. A`

相対ドメイン名

- pingを実行したときに、相対ドメイン名だけで名前解決をしてくれる場合がある
  - リゾルバで検索リストにより親ドメインを補完してくれるから
- ゾーンファイルを記述するときもオリジンに対する相対ドメイン名で記述できる
  - $ORIGIN example.jp.
    www IN A 192.0.2.4

検索リスト

- リゾルバの機能で、相対ドメイン名に対する親ドメイン名を補完するためのドメイン名のリスト
  - /etc/resolv.confの"domain"と"search"

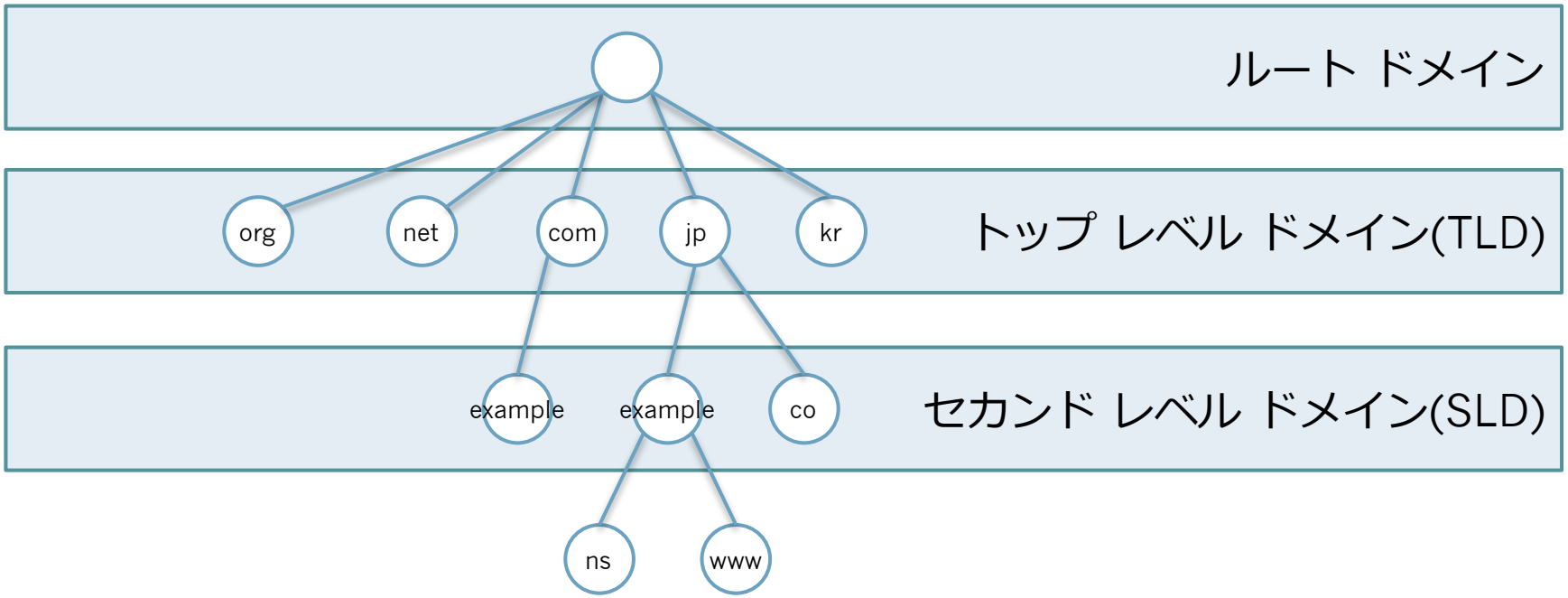
```
domain example.jp
nameserver 192.0.2.1
nameserver 192.0.2.2
```

検索リスト

- ルート"."は暗黙の検索リストのメンバー
  - "example.jp"は最終的には"example.jp."と解釈される。
  - そのため、実際にはアプリケーションに対して"example.jp"のように記述して使うことができる。
    - →FQDN（後述します）

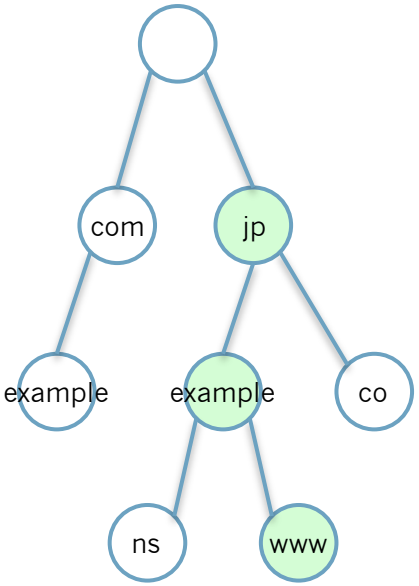
ルートドメイン、TLD、SLD

- 各ノードはノードの階層の深さによって呼び名が付く



完全修飾ドメイン名(FQDN)

- トップレベルドメインまでのすべてのラベルを含んだドメイン名を完全修飾ドメイン名（Fully Qualified Domain Name、略称FQDN）と呼ぶ
  - 例: "www.example.jp"

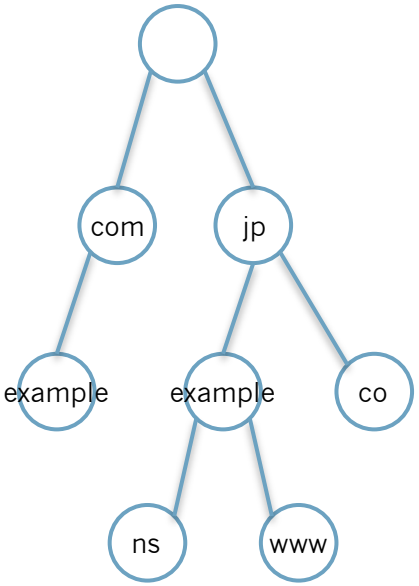


完全修飾ドメイン名(FQDN)

- ソフトウェアがドメイン名を扱うときにはFQDNを用いる
- ソフトウェアにドメイン名を設定/入力するときにはFQDNを入力する
- ルートドメインに対する相対ドメイン名と考えるとよい
  - 検索リストのメンバーとしてルート"."が解釈されるため

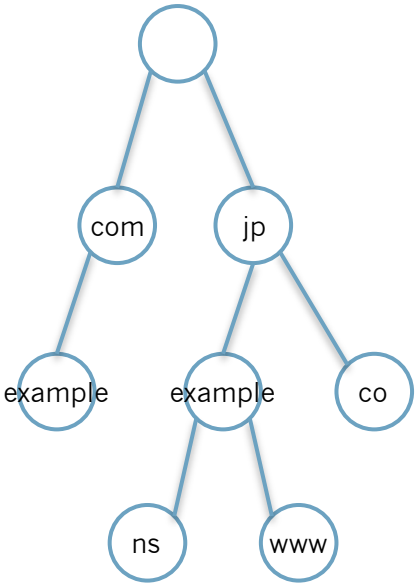
サブドメイン

- あるドメインに所属しているドメインをサブドメインと呼ぶ
- ルートドメインのサブドメイン
  - com
  - jp
- "jp"ドメインのサブドメイン
  - co.jp
  - example.jp
  - www.example.jp



サブドメイン

- サブドメインの側から見ると、"www.example.jp"は次のドメインのサブドメイン
  - example.jp
  - jp
  - ルートドメイン



ドメイン名のラベルの規則

- ホスト名の規則（RFC 952, RFC 1123）に従う
  - 英文字あるいは数字で始まる
  - 英文字あるいは数字で終わる
  - 間の文字は英文字、数字、ハイフンが使える

ドメイン名のラベルの規則

- 0オクテット以上63オクテット以下の文字列である
  - 0オクテットはルートドメインの空ラベルとして予約
- 兄弟ノードでは同じラベルを使用できない
- ルートドメインは次の規則に従う
  - 0オクテットの空のラベルである
  - "."と表すことがある

ドメイン名のラベルの規則

- ホスト名の規則に従わないケース
  - プロトコル上はどの8bitコードも許容されている
  - ホスト名のラベルとの衝突を防ぐために"_"で始まるものがある
    - RFC 2782 "A DNS RR for specifying the location of services (DNS SRV)"
      - _ldap._tcp.example.com.
    - RFC 6376 DomainKeys Identified Mail (DKIM) Signatures
      - foo.bar._domainkey.example.com.

ドメイン名とホスト名の長さ

- 絶対ドメイン名は255オクテット以下に制限されている
  - FQDNとして扱える文字数はルートドメインの空ラベル分を除いて253文字以下になる
- RFC 1123
  - ソフトウェアは63文字までのホスト名を扱えなければならない（MUST）
  - 255文字までのホスト名を扱うべき（SHOULD）
  - ネットワーク関連のソフトウェアの開発者はこの点に注意を払う必要がある

大文字小文字の取り扱い

- 大文字小文字の区別
  - ラベルやドメイン名などの比較の際には大文字小文字を区別しない。
- 大文字小文字の維持
  - 可能な限り大文字小文字を維持する。

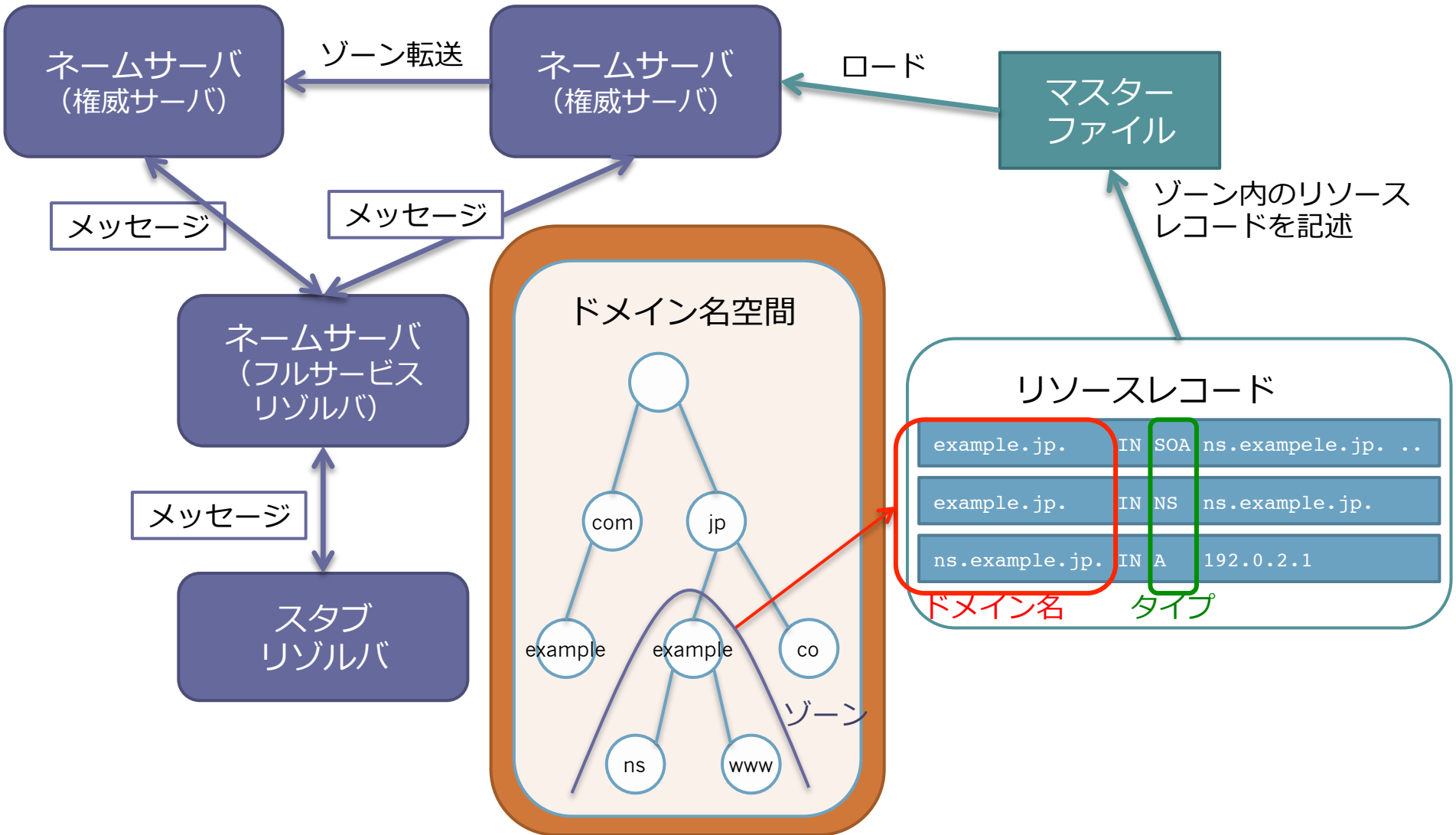
参考
RFC 4343 "Domain Name System (DNS) Case Insensitivity Clarification"

