学生実験
3日目
DNS

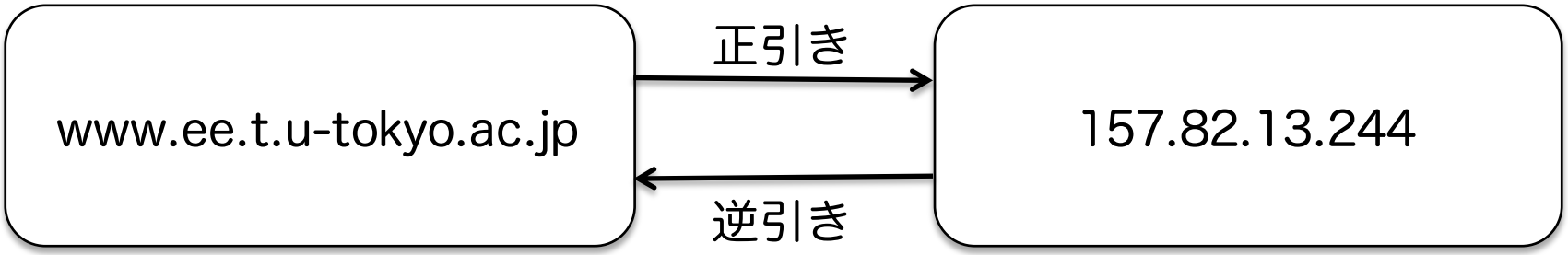
IPネットワークアーキテクチャ

江崎研究室

DNS

Domain Name System

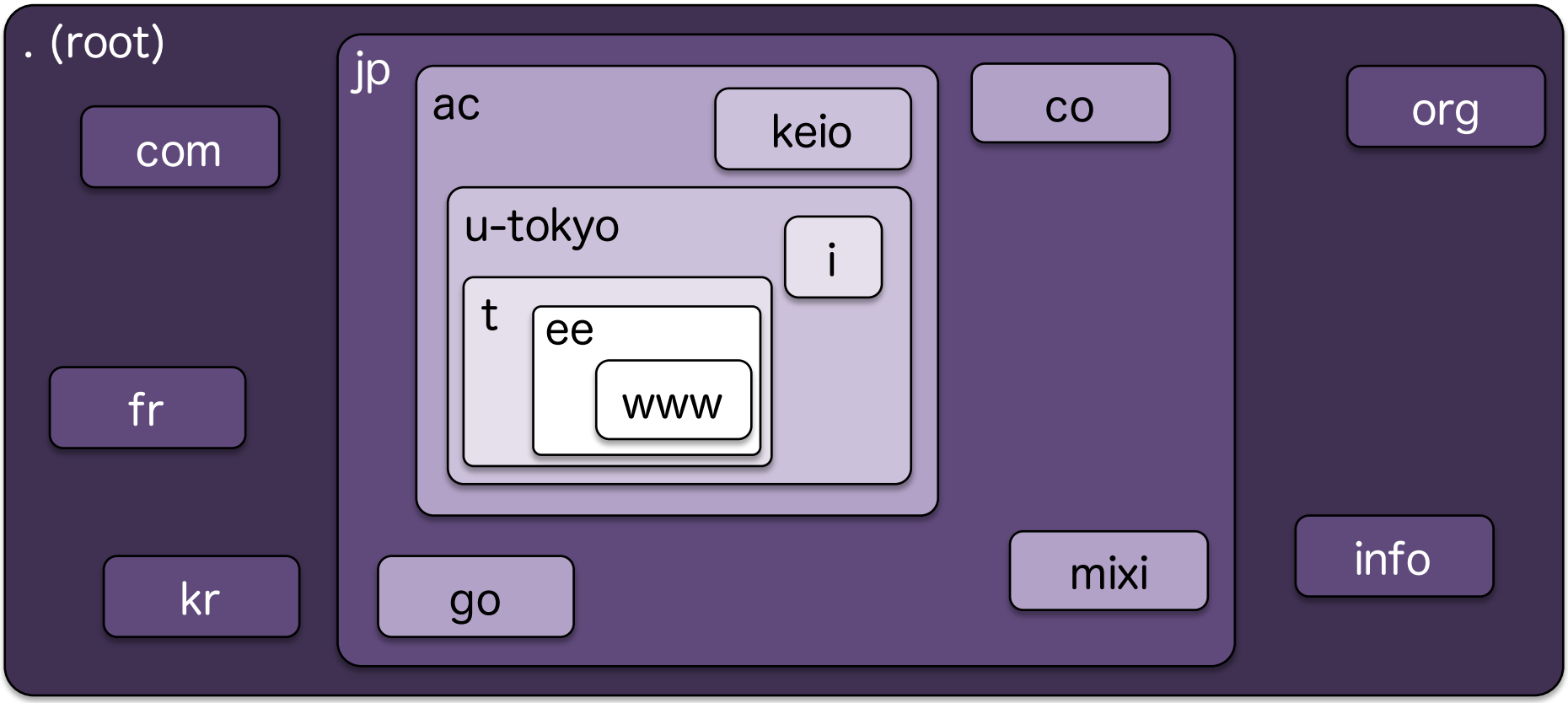
インターネット上の名前解決を実現



