2-3　データ収集

東京大学 数理・情報教育研究センター

2021年4月19日

概要

- Webサイトやエッジデバイスからのデータ収集方法を学びます

本教材の目次

通信技術・通信プロトコル

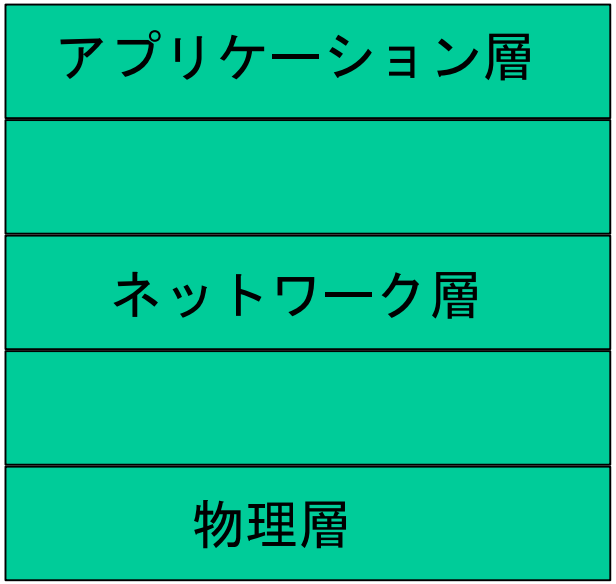
データの収集には、通信ネットワーク、通信技術が必須です。

そして、情報を収受するには多くのルールが必要です。
例えば、AさんがBさんに情報を伝える場合、
- 英語を用いるのか、日本語なのか
- 手紙を渡すのか、伝言するのか
- 直接会うのか、誰に仲介を頼むか
- 仲介者はどのように決めるのか
- 情報を受けとったことを確認する方法はどうするのか

などを決める必要があります。通信に関わるこうしたルールを通信プロトコルと言います。

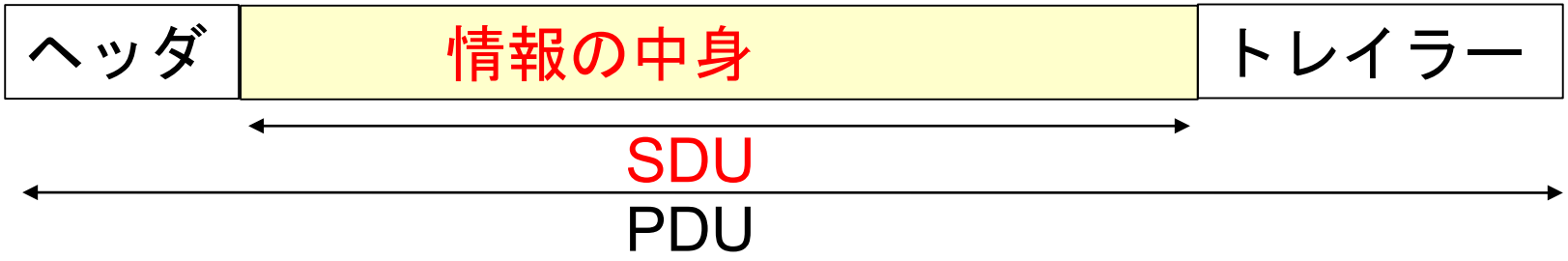
通信プロトコルレイヤ

通信プロトコルは、通常、レイヤ（層）と呼ばれる階層に分かれており、階層ごとに用いるプロトコルを決めることで通信が成立します。また、階層化することで各階層のプロトコルを適切に組み合わせることが可能になります。また、開発もプロトコルごとに別々に開発可能。最上位層をアプリケーション層、最下位層を物理層と呼びます。アプリケーション層と物理層の間にあり、どこを経由して相手先まで情報を届けるか（経路選択あるいはルーチングなどと呼ばれる）を、主に行っている層をネットワーク層と呼びます。