このセクションのまとめ

- ドメインの階層構造
- ルートドメイン、TLD、SLD
- 絶対ドメイン名
- 相対ドメイン名
- 完全修飾ドメイン名（FQDN）
- ドメイン名のラベルの規則
- ドメイン名とホスト名の長さ

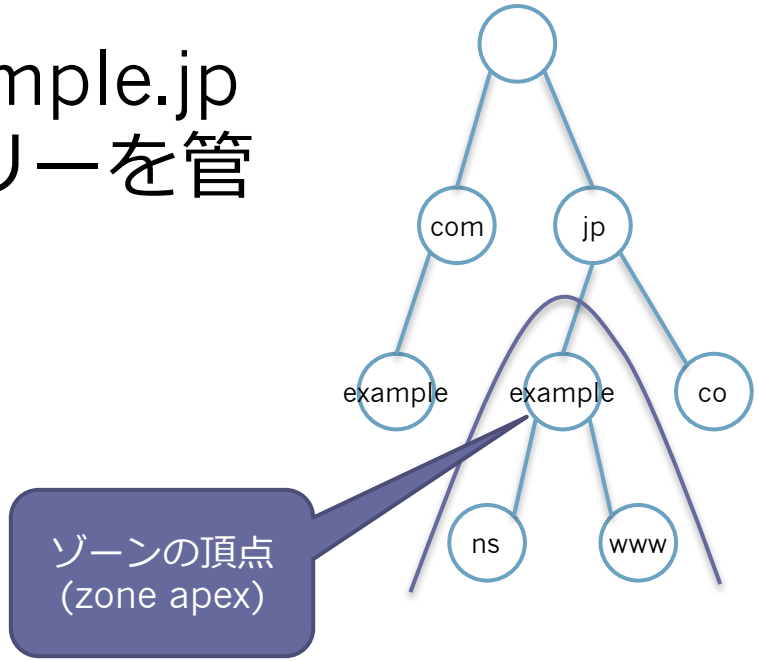
ドメイン名の管理

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
メッセージ
メッセージ
ゾーン内のリソース
レコードを記述
ネームサーバ
（フルサービス
リゾルバ）
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ
メッセージ
スタブ
リゾルバ

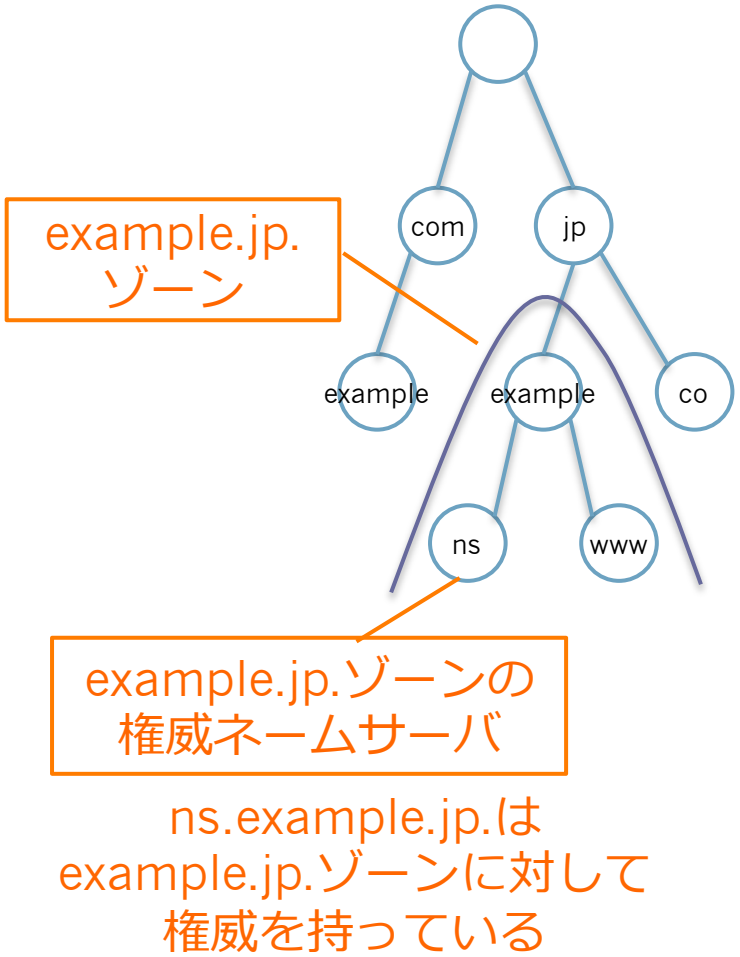
ゾーン

- ドメインを管理する単位をゾーンと呼ぶ
- "example.jp"ゾーンはexample.jpをゾーンの頂点とするツリーを管理する



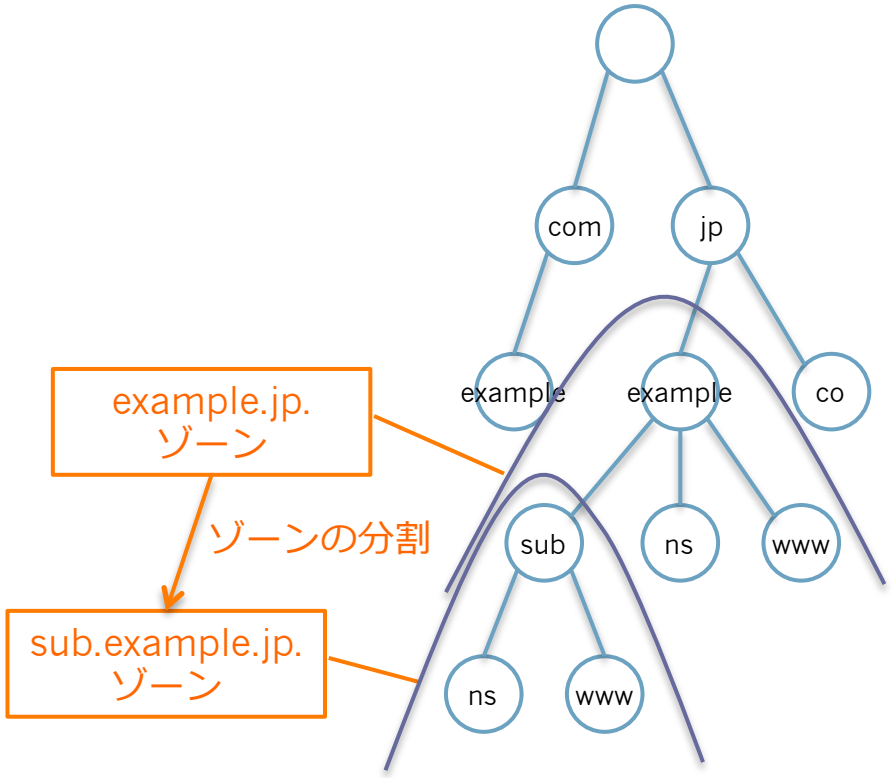
ゾーンと権威

- ネームサーバがそのゾーンを管理できる正式な権限を持っているときには、ネームサーバはそのゾーンに対する権威（authority）となる
- 権威を持つネームサーバを権威ネームサーバ（authoritative name server）と呼ぶ



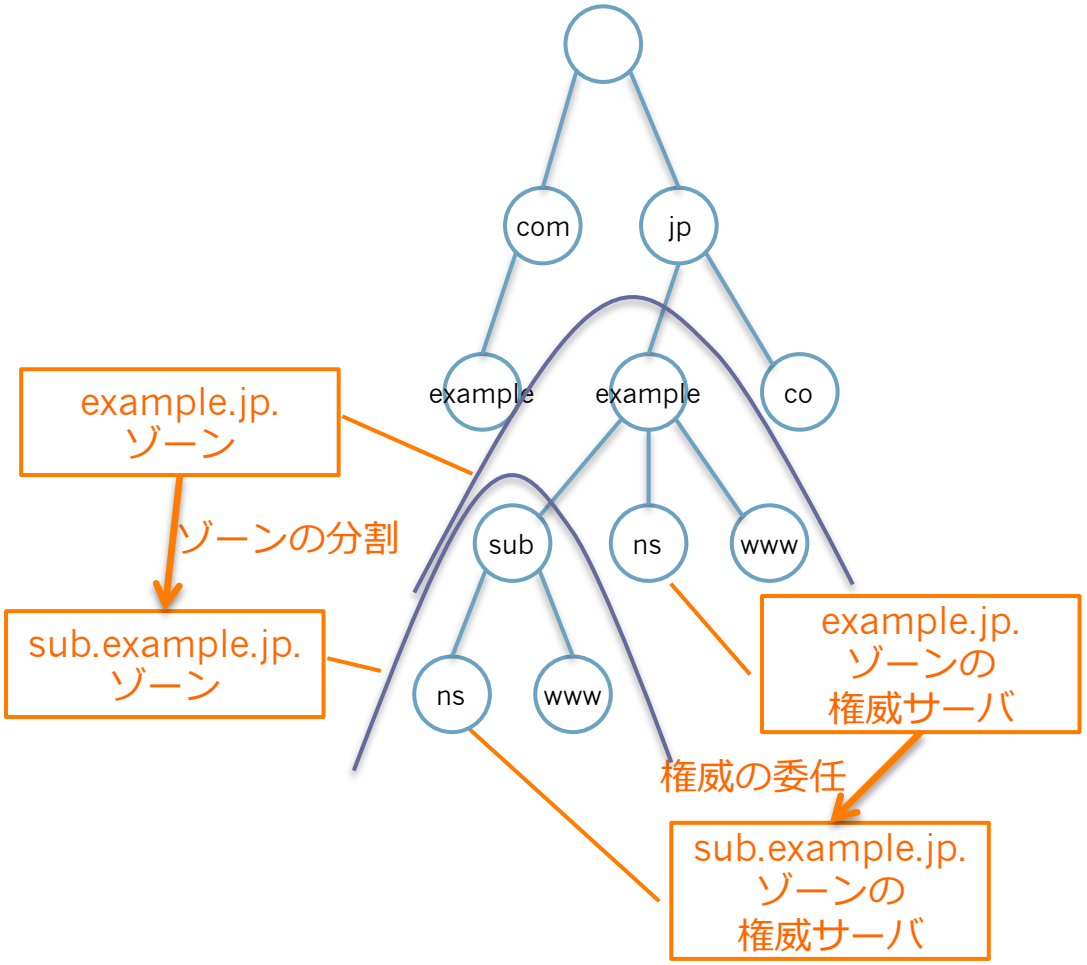
ゾーンの分割

- 各ドメインのゾーンはサブドメインのゾーンを分割することができる
- "example.jp"ドメインのサブドメインである"sub.example.jp"を別のゾーン（サブゾーン）として分割することができる