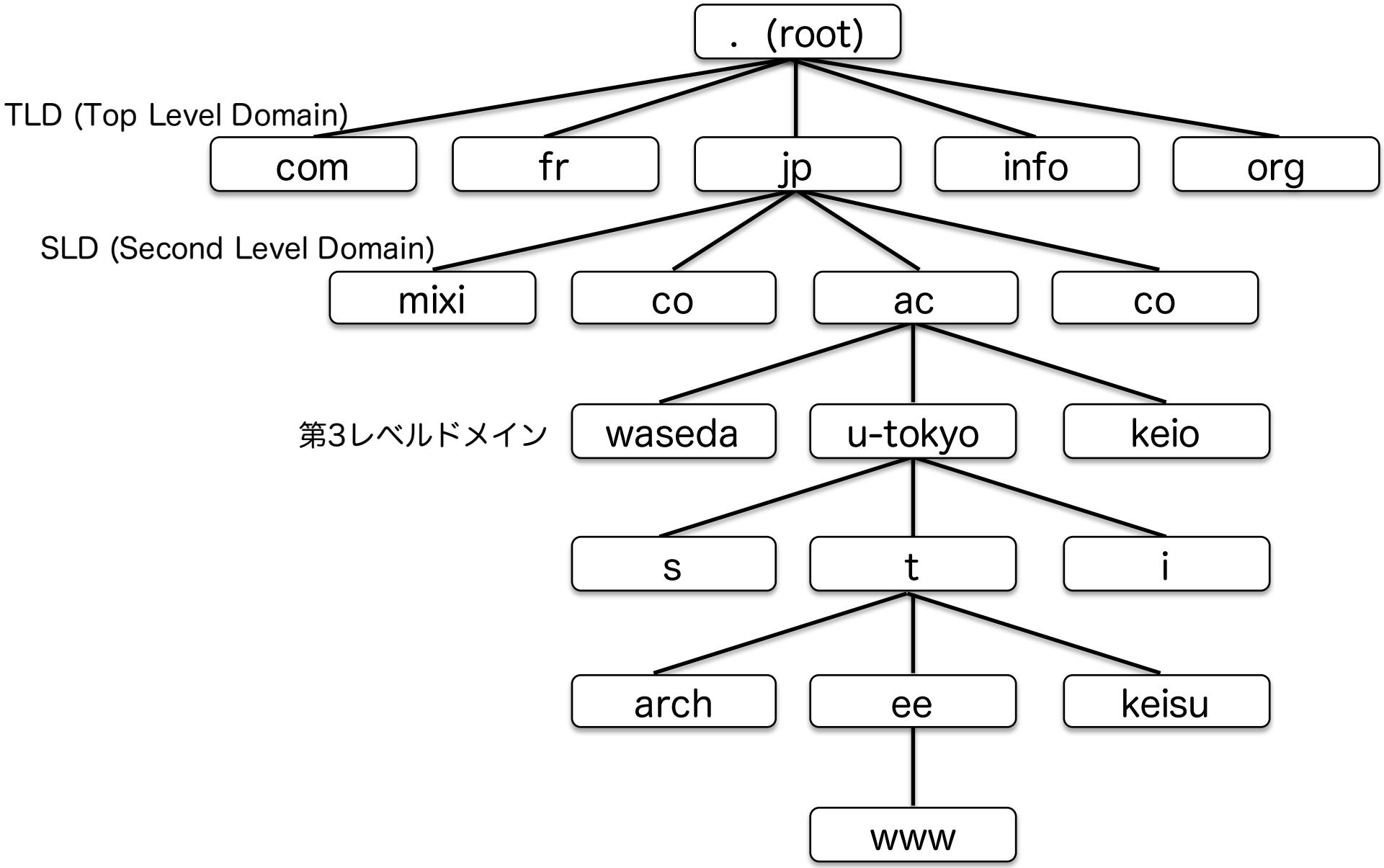
名前空間

インターネットで唯一
ドメイン＝名前空間内の範囲

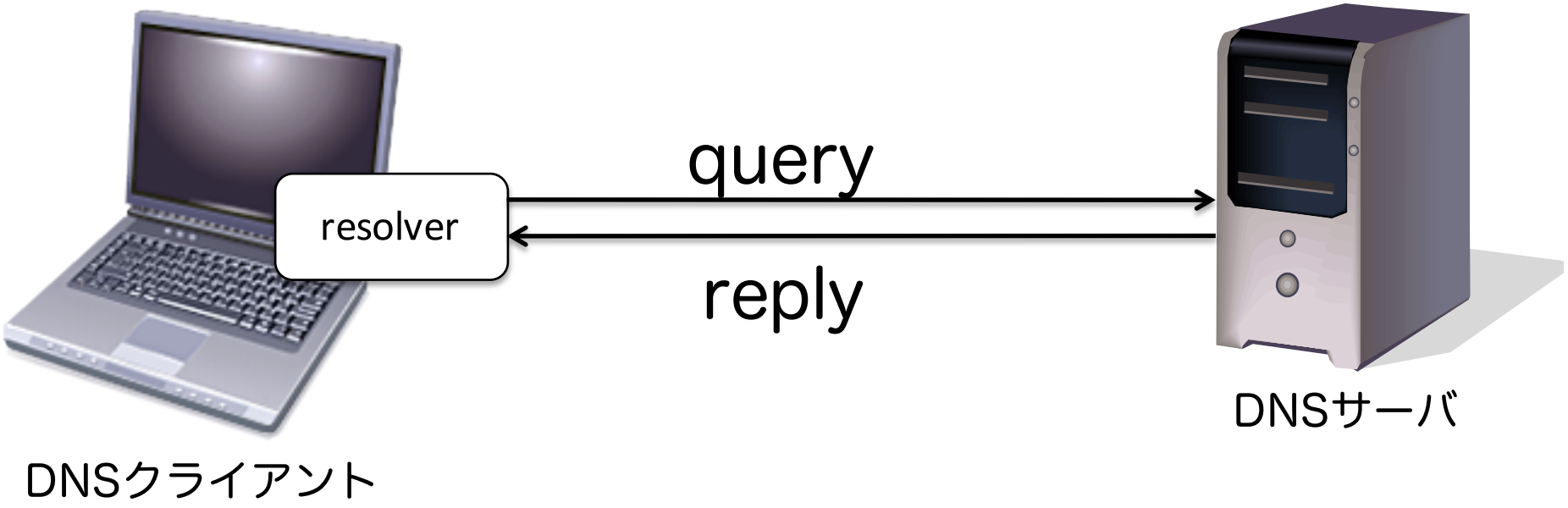
www.ee.t.u-tokyo.ac.jp の場合



ドメインの階層構造


. (root)
TLD (Top Level Domain)
com
fr
jp
info
org
SLD (Second Level Domain)
mixi
co
ac
co
第3レベルドメイン
waseda
u-tokyo
keio
s
t
i
arch
ee
keisu
www