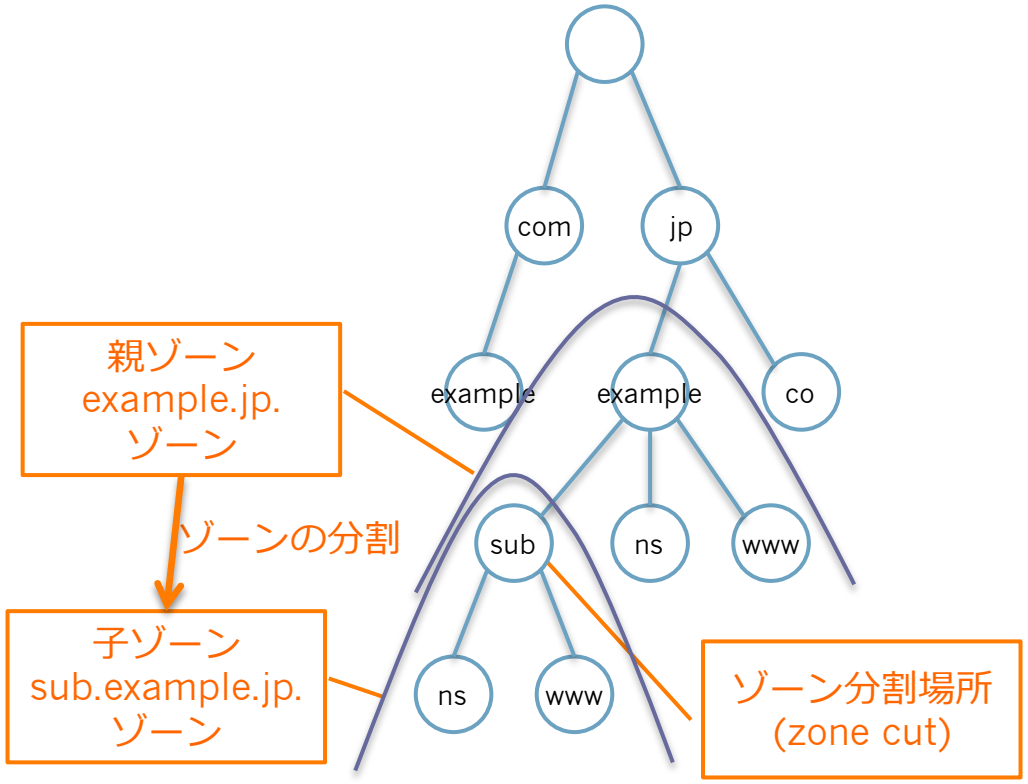
権威の委任

- この分割されたゾーンを管理する正式な権限を他のネームサーバに譲ることを権威の委任と呼ぶ



権威の委任

- 権威を委任する側を親ゾーン、
- 委任される側を子ゾーン、
- ゾーンを分割している場所をゾーン分割場所（zone cut）と呼ぶ



権威の委任

- ゾーン分割場所（zone cut）のドメイン名に対してグルー（glue）RRsを登録する。
  - 親ゾーンは子ゾーンの権威ネームサーバのNSレコードとAレコードを登録する
  - 子ゾーンは自身のゾーンのSOAレコードとNSレコードとAレコードを登録する。

権威の委任のためのRRsの記述例

- 親ゾーン example.jp.

```
sub.example.jp.     IN NS ns.sub.example.jp.
ns.sub.example.jp.  IN A  192.0.2.4
```

- 子ゾーン sub.example.jp.

```
@   IN SOA ns hostmaster 2013071901 3600 900 604800 900
    IN NS  ns.sub.example.jp.
ns  IN A   192.0.2.4
```

ゾーンの分割や権威の委任に関するよくある注意点

- サブドメイン毎にゾーンを分割しなければいけないわけではない
  - 同じサーバで運用するのであれば分割する必要はない
  - 次のような書き方もできる
    - `www.sub  IN A  192.0.2.4`

分散管理

- DNSはルートゾーンから下位のゾーンに対して権威を委任することにより、分散管理が成り立っている
  - ルートゾーン→TLDのゾーン
  - TLDのゾーン→SLDのゾーン
  - TLD,SLDのゾーン→各組織のドメインのゾーン
- この分散管理を行う主要な組織について説明する

ドメイン名の管理組織

- IANAとICANN
- レジストリ
- レジストラ
- 個人や組織

IANA

- IANA
  - Internet Assigned Numbers Authority
  - "アイアナ"と読む
- DNSのルートゾーンの管理を行っている組織
- DNSのルートドメインの管理、IPアドレスやAS番号の調整、プロトコルの名前や番号の管理などを行っている
- INTドメインとARPAドメインのゾーンの管理も行っている

IANAとICANN

- IANA自体はARPANET時代から運用されていた
- 1998年にICANN（Internet Corporation for Assigned Names and Numbers、読みは"アイキャン"）が民間の非営利法人として設立
- IANAの業務はICANNに移管され、IANAはICANNの実行部門として組み込まれた

レジストリ

- 各トップレベルドメインのゾーンの管理はレジストリと呼ばれる組織により行われている
- 各レジストリはICANNとの契約に基づき、委任されたトップレベルドメインのゾーンを一元的に管理する
- 例
  - JPドメインのレジストリは「株式会社日本レジストリサービス」（略称JPRS）

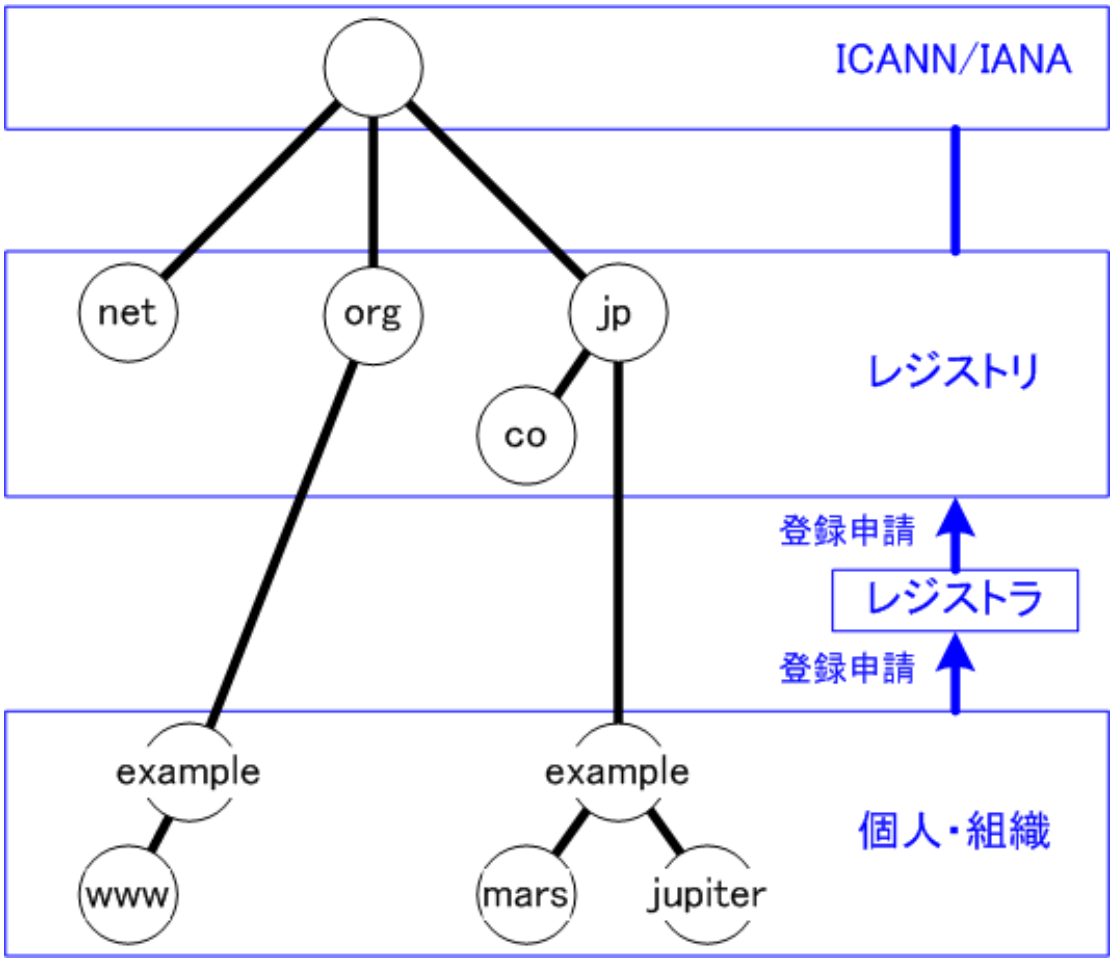
レジストラ

- 個人や組織などからのドメインの登録申請の依頼に対して、レジストリに登録申請する仲介業者

個人や組織

- 個人や組織がドメインの登録申請を行うときには、レジストラに登録申請の仲介を依頼する
- ドメインの登録が行われたら、運用管理しているネームサーバにそのドメインのゾーンの登録を行い、ドメインの運用を行う

ドメイン名を管理する組織



トップ レベル ドメイン

- トップ レベル ドメインの分類
  - ジェネリック トップ レベル ドメイン
    - generic top-level domain、略称gTLD
  - 国コード トップ レベル ドメイン
    - country-code top-level domain、略称ccTLD
  - 予約済みトップ レベル ドメイン

ジェネリック トップレベル ドメイン (gTLD)

ドメイン名	用途	タイプ
ARPA	インターネットのインフラのための ドメイン名空間	インフラストラクチャ
BIZ	ビジネス	制限あり
COM	商用の組織向け。現在は制限なし。	制限なし
INFO	情報提供。現在は制限なし。	制限なし
NAME	個人名	制限あり
NET	ネットワーク・プロバイダ向け。 現在は制限なし。	制限なし
ORG	その他の組織向け。現在は制限なし。	制限なし
PRO	有資格の専門職 （弁護士、公認会計士、医師等）	制限あり

ジェネリック トップレベル ドメイン (gTLD)

ドメイン名	用途	タイプ
AERO	航空運輸業界	スポンサー付き
ASIA	アジア太平洋地域	スポンサー付き
CAT	カタルーニャ言語・文化圏	スポンサー付き
COOP	協同組合	スポンサー付き
EDU	アメリカ合衆国の4年制大学	スポンサー付き
GOV	アメリカ合衆国の（連邦）政府機関	スポンサー付き
INT	国際条約に基づいて設立された組織	スポンサー付き
JOBS	人的資源	スポンサー付き
MIL	アメリカ合衆国軍	スポンサー付き
MOBI	モバイル機器・サービス	スポンサー付き
MUSEUM	博物館・美術館	スポンサー付き
TEL	テレコミュニケーション	スポンサー付き
TRAVEL	旅行業界	スポンサー付き
XXX	アダルト	スポンサー付き

国コード トップ レベルド メイン (ccTLD)

ドメイン名	国名／地域
BR	ブラジル
CA	カナダ
CN	中国
DE	ドイツ
EU	欧州連合
FR	フランス
IN	インド
IT	イタリア
JP	日本
KR	韓国
RU	ロシア
UK	イギリス
US	アメリカ合衆国

国コード トップ レベルド メイン (ccTLD)

ドメイン名	国名／地域
中国, 中國	中国(CN)
香港	香港(HK)
台湾, 台灣	台湾(TW)
한국	韓国(KR)
ไทย	タイ(TH)
ভারত, ભારત, ଭାରତ, ಭಾರತ, ভারত, இந்தியா	インド(IN)
சிங்கப்பூர், 新加坡	シンガポール(SG)
Р Ф	ロシア連邦(RU)
ලංකා	スリランカ(LK)
УКР	ウクライナ(UA)
السعودية	サウジアラビア(SA)

予約済みトップレベルドメイン

ドメイン名	用途
test	DNS関連のコードのテスト用
example	例示用
invalid	不正なドメイン名の例示用
localhost	ループバック用
local	リンクローカル用

参考
RFC 2606 Reserved Top Level DNS Names
RFC 6761 Special-Use Domain Names
RFC 6762 Multicast DNS

評価用 国際化 トップ レベル ドメイン

ドメイン名	ラベル	言語/文字
إختبار	XN--KGBECHTV	アラビア語/アラビア文字
آزمایشی	XN--HGBK6AJ7F53BBA	ペルシャ語/アラビア文字
测试	XN--0ZWM56D	中国語/簡体字
測試	XN--G6W251D	中国語/繁体字
испытание	XN--80AKHBYKNJ4F	ロシア語/キリル文字
परीक्षा	XN--11B5BS3A9AJ6G	ヒンディー語/ディーヴァナーガリー
δοκιμή	XN--JXALPDLP	ギリシア語/ギリシア文字
테스트	XN--9T4B11YI5A	朝鮮語/ハングル
טעסט	XN--DEBA0AD	イディッシュ語/ヘブライ文字
テスト	XN--ZCKZAH	日本語/片仮名
பரிட்சை	XN--HLCJ6AYA9ESC7A	タミル語/タミル文字

ARPAドメイン

- "ARPA"の経緯
  - "ARPA"というドメイン名は"ARPANET"を由来とする
  - HOSTS.TXTからDNSへの移行時にARPANETの各ホストを一時的に格納するドメイン名空間
  - IPアドレスからホスト名を逆引きするときに使うドメイン名空間"IN-ADDR.ARPA"

ARPAドメイン

- 現在
  - インターネットのインフラのためのドメイン名空間
    - E164.ARPA, IN-ADDR.ARPA, IP6.ARPA, IRIS.ARPA, URI.ARPA, URN.ARPA
  - "ARPA"は"Address and Routing Parameter Area"の略語に再定義
    - RFC 3172 Management Guidelines & Operational Requirements for the Address and Routing Parameter Area Domain ("arpa")

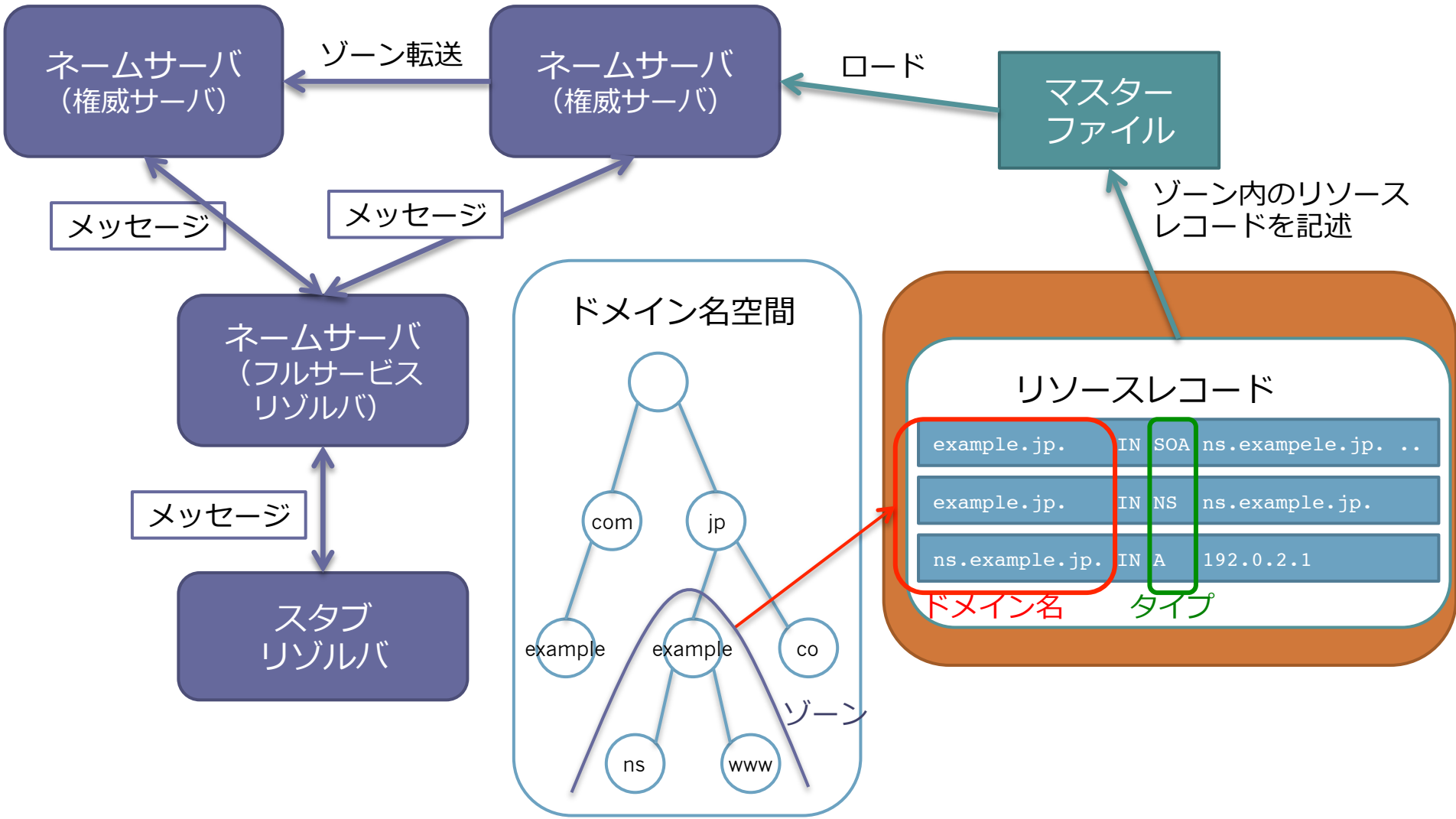
このセクションのまとめ

- ゾーン
- ゾーンの分割と権威の委任
- ドメイン名の管理組織
- トップレベルドメイン
- ARPAドメイン

リソース レコード

リソースレコードの意味と記述方法について理解してもらう。

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
メッセージ
メッセージ
ゾーン内のリソース
レコードを記述
ネームサーバ
（フルサービス
リゾルバ）
メッセージ
スタブ
リゾルバ
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ

リソースレコードとは

- ホスト名やIPアドレスといった資源に関するデータをリソースレコードという
- RRと略す
- 同じドメイン名とリソースタイプの集まりをリソース レコード セットと呼び、RRsetと略す

リソースレコードの形式

- 形式
  - OWNER TTL CLASS TYPE RDATA
- 例
  - `www.example.jp. 3600 IN A 192.0.2.1`

リソースレコードの形式

- OWNER TTL CLASS TYPE RDATA
  - `www.example.jp. 3600 IN A 192.0.2.1`
- OWNER（オーナー）
  - このリソースレコードがあるドメイン名

リソースレコードの形式

- OWNER TTL CLASS TYPE RDATA
  - `www.example.jp. 3600 IN A 192.0.2.1`
- TTL（生存期間）
  - リソースレコードの生存期間。秒単位の32ビット整数。
  - リゾルバがキャッシュするときに使う。TTLはRRが破棄されるまでにキャッシュしてよい期間を示す。
  - 値の定義
    - 符号無し整数
    - 最小値: 0 （0はキャッシュ禁止を表す）
    - 最大値: 2147483647 (2^31 − 1)
    - 最上位ビットが1であるときにはTTLを0と扱うべき
  - 参考: RFC 2181 Clarifications to the DNS Specification "8. Time to Live (TTL)"

リソースレコードの形式

- OWNER TTL CLASS TYPE RDATA
  - `www.example.jp. 3600 IN A 192.0.2.1`
- CLASS（クラス）
  - プロトコルファミリーを識別する符号化された16ビットの数
  - テキスト表現としてはニーモニックが使われる

ニーモニック	値	説明
IN	1	Internet
CH	3	Chaos
HS	4	Hesiod

本来の用途とは異なり、現在はネームサーバの情報の取得に使われている。
$ dig version.bind. TXT CH

リソースレコードの形式

- OWNER TTL CLASS TYPE RDATA
  - `www.example.jp. 3600 IN A 192.0.2.1`
- TYPE（タイプ）
  - このリソースレコードのリソースのタイプを識別する符号化された16ビットの値。
  - テキスト表現としてはニーモニックが使われる
    - A, CNAME, MX, NS, PTR, SOA, TXT

主要なタイプとニーモニック

ニーモニック	値	説明
A	1	IPv4のIPアドレス
NS	2	ゾーンの権威ネームサーバ
CNAME	5	別名に対する正式名
SOA	6	ゾーンの権威の開始
PTR	12	IPアドレスに対するホスト名を示すポインタ
MX	15	メールの送信先
TXT	16	テキスト
AAAA	28	IPv6のIPアドレス

リソースレコードの形式

- OWNER TTL CLASS TYPE RDATA
  - `www.example.jp. 3600 IN A 192.0.2.1`
- RDATA（資源データ）
  - タイプとクラスに依存するデータ

主要なタイプ（再掲）

ニーモニック	値	説明
A	1	IPv4のIPアドレス
NS	2	ゾーンの権威ネームサーバ
CNAME	5	別名に対する正式名
SOA	6	ゾーンの権威の開始
PTR	12	IPアドレスに対するホスト名を示すポインタ
MX	15	メールの送信先
TXT	16	テキスト
AAAA	28	IPv6のIPアドレス

SOA (Start Of Authority)

- ゾーンの権威の開始
- ゾーンそのものについての情報を記載する
- セカンダリ ネームサーバへのゾーン転送はこのRRで設定した値に基づいて動作する

SOAの記述方法

- *OWNER TTL* IN SOA *MNAME RNAME* (
  SERIAL
  REFRESH
  RETRY
  EXPIRE
  MINIMUM)
- OWNER
  - ゾーン名

SOAの記述方法

- MNAME
  - このゾーンのデータのオリジナルあるいはプライマリ（プライマリ マスター）であるネームサーバーのドメイン名。
  - プライマリ マスター
    - ゾーン転送におけるNOTIFYの送信元
    - DNS UPDATEのリクエスト先
  - RFC 2181 Clarifications to the DNS Specification "7.3. The SOA.MNAME field"
    - SOAレコードのMNAMEフィールドはゾーンのマスターサーバの名前を設定する。
    - ゾーン自体の名前を書くべきではない。

SOAの記述方法

- RNAME
  - このゾーンの責任者のメールアドレス。
  - メールアドレスの"@"を"."に置き換える。
    - 例）"foo@example.com"は"foo.example.com."に。

SOAの記述方法

- SERIAL（シリアル値）
  - ゾーンのオリジナルコピーの符号無し32ビットバージョン番号。ゾーン転送はこの値を維持する。
  - この値は周回し、sequence space arithmeticを使って比較する。
    - RFC 1982 Serial Number Arithmeticで比較について詳細な説明がある。
- REFRESH（更新）
  - セカンダリ ネームサーバがプライマリ ネームサーバに更新を確認する間隔
- RETRY（再試行）
  - セカンダリ ネームサーバが更新に失敗した後に再試行する間隔

SOAの記述方法

- EXPIRE（満期）
  - セカンダリ ネームサーバが更新できないときに、データを期限切れにするまでの上限値
- MINIMUM（最小）
  - 元々の意味はこのゾーンのRRに適応される最小のTTL
  - RFC 2308 "Negative Caching of DNS Queries (DNS NCACHE)"により再定義され、現在はネガティブキャッシュ（存在しないことのキャッシュ）のTTLとして使われている

SOAの記述例

- 記述例

```
example.jp. IN SOA ns.example.jp. hostmaster.example.jp. (
                          2013071901 ;serial
                          3600       ;refresh
                          600        ;retry
                          604800     ;expire
                          900 )      ;minimum
```

- 記述例の説明
  - プライマリ ネームサーバは"ns.example.jp."
  - 責任者のメールアドレスは"hostmaster@example.jp"

NS (Name Server)

- ゾーンの権威ネームサーバ
- NSレコードの値には正式名を記述する
  - CNAMEで定義される別名を使ってはいけない。

NSの記述方法

- *OWNER TTL* IN NS *NSDNAME*
  - NSDNAME（ネームサーバの名前）
    - ゾーンの権威ネームサーバのドメイン名
- 記述例

```
example.jp. 86400 IN NS ns.example.jp.
```

- 記述例の説明
  - ゾーン"example.jp."の権威ネームサーバは"ns.example.jp."である。

A (Address)

- IPv4のIPアドレス

Aの記述方法

- *OWNER TTL* IN A *ADDRESS*
  - ADDRESS（アドレス）
    - IPv4のIPアドレスをドット付き10進記法で記述する
- 記述例

```
www.example.jp. 86400 IN A 192.0.2.1
```

AAAA

- IPv6のIPアドレスを記述する
- 「クワッド エイ」と読む

AAAAの記述方法

- *OWNER TTL* IN AAAA *ADDRESS*
  - ADDRESS
    - IPv6のIPアドレス
- 記述例

```
www.example.jp. 86400 IN AAAA 2001:db8:dead:beef::1
```

- 記述例の説明
  - www.example.jp.のIPv6アドレスは"2001:db8:dead:beef::1"である

CNAME (Canonical Name)

- 別名に対する正式名を指定する
- 別名を定義するために使われる。
- 制限
  - CNAMEで定義した別名をNSやMXなどのデータには利用してはいけない

CNAMEの記述例

- *OWNER TTL* IN CNAME *CNAME*
  - CNAME
    - 別名に対する正式名を記述する
- 記述例

```
foo.example.jp. IN A     192.0.2.1
www.example.jp. IN CNAME foo.example.jp.
```

- 記述例の説明
  - "foo.example.jp."の別名として"www.example.jp."を定義する。

PTR (Pointer)

- IPアドレスに対するホスト名を示すポインタ
- 逆引き（IPアドレスからホスト名を求める）のために使われる

PTR (Pointer)のIPアドレスの表記

- IPv4アドレスはIPアドレスを逆の順番にしてin-addr.arpa.を付ける
  - 192.0.2.1の表記
    - 1.2.0.192.in-addr.arpa.
- IPv6アドレスはIPアドレスを逆の順番にして、ip6.arap.を付ける
  - 2001:db8:dead:beef::1の表記
    - 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa.