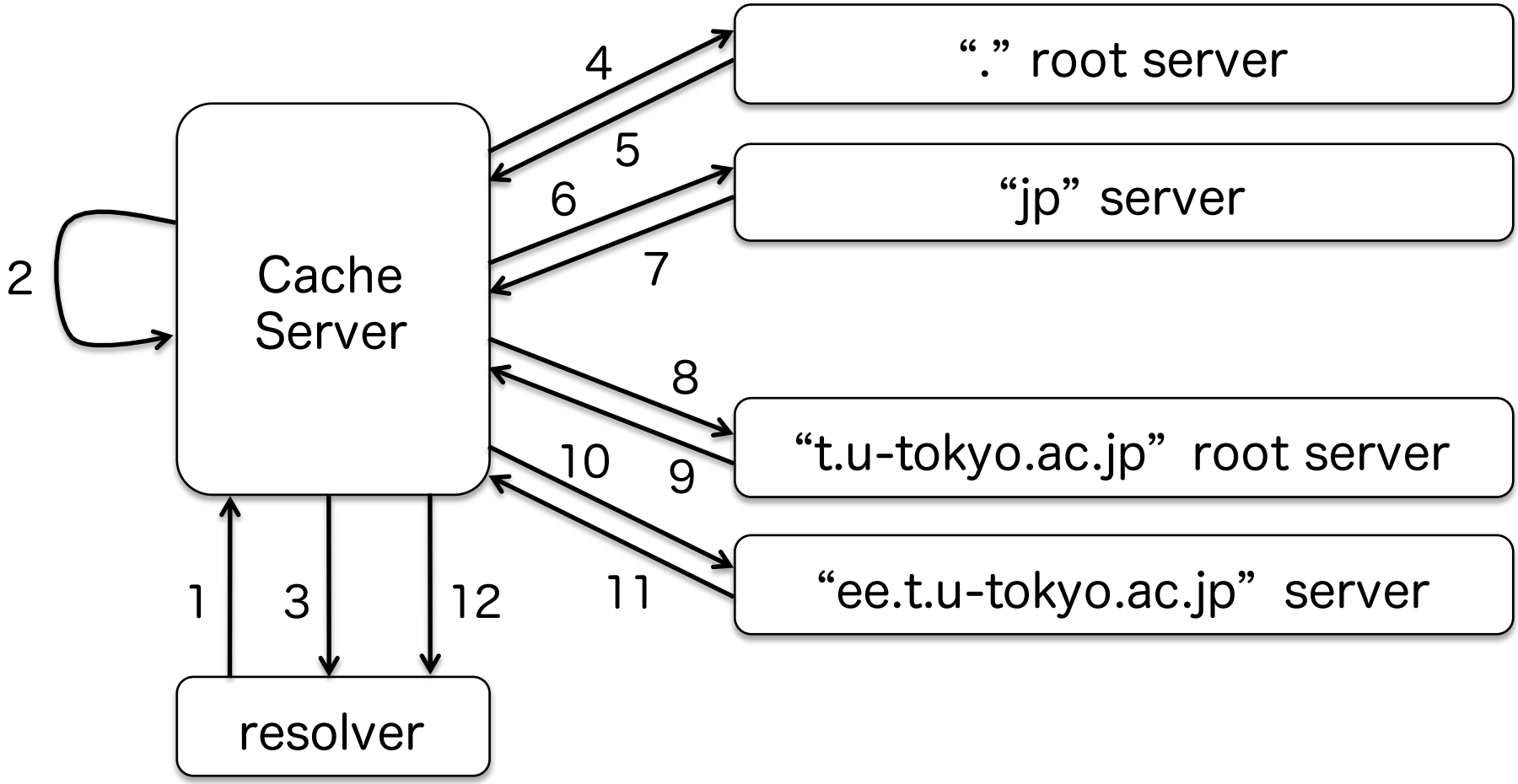
DNSの動作原理



課題(1) 動作確認

インターネットに接続されたホストにて、
URLのホスト部の最も右に “.” を付けて
ウェブブラウズした時に、ブラウザの挙動
が異なるかどうか確認しなさい。また、同
様に、pingやその他のアプリケーションを
用いた場合の動作も確認しなさい。