PTRの記述方法

- *OWNER TTL* IN PTR *PTRDNAME*
  - OWNER
    - in-addr.arpa.やip6.arpa.の名前空間でのIPアドレスの表記
  - PTRDNAME
    - IPアドレスに対するドメイン名

PTRの記述例

- IPv4の場合の記述例
  - `1.2.0.192.in-addr.arpa. IN PTR www.example.jp.`
  - IPアドレス"192.0.2.1"のホスト名は"www.example.jp."である。
- IPv6の場合の記述例
  - `1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.f.e.e.b.d.a.e.d.8.b.d.0.1.0.0.2.ip6.arpa. IN PTR www.example.jp.`
  - IPv6アドレス"2001:db8:dead:beef::1"のホスト名は"www.example.jp."である。

PTR (Pointer)

- 制限
  - PTRレコードのデータにはCNAMEで定義される別名を使ってはいけない
- よくある間違い
  - 誤: 一つのIPアドレスに対してPTRレコードは一つだけしか記述できない
  - 正: 一つのIPアドレスに対して複数のPTRレコードを記述できる

MX (Mail Exchanger)

- メールの送信先
- 制限
  - MXレコードの値にはCNAMEで定義される別名を使ってはいけない。

MXの記述方法

- *OWNER TTL* IN MX *PREFERENCE EXCHANGE*
  - OWNER
    - メールの宛先メールアドレスのドメイン名
  - PREFERENCE
    - 優先度を示す数値
    - 小さい方が優先度が高い
  - EXCHANGE
    - 所有者名のドメイン名に対するメールの送信先のサーバのドメイン名

MXの記述例

- 記述例

```
example.jp.  IN MX 10 mx1.example.jp.
             IN MX 20 mx2.example.jp.
```

- 記述例の説明
  - example.jp.ドメイン宛のメールの送信先のメールサーバはmx1.example.jp.とmx2.example.jp.である
  - mx1.example.jp.の優先度が高い

TXT (Text)

- テキスト
- 任意の文字列を記述できる
- 様々な目的で利用される
  - SPF
  - DKIM
  - RBL

TXTの記述方法

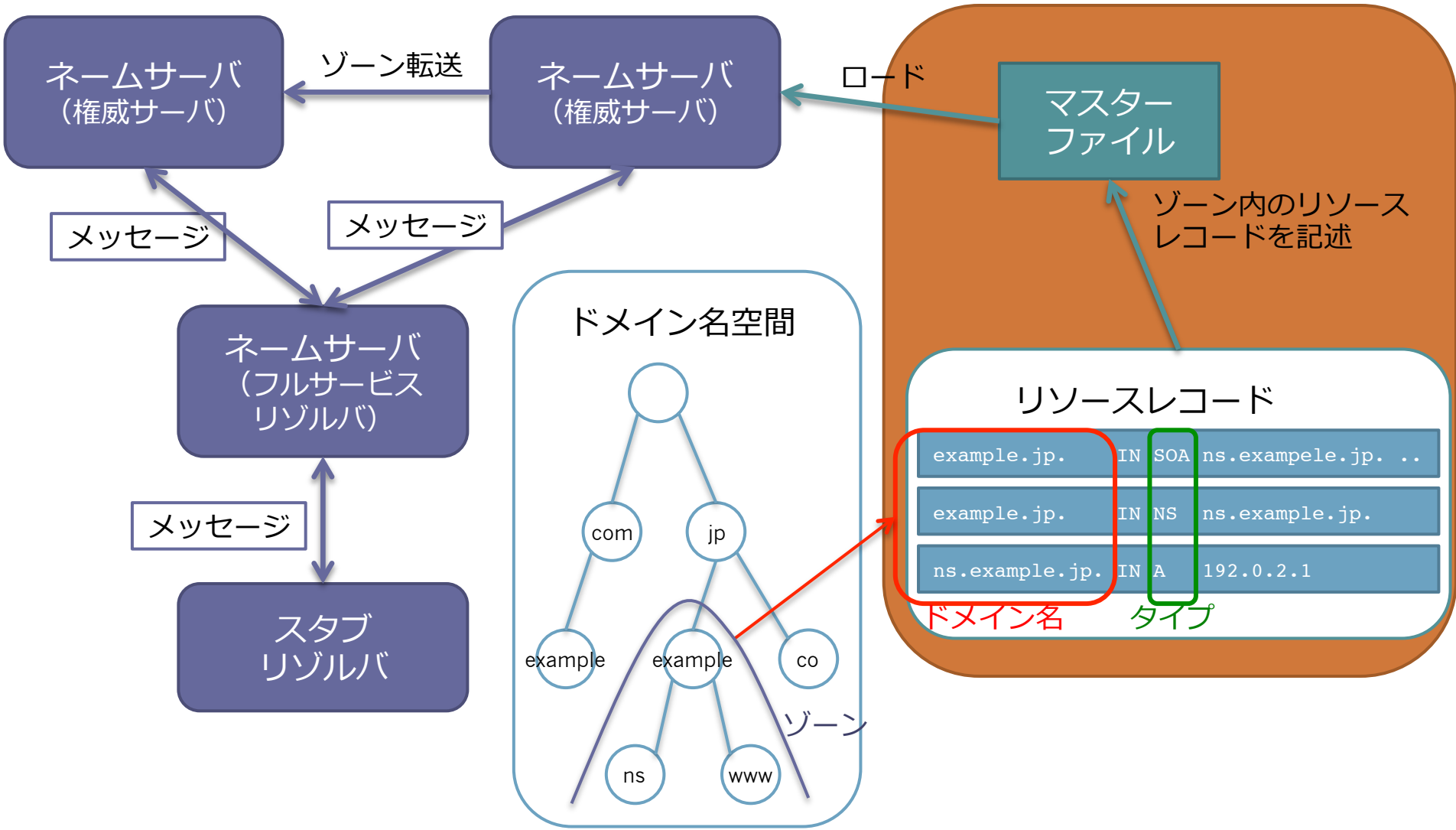
- *OWNER TTL* IN TXT *TXT-DATA*
  - TXT-DATA
    - 一つ以上の文字
- 記述例
  - example.jp. IN TXT "nullpo. ga."

このセクションのまとめ

- リソースレコード
- 各タイプのリソースレコードについての説明
  - SOA, NS, A, AAAA, CNAME, PTR, MX, TXT

マスターファイル

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
ゾーン内のリソース
レコードを記述
メッセージ
メッセージ
ネームサーバ
（フルサービス
リゾルバ）
メッセージ
スタブ
リゾルバ
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ

マスターファイル

- マスターファイルはゾーンのリソースレコードの集まりを記述したテキストファイル
- ゾーンファイルとも呼ばれる。
- 権威ネームサーバはこのマスターファイルをロードしてサービスを提供する。

マスターファイルの記述例

```
$ORIGIN example.jp.
$TTL 86400
@     IN  SOA ns.example.jp. hostmaster.example.jp. (
              2011061801 ; Serial
              3600       ; Refresh
              900        ; Retry
              604800     ; Expire
              3600 )     ; Minimum
      IN  NS  ns.example.jp.
      IN  MX  10 mx.example.jp.
      IN  A   192.0.2.2
ns    IN  A   192.0.2.1
www   IN  A   192.0.2.2
mx    IN  A   192.0.2.3
```

マスターファイルの形式

- ";"はコメントの開始を意味する。
- 空行は無視される。

マスターファイルの形式

- RRのエントリは1行で示される。

```
example.com. 172800 IN NS a.iana-servers.net.
```

- 複数行になる場合には括弧を使う。

```
example.com.    3600 IN SOA dns1.icann.org. (
                hostmaster.icann.org.
                2012080872 7200 3600 1209600 3600 )
```

- 行の先頭はRRのオーナー。

```
example.com. 172800 IN NS a.iana-servers.net.
```

- 空白で始まる行は、オーナーが前のRRと同じと想定される。

```
example.com. 172800 IN NS a.iana-servers.net.
             172800 IN NS b.iana-servers.net.
```

マスターファイルの形式

- $ORIGIN ドメイン名
  - オリジンを指定したドメイン名に変更する。
    - ・オリジンは相対ドメイン名を補完するドメイン名
- $INCLUDE ファイル名
  - この場所に指定したファイル名のファイルを挿入する。
  - 挿入されたファイルにより親ファイルのオリジンには影響を与えない。
- $TTL TTL
  - デフォルトのTTLを指定した値に変更する。

マスターファイルの形式

- TTLとクラス"IN"は省略可能。TTLを省略する$TTLで定義した値になる。

```
example.com. NS a.iana-servers.net.
```

- "@"はオリジン（相対ドメイン名を補完するドメイン名）を意味する。オリジンのデフォルトはゾーン頂点のドメイン名。example.com.ゾーンの次の２行は同じ意味になる。

```
example.com. 172800 IN NS a.iana-servers.net.
@            172800 IN NS b.iana-servers.net.
```

マスターファイルの記述例

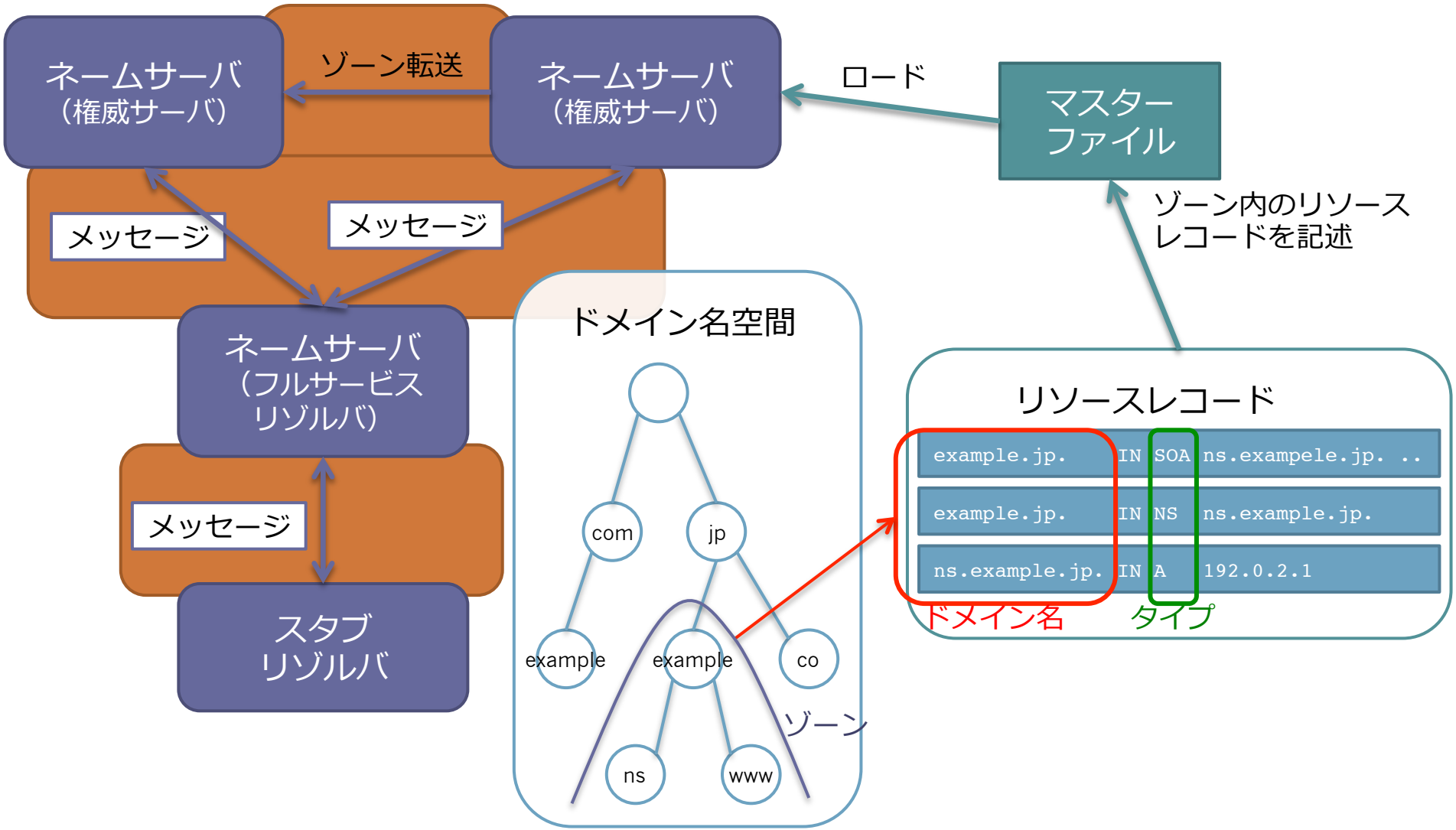
```
$ORIGIN example.jp.
$TTL 86400
@     IN  SOA ns.example.jp. hostmaster.example.jp. (
              2011061801 ; Serial
              3600       ; Refresh
              900        ; Retry
              604800     ; Expire
              3600 )     ; Minimum
      IN  NS  ns.example.jp.
      IN  MX  10 mx.example.jp.
      IN  A   192.0.2.2
ns    IN  A   192.0.2.1
www   IN  A   192.0.2.2
mx    IN  A   192.0.2.3
```