名前空間における検索



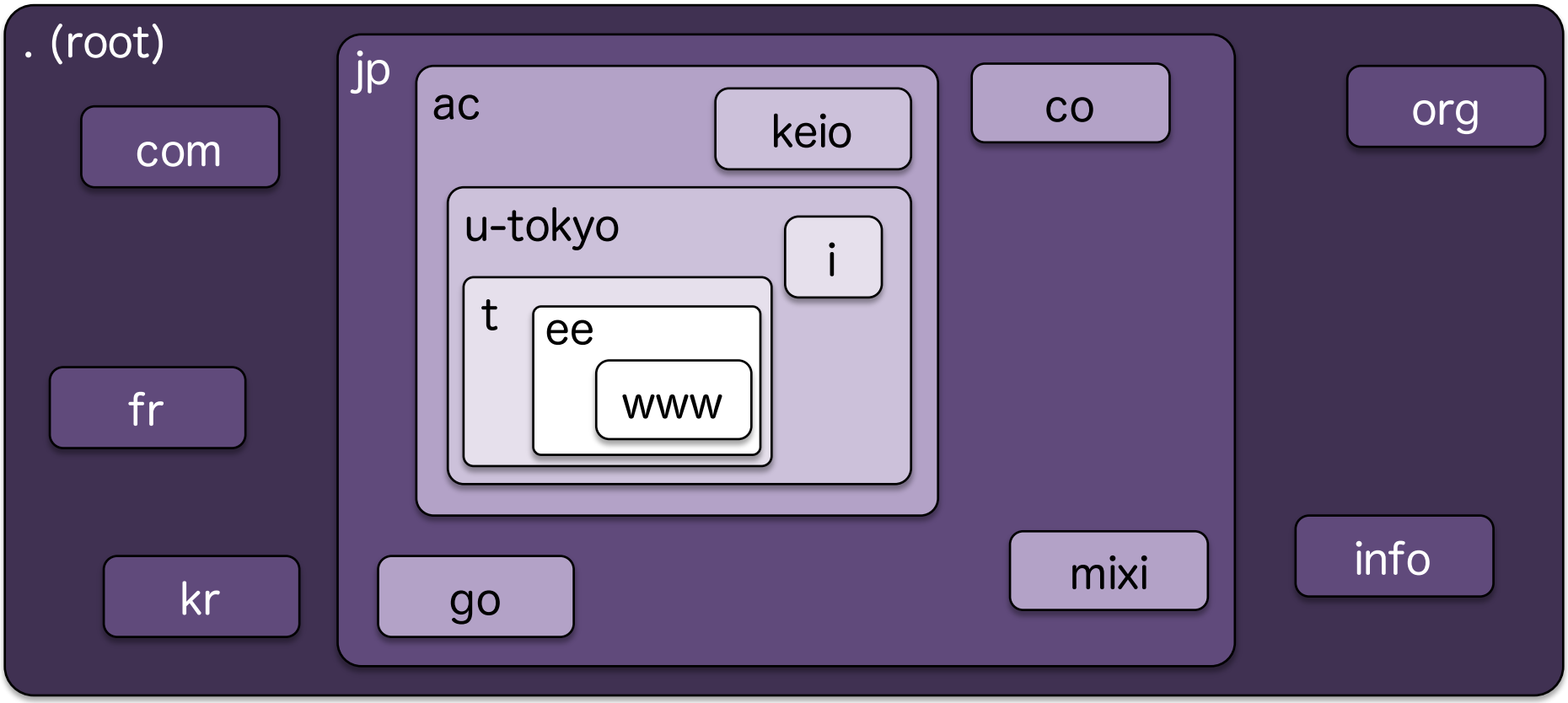
詳細は教科書を参照すること

課題(2) リゾルバの確認

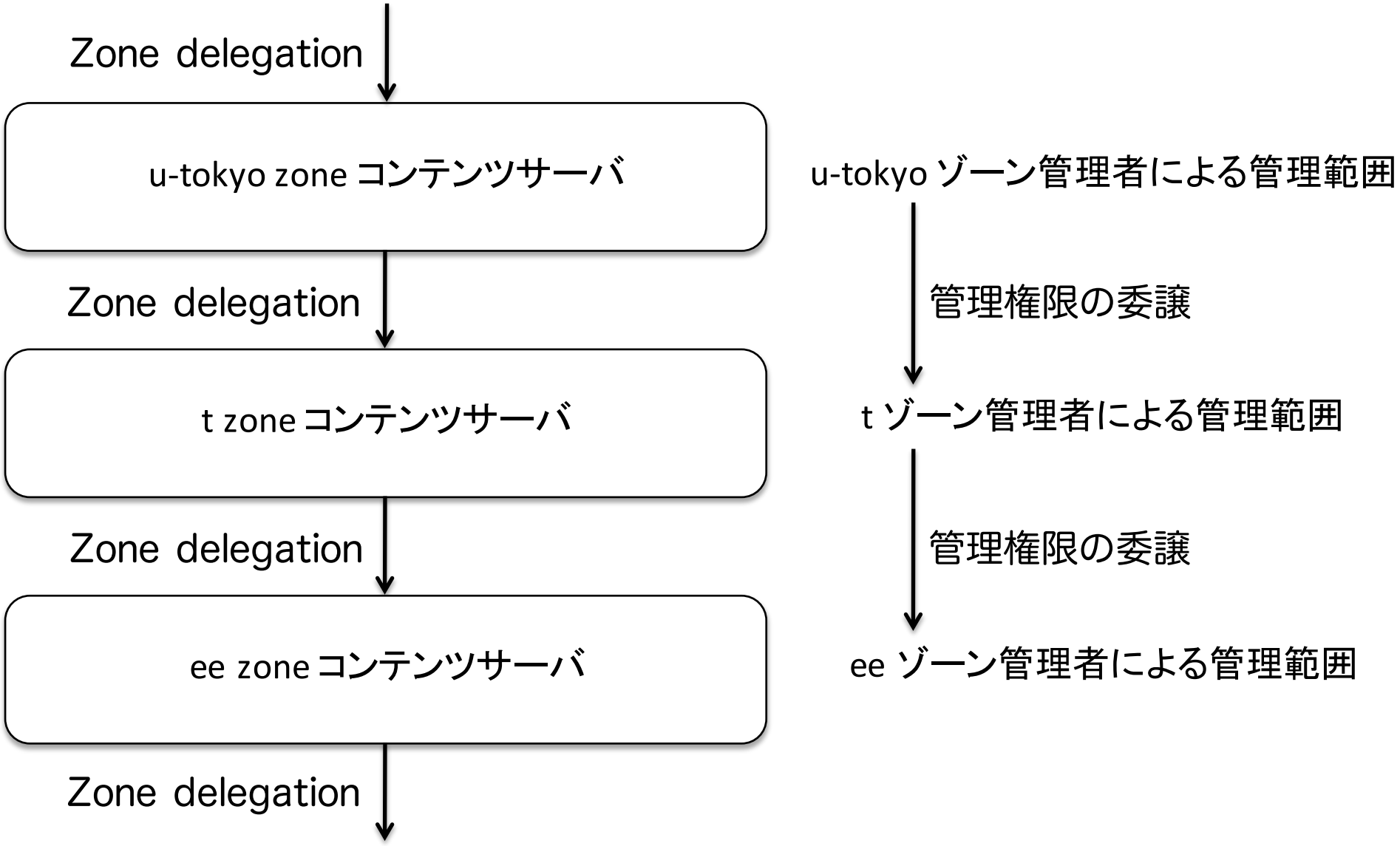
リゾルバ上のキャッシュサーバのIPアドレスは、UNIXシステムでは “/etc/resolv.conf” に登録されている。このファイルを確認すること。また、このファイルを編集し、誤ったIPアドレスが指定された場合、インターネットの利用にどのような影響が発生するか予想し、実際に確認しなさい。