このセクションのまとめ

- マスターファイルの記述方法

DNSメッセージ

DNSの構成要素

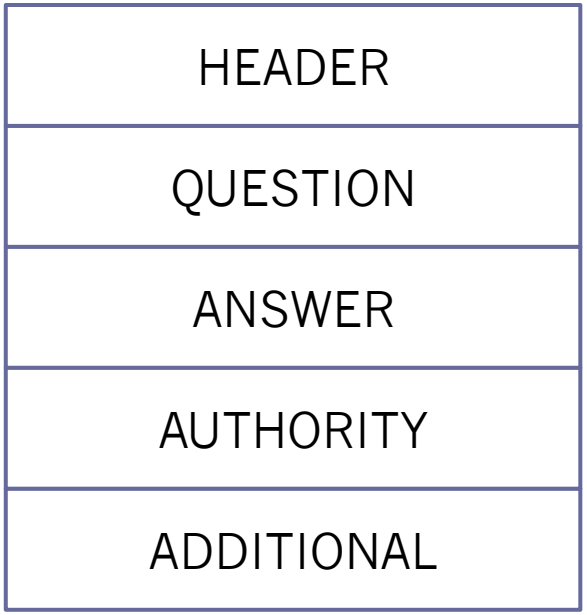

ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
ゾーン内のリソース
レコードを記述
メッセージ
メッセージ
ネームサーバ
（フルサービス
リゾルバ）
メッセージ
スタブ
リゾルバ
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ

DNSメッセージ

- DNSの問い合わせと応答はDNSメッセージで運ばれる。
- DNSメッセージのサイズ
  - UDPメッセージは512オクテット以下に制限される。
  - TCPでは512オクテット以上のメッセージを送ることができる。
  - EDNS0を使うと、UDPで512オクテットより大きいメッセージを送ることができる。

メッセージフォーマット

- ヘッダセクションとリソースレコードを扱う４つのセクションから成り立つ




```
$ dig @ns.example.jp. example.jp. +norec

; <<>> DiG 9.8.3-P1 <<>> @ns.example.jp. example.jp.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37953
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;example.jp.                    IN      A

;; ANSWER SECTION:
example.jp.             3600    IN      A       192.0.2.4

;; AUTHORITY SECTION:
example.jp.             86400   IN      NS      ns.example.jp.
example.jp.             86400   IN      NS      ns2.example.jp.

;; ADDITIONAL SECTION:
ns.example.jp.          86400   IN      A       192.0.2.1
ns2.example.jp.         86400   IN      A       192.0.2.2
```

セクション

セクション	説明
HEADER	いくつかの固定フィールドと問い合わせパラメータ
QUESTION	問い合わせ名と他の問い合わせパラメータ
ANSWER	回答のRR
AUTHORITY	他の権威サーバーを示すRR。 ANSWERセクションの権威データのSOA RRでもよい。
ADDITIONAL	他のセクションのRRを使う際に役に立つかもしれないRR

ヘッダ セクション フォーマット

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ID															
QR	OPCODE				AA	TC	RD	RA	Z	AD	CD	RCODE			
QDCOUNT															
ANCOUNT															
NSCOUNT															
ARCOUNT															

項目	説明
ID	16ビットの識別子。
QR	Query(0)かResponse(1)かを示す1ビット 0: Query 1: Response
OPCODE	問い合わせの種類を示す4ビット。 0: Query 2: Status 4: Notify 5: Update

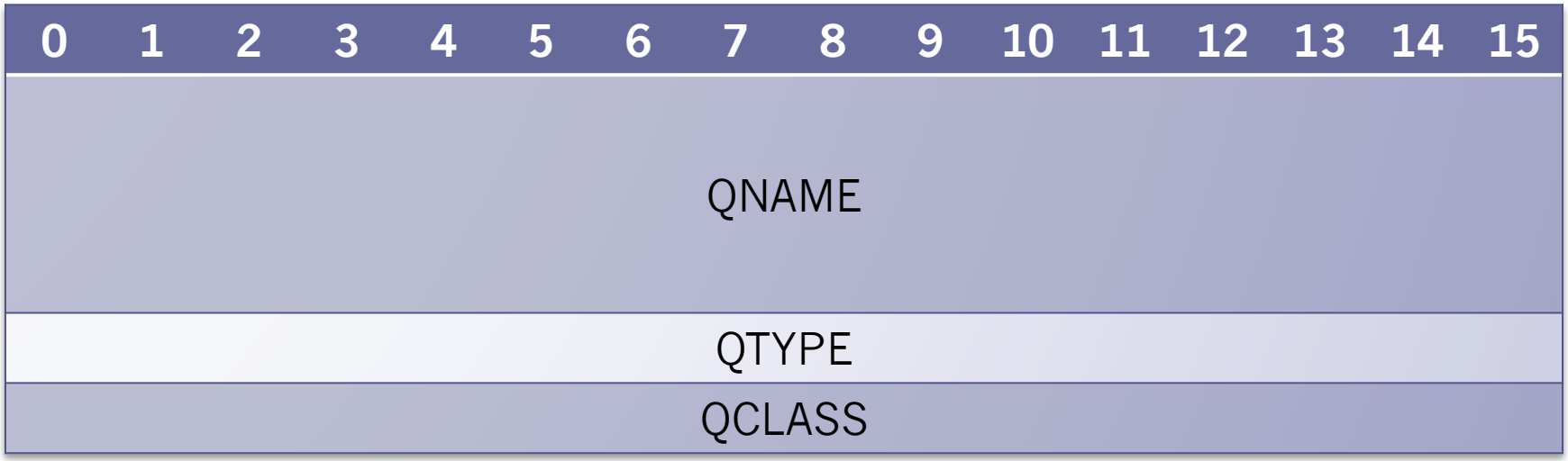
項目	説明
AA	Authoritative Answer 対応したネームサーバがQUESTIONセクションのドメイン名に対する権威を持っているかを示す。
TC	TrunCation メッセージが大きくて切り詰められたことを示す。
RD	Recursion Desired ネームサーバに再帰検索要求であることを指示する。
RA	Recursion Available ネームサーバが再帰検索要求を処理できるかを示す。
Z	将来のための予約。0にする。
AD	Authentic Data 問い合わせにおいては、DNSSECの検証を指示する。 応答においては、DNSSECの検証に成功したかを示す。 成功したら1、失敗した、あるいは検証していなければ0
CD	Checking Disabled ネームサーバにDNSSECの検証を行わないことを指示する。

項目	説明
RCODE	応答コード 0: No error condition (NoError) 1: Format error (FormErr) 2: Server failure (ServFail) 3: Name Error (NXDomain) 4: Not Implemented (NotImp) 5: Refused
QDCOUNT	QUESTIONセクションのエントリーの数を示す符号無し16ビット整数
ANCOUNT	ANSWERセクションのエントリーの数を示す符号無し16ビット整数
NSCOUNT	AUHTORITYレコードセクションのリソースレコードの数を示す符号無し16ビット整数
ARCOUNT	ADDITIONALレコードセクションのリソースレコードの数を示す符号無し16ビット整数

digの結果におけるヘッダセクション

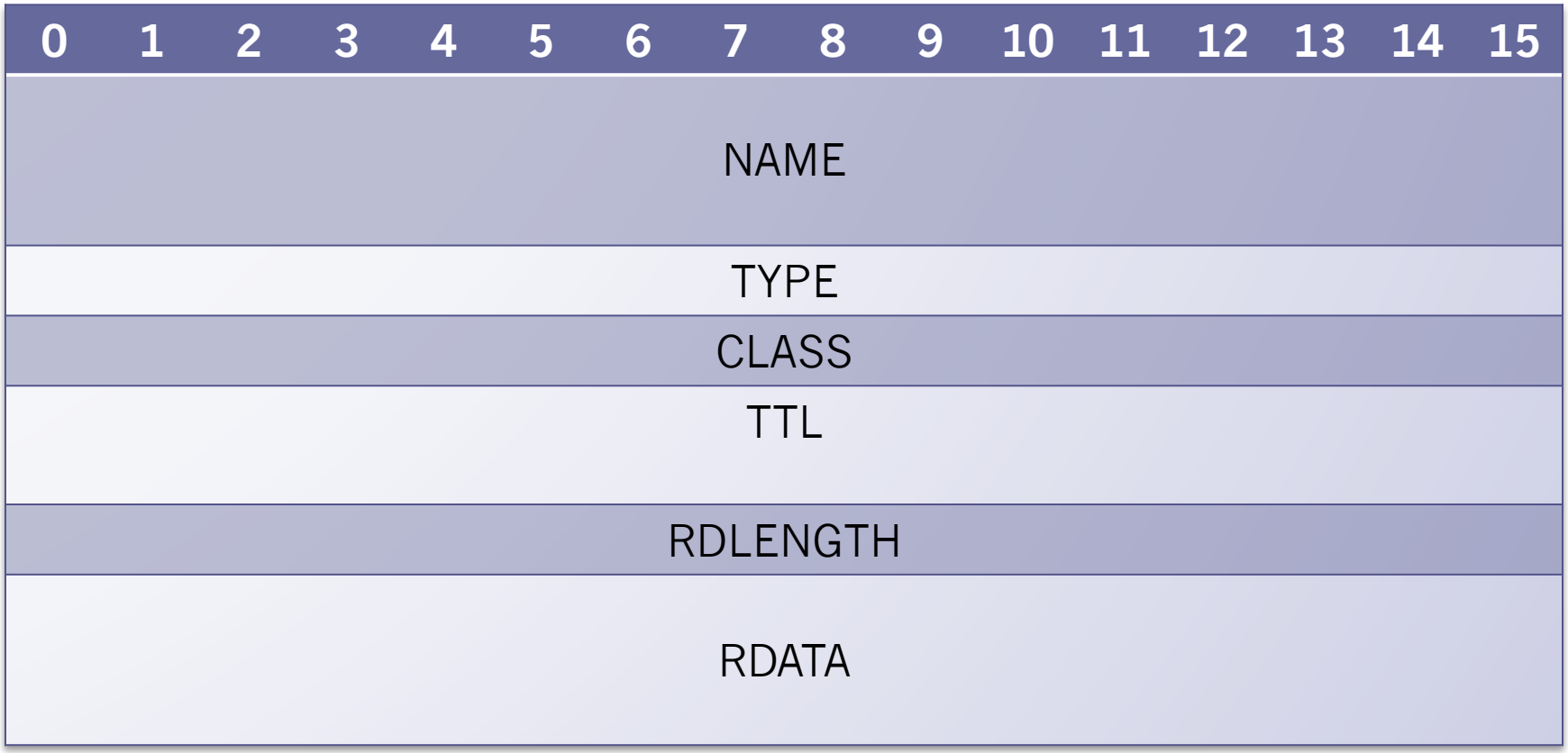
```
$ dig @ns.example.jp. example.jp. +norec

; <<>> DiG 9.8.3-P1 <<>> @ns.example.jp. example.jp.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37953
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2
```

QUESTIONセクション フォーマット



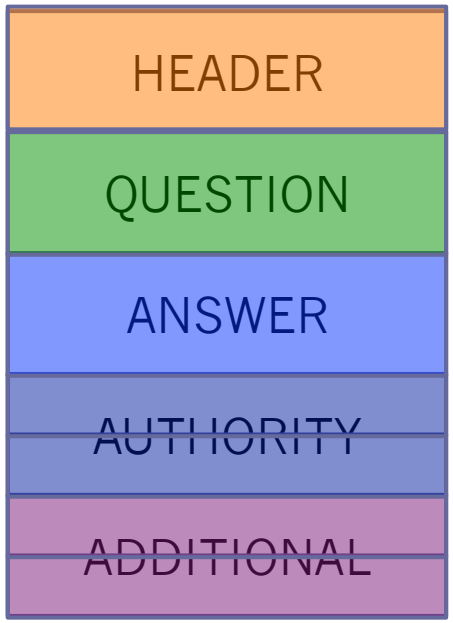
項目	説明
QNAME	ドメイン名
QTYPE	問い合わせのタイプを示す16ビット。 通常のタイプに加えて以下のものが使用できる。 TSIG (250) IXFR (251) AXFR (252) ANY (255)
QCLASS	問い合わせのクラスを示す16ビット。 通常のクラスに加えて以下のものが使用できる。 ANY (255)