DNSは分散データベース

それぞれのドメインを個別のDNSサーバが管理
ドメインごとにDNSサーバが必要になる

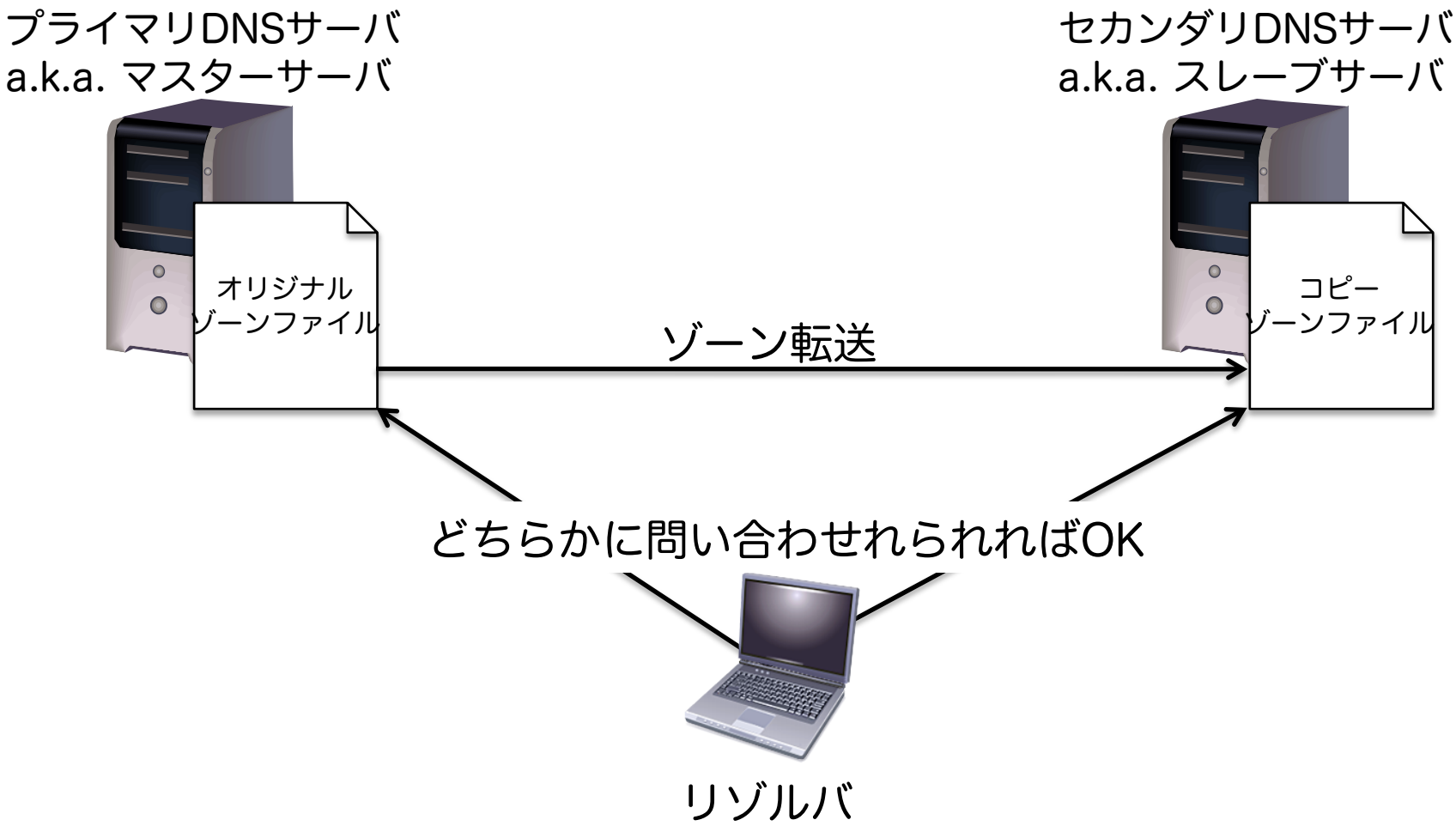


管理権限の委譲（delegation）



プライマリ＆セカンダリ

DNSサーバが一台だと心許ない



課題(3) digコマンドの利用

リゾルバとして稼働するホスト上で、名前解決専用アプリケーション(dig)を利用して、DNSの検索クエリを発行できる。digコマンドを利用してリゾルバとキャッシュサーバ間の通信を確認しなさい。同時にwiresharkを実行し、どのようなパケットが流れているか確認しなさい。

課題(4) 名前解決の流れ確認

リゾルバとして稼働するUNIXシステムで、一時的にキャッシュサーバの機能を扱えるdnstracerを利用し検索状態を把握し、再帰的な検索が行われていることを確認しなさい。

課題(5) DNSサーバの実行

リゾルバ、キャッシュサーバ、コンテンツサーバは、実態ではなく機能なので、それぞれの機能を同一のホストで稼働させられる。

1) 各自のPCにDNSサーバ(bind)をインストールし、リゾルバ キャッシュサーバが同時に稼働する構成にしなさい。

2) リゾルバより、検索クエリをキャッシュサーバに投げかけ、キャッシュサーバが再帰的に検索することを確認しなさい。また、キャッシュサーバが検索結果をキャッシュすることを確認しなさい

キャッシュサーバの設定(1)

Bind9のインストール：
```
# apt-get install bind9
```

キャッシュサーバの設定：
/etc/bind/named.conf.optionsを編集
再帰検索を許可するよう設定する

編集例は実験webページの
TIPSを参照

設定ファイルのチェック：
```
$ named-checkconf /etc/bind/named.conf
```

Bind9の再起動：
```
# /etc/init.d/bind9 restart
```

次のスライドに続く

キャッシュサーバの設定(2)