リソースレコード フォーマット



項目	説明
NAME	オーナー
TYPE	タイプ
CLASS	クラス
TTL	TTL
RDLENGTH	RDATAの長さ（オクテット）を示す符号なし16ビット整数。
RDATA	リソースのデータ

digによるDNSメッセージの確認



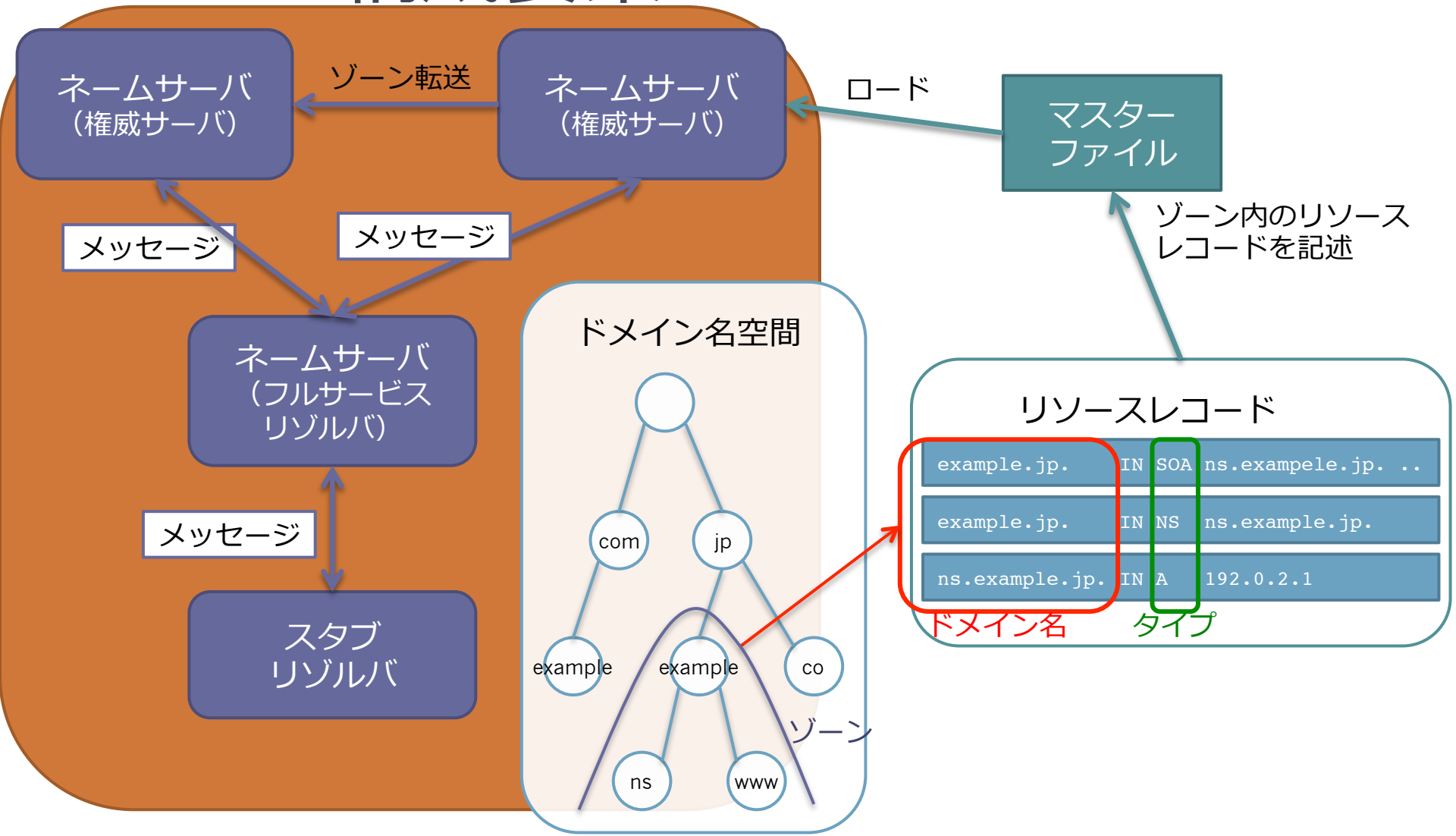

```
$ dig @ns.example.jp. example.jp. +norec

; <<>> DiG 9.8.3-P1 <<>> @ns.example.jp. example.jp.
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37953
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;example.jp.                    IN      A

;; ANSWER SECTION:
example.jp.             3600    IN      A       192.0.2.4

;; AUTHORITY SECTION:
example.jp.             86400   IN      NS      ns.example.jp.
example.jp.             86400   IN      NS      ns2.example.jp.

;; ADDITIONAL SECTION:
ns.example.jp.          86400   IN      A       192.0.2.1
ns2.example.jp.         86400   IN      A       192.0.2.2
```

このセクションのまとめ

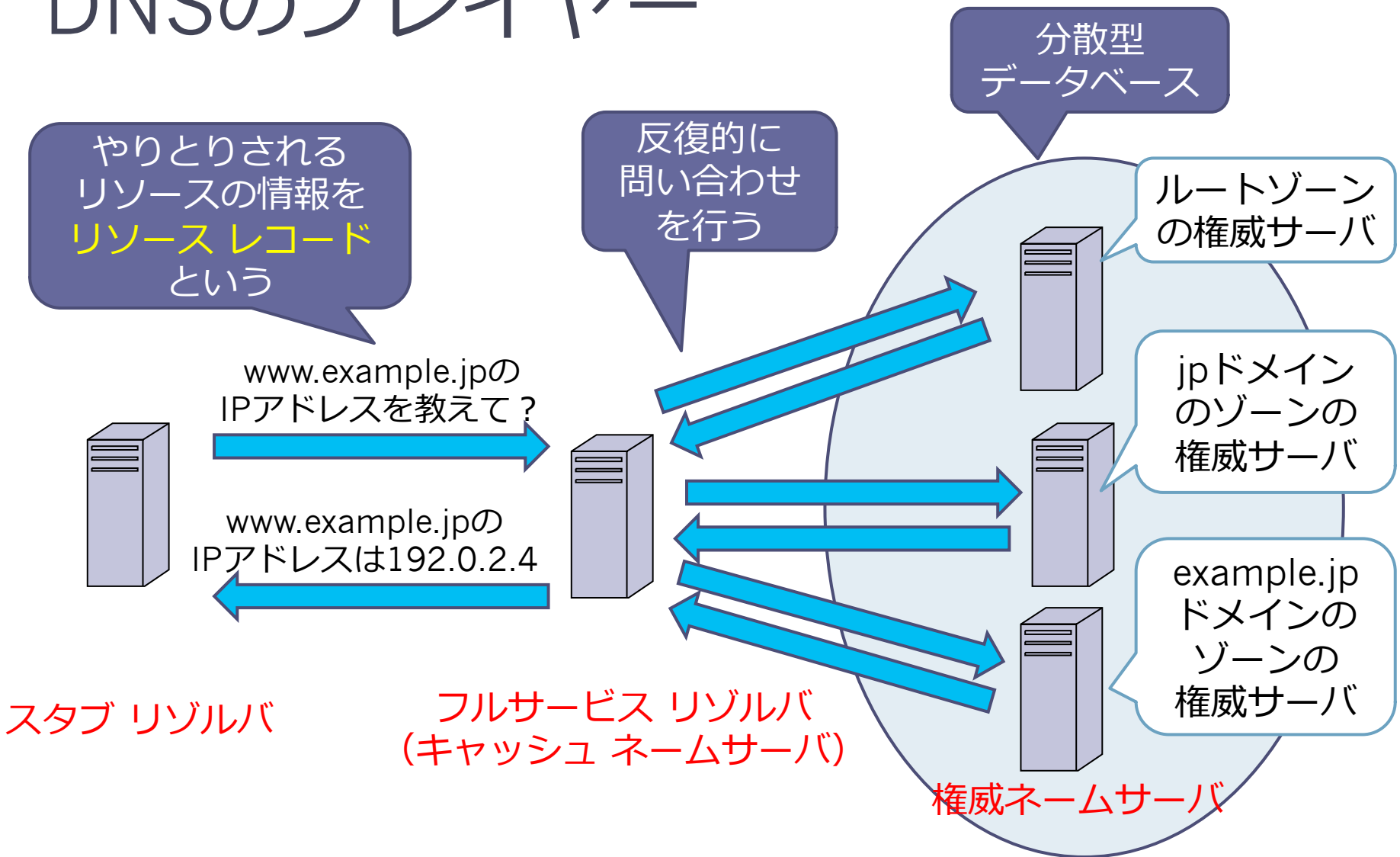
- DNSメッセージのフォーマット

リゾルバとネームサーバ

DNSの構成要素


ネームサーバ
（権威サーバ）
ゾーン転送
ネームサーバ
（権威サーバ）
ロード
マスター
ファイル
ゾーン内のリソース
レコードを記述
メッセージ
メッセージ
ネームサーバ
（フルサービス
リゾルバ）
メッセージ
スタブ
リゾルバ
ドメイン名空間
com
jp
example
example
co
ns
www
ゾーン
リソースレコード
example.jp. IN SOA ns.exampele.jp. ..
example.jp. IN NS ns.example.jp.
ns.example.jp. IN A 192.0.2.1
ドメイン名
タイプ

DNSのプレイヤー


やりとりされる
リソースの情報を
リソース レコード
という
反復的に
問い合わせ
を行う
分散型
データベース
ルートゾーン
の権威サーバ
jpドメイン
のゾーンの
権威サーバ
example.jp
ドメインの
ゾーンの
権威サーバ
www.example.jpの
IPアドレスを教えて？
www.example.jpの
IPアドレスは192.0.2.4
スタブ リゾルバ
フルサービス リゾルバ
（キャッシュ ネームサーバ）
権威ネームサーバ

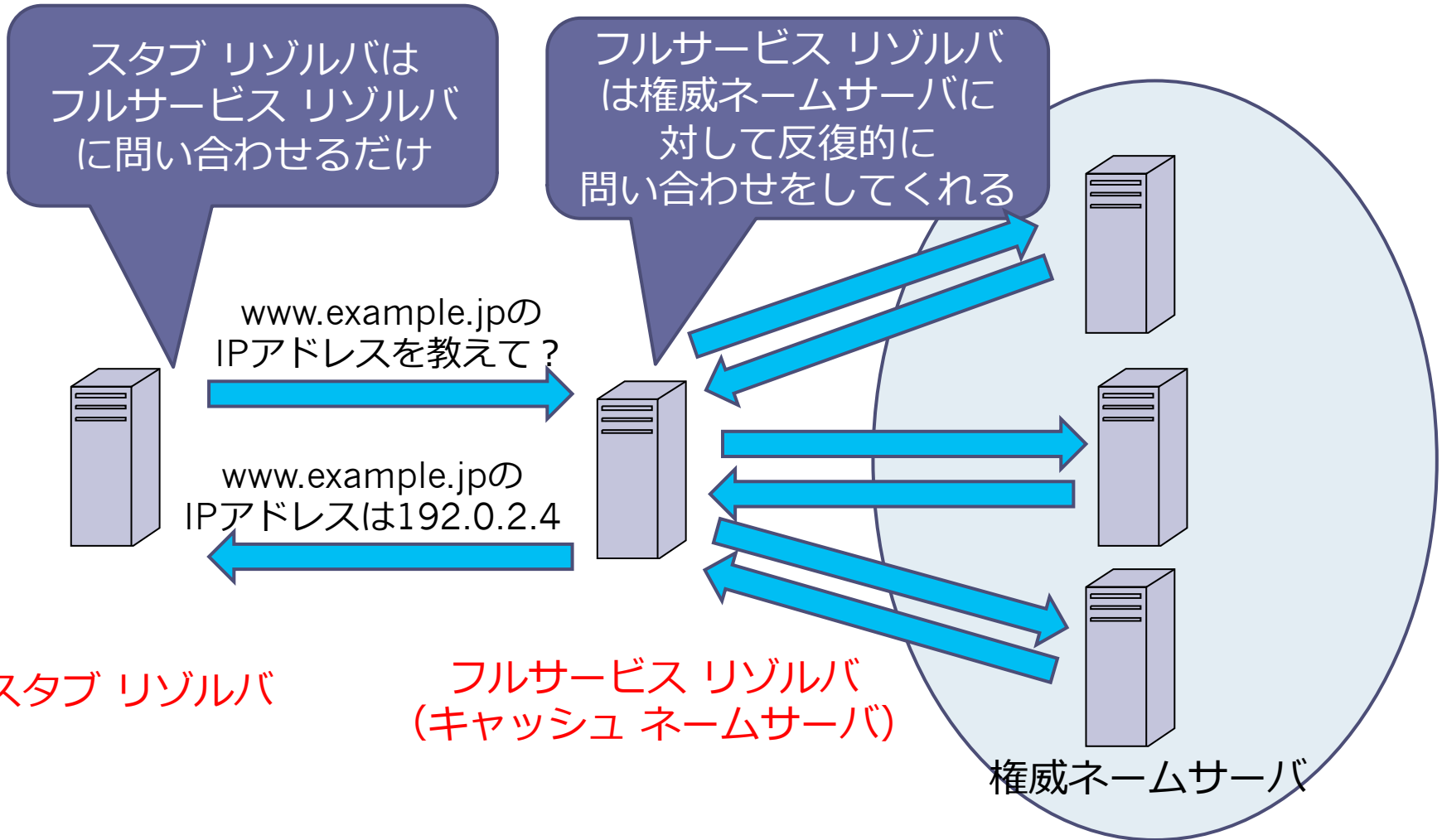
スタブ リゾルバ

- 名前解決を要求する（クライアント）側の機能
- OSやライブラリの機能（関数/API）として実装されている
- フルサービス リゾルバに再帰検索要求（RDフラグあり）リクエストを送って、名前の解決を行う
  - OSのネットワーク設定の「ネームサーバ」欄に利用するフルサービス リゾルバのIPアドレスを設定する
  - UNIX系OSの場合は/etc/resolv.conf