キャッシュサーバの動作確認：
`$ dig @127.0.0.1 (適当なアドレス)`

キャッシュ状況の確認：
同じアドレスを複数回digした時に流れるパケットの様子を
tcpdumpやwiresharkで確認する

注) そのために存在するが滅多にアクセスしない名前を検索すること

課題(6) 規模性

DNSは階層構造を利用することで規模性を確保できる。これをどのように実現しているか考察しなさい。

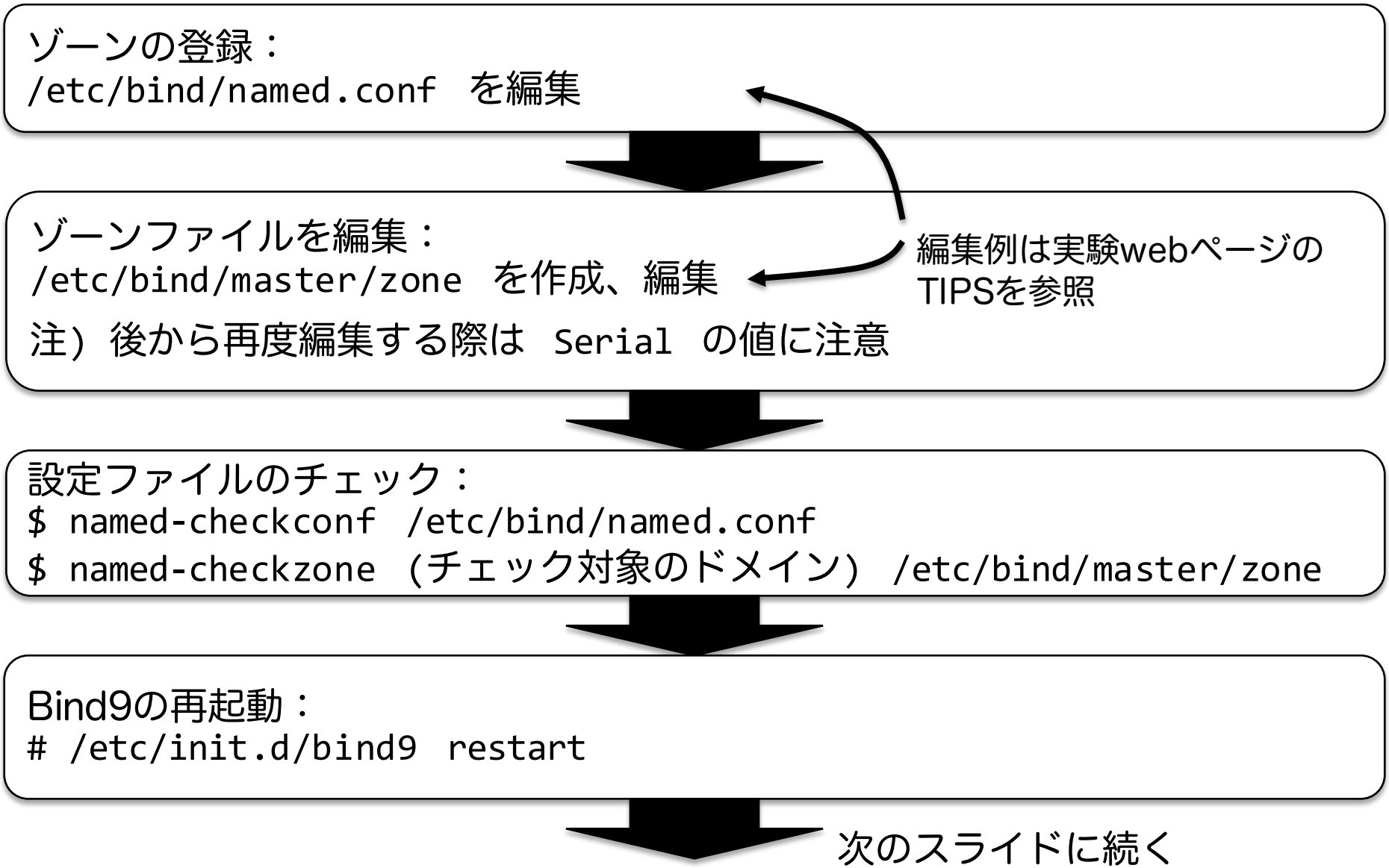
課題(7) コンテンツサーバの設定

各自のPCでコンテンツサーバと接続確認用のweb
　サーバを設定しなさい。動作確認ではdigを利用す
　ること。

webブラウザにてFQDNを指定して、
　自身のPC内のwebサーバへの接続を確認しなさい。

各チーム内でサブドメインを委譲する設定を行い、
　webブラウザにてFQDNを指定して、
　それぞれのwebサーバへの接続を確認しなさい。

コンテンツサーバの設定(1)



コンテンツサーバの設定(2)

Apache2のインストール：
apt-get install apache2

コンテンツの準備：
/var/www/index.html を編集
ブラウザで http://localhost/index.html を指定して表示できるか確認

コンテンツサーバの動作確認：
自分または他のPCのブラウザから
ゾーンファイルで指定したドメインでコンテンツが表示できるか確認

SOAレコード

- Start of Authority
  - ゾーンにおけるプライマリサーバの指定(FQDNで)
  - ゾーン管理者のメールアドレス（”@”➔”.”）
  - SERIAL…データベースのバージョン番号
  - REFRESH…セカンダリの更新確認間隔
  - RETRY…REFRESHに失敗したときの再試行周期
  - EXPIRE…セカンダリのデータ保持期間
  - MINIMUM…リソースレコードのデフォルトTTL

課題(8) レコードタイプの調査

教科書に載っているAレコード、AAAAレコード、NSレコード、SOAレコードの他にも様々なレコードタイプがある。他にどのようなレコードタイプがあるか調べなさい。また、それらのレコードタイプを各自のコンテンツサーバに設定し、利用してみなさい。

リソースレコード(RR)とレコードタイプ (抜粋)

レコードタイプ	意味
A	IPv4アドレス
AAAA	IPv6アドレス
NS	ドメインに対するDNSサーバの指定
SOA	ゾーン情報に対するパラメータの設定
CNAME	エイリアス
MX	ドメインに対するメールサーバの指定
PTR	逆引き用のホスト名

課題(9) 逆引きの調査

IPアドレスからFQDNを調べることを逆引きという。逆引きにはPTRレコードを利用するが、逆引きがどのように動作するか調べなさい。