スタブ リゾルバ

- 同じ要求の繰り返しのリクエストを避けるために、キャッシュ機能を持ってもよい(MAY)

スタブ リゾルバ


スタブ リゾルバは
フルサービス リゾルバ
に問い合わせるだけ
フルサービス リゾルバ
は権威ネームサーバに
対して反復的に
問い合わせをしてくれる
www.example.jpの
IPアドレスを教えて？
www.example.jpの
IPアドレスは192.0.2.4
スタブ リゾルバ
フルサービス リゾルバ
（キャッシュ ネームサーバ）
権威ネームサーバ

フル サービス リゾルバ

- リゾルバ サービスの完全な実装であり、自身で名前の解決を行う機能
- スタブ リゾルバから再帰検索要求（RDフラグあり）のリクエストを受け取り、
権威ネームサーバに対して（RDフラグなしで）、
反復的な問い合わせを行い、
名前の解決を行う。
- 同じ要求の繰り返しのリクエストを避けるために、キャッシュ機能を持たなければならない(MUST)

フル サービス リゾルバ

- クライアントに対してサービスを提供するサーバである
  - クライアントのOSのネットワーク設定の「ネームサーバ」欄にこのフルサービス リゾルバのIPアドレスを設定する
  - UNIX系OSの場合は/etc/resolv.conf

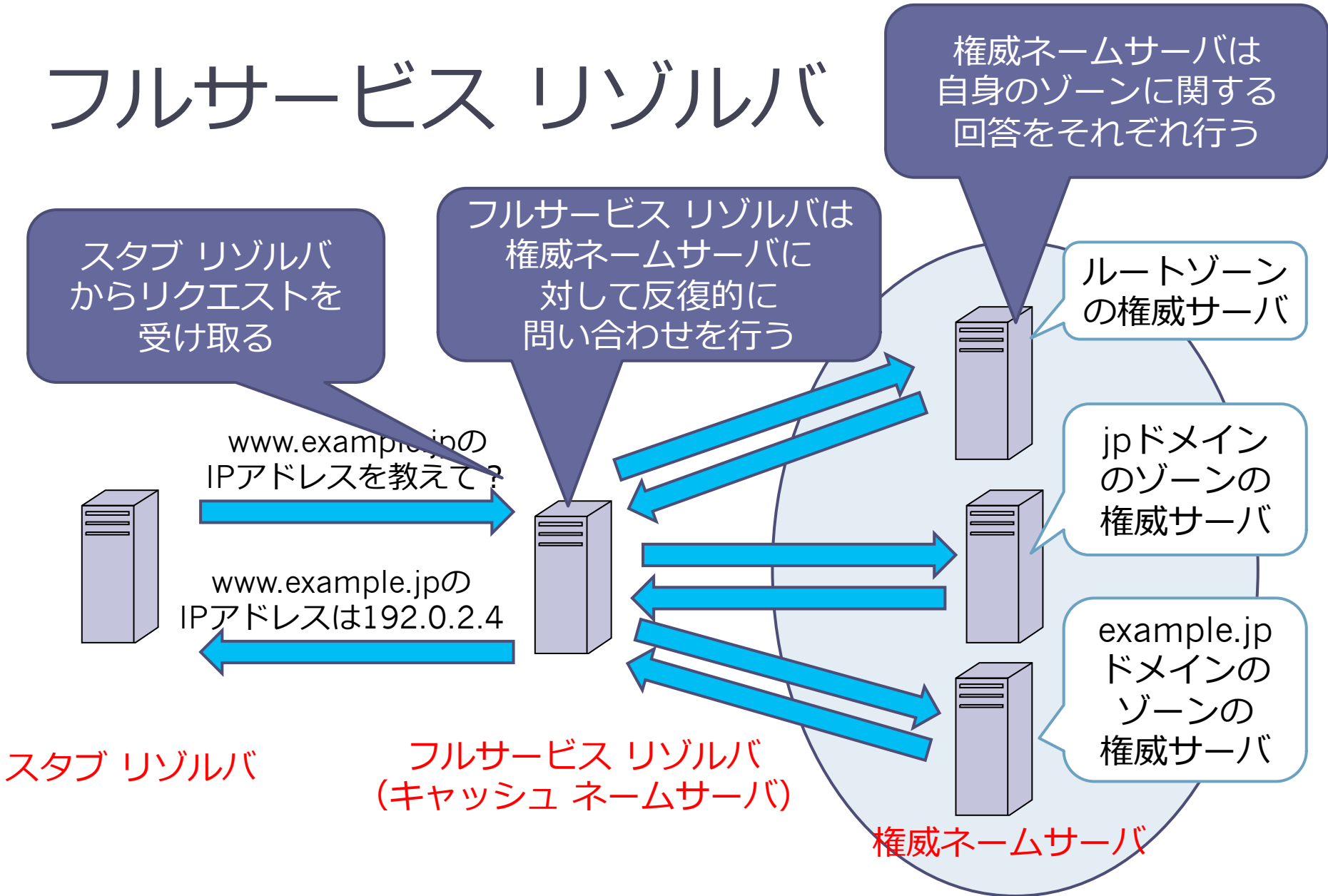
フル サービス リゾルバ

- キャッシュ ポイズニング
  - 誰でもアクセスできるリゾルバ（オープン リゾルバ）はキャッシュ ポイズニングの危険性が高くなる。
  - クライアントに対するサービスであるため、決められたクライアント以外にアクセスを許可する必要はない。
  - オープンリゾルバにならないように、アクセス制御を行う必要がある。
    - LAN内に置いてもいいんだよ

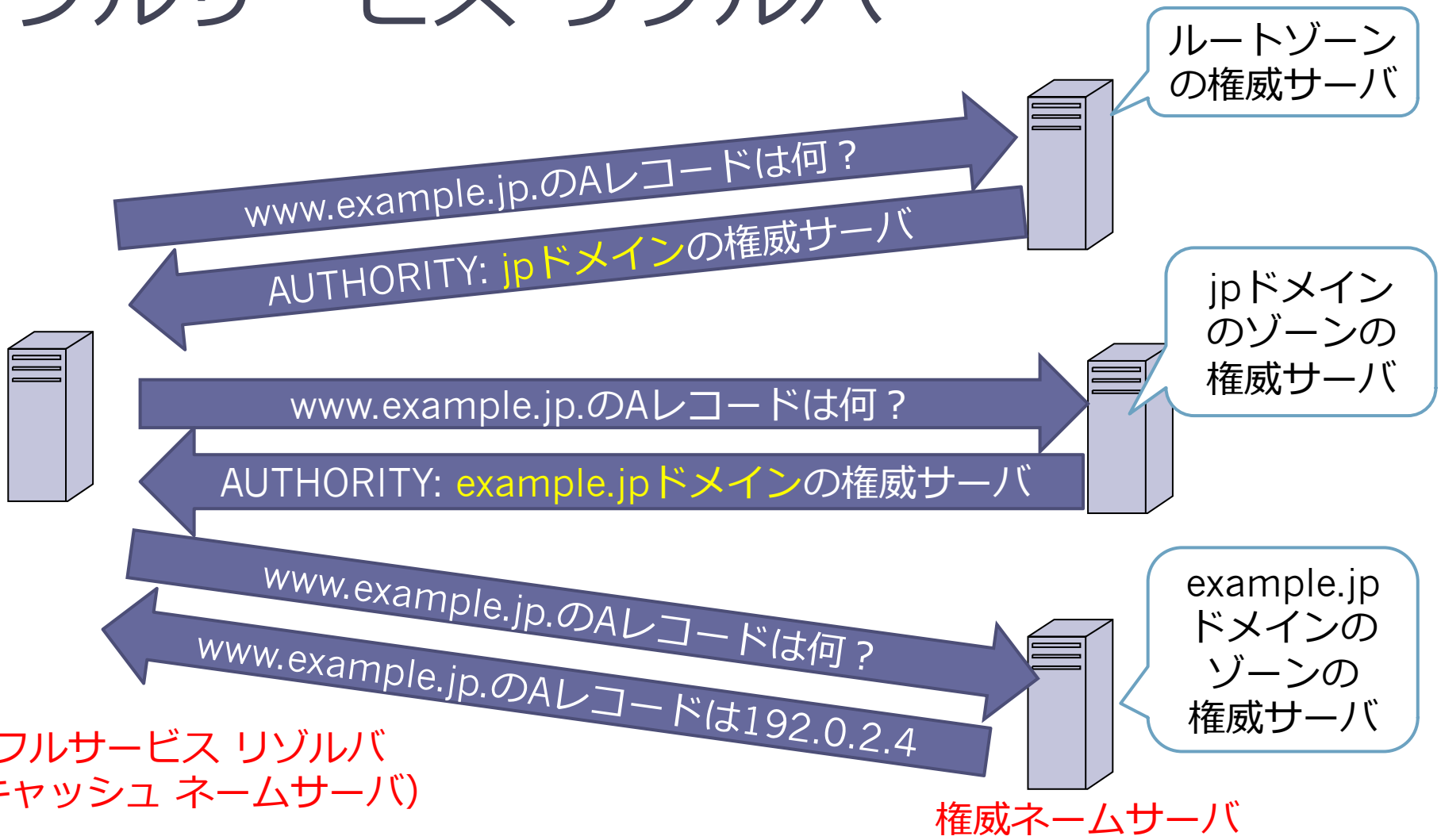
フル サービス リゾルバ

- 様々な呼び方がある
  - キャッシュ ネームサーバ
  - キャッシュDNSサーバ
  - DNSキャッシュサーバ
  - キャッシュサーバ

フルサービス リゾルバ


権威ネームサーバは
自身のゾーンに関する
回答をそれぞれ行う
スタブ リゾルバ
からリクエストを
受け取る
フルサービス リゾルバは
権威ネームサーバに
対して反復的に
問い合わせを行う
ルートゾーン
の権威サーバ
jpドメイン
のゾーンの
権威サーバ
example.jp
ドメインの
ゾーンの
権威サーバ
www.example.jpの
IPアドレスを教えて？
www.example.jpの
IPアドレスは192.0.2.4
スタブ リゾルバ
フルサービス リゾルバ
（キャッシュ ネームサーバ）
権威ネームサーバ

フルサービス リゾルバ


ルートゾーンの権威サーバ
www.example.jp.のAレコードは何？
AUTHORITY: jpドメインの権威サーバ
jpドメインのゾーンの権威サーバ
www.example.jp.のAレコードは何？
AUTHORITY: example.jpドメインの権威サーバ
example.jpドメインのゾーンの権威サーバ
www.example.jp.のAレコードは何？
www.example.jp.のAレコードは192.0.2.4
フルサービス リゾルバ
（キャッシュ ネームサーバ）
権威ネームサーバ

権威ネームサーバ

- 自身が権威を持っているゾーンの情報（リソース レコード）を提供する機能
- 権威を持っていない情報に関しては情報を提供しない
  - 例外はグルー
- 様々な呼び方がある
  - 権威DNSサーバ
  - DNS権威サーバ
  - コンテンツサーバ
  - 権威サーバ

このセクションのまとめ

- スタブ リゾルバ
- フルサービス リゾルバ
- 権威ネームサーバ

おわり