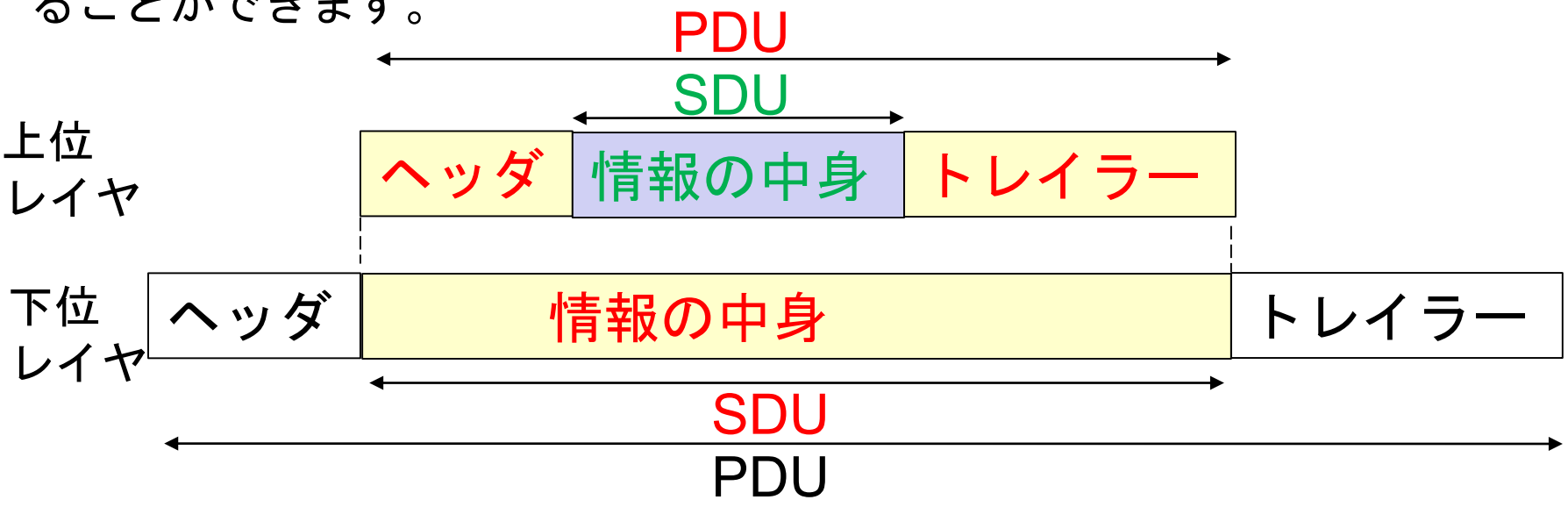
ヘッダ

各レイヤの情報は、先頭にヘッダと最後にトレイラーと呼ばれる付加情報に挟まれて送られます。（トレイラーがないプロトコルも多いです。）ヘッダには、あて先アドレスなどの情報が格納されます。あて先アドレスは仲介者（中継ノード）のアドレスの場合や最終目的地の場合など、プロトコル、レイヤで変わります。アドレス体系もプロトコルで変わります。電話番号と郵便番号のようなものです。全体として、封筒とその中に入れた手紙のような構造になります。



PDU

ヘッダ、トレイラーを含めた全体をプロトコルデータユニット（PDU)、情報の中身の部分をサービスデータユニット（あるいはペイロード）と呼びます。１つ上位のレイヤのPDUは、その下のレイヤのSDU（情報の中身）になるという形の入れ子構造になります。情報を送る側は、最上位のアプリケーション層から順にPDUを構成し下位層に渡します。PDUを受け取った最下位層の物理層において無線や光などの物理的な手段で実際の通信を実行します。この通信を受信した相手先は、今度は、逆に最下位層から順に情報の中身を取り出して１つずつ上位層に渡していくことでアプリケーション層で、送信側のアプリケーション層が作った情報の中身を得ることができます。



インターネット

広域にまたがるコンピュータのネットワークには、多くの場合、インターネットと呼ばれるネットワークが使われています。インターネットのネットワーク層のプロトコルとしては、インターネット・プロトコル(IP)が用いられます。
IPで用いられるPDUはIPパケットと呼ばれ、そのヘッダ（IPヘッダ）にはIPアドレスと呼ばれるあて先アドレスが含まれます。
IPアドレスには32ビットで表現するIPv4（バージョン４）アドレスと128ビットで表現するIPv6（バージョン６）アドレスがあります。IPv4アドレスは、ほぼ使い尽くされているので、今後は、IPv6アドレスの使用が盛んになると思われます。

インターネットのネットワーク層はIPですが、それ以外の階層には多様なプロトコルを用いることができます。従って、インターネットを構成する物理的なネットワーク（物理層）として光や無線が混在することが可能になります。また、アプリケーション層としては後述するHTTPやFTPなどの多様なプロトコルを用いることができます。

IOTとは

もともとコンピュータ間の通信に用いられるインターネットですが、それを物との通信にも使おうという試みがあります。それが物のインターネットであるIoT(Internet of Things)です。
多くの場合、物というのはセンサー（温度計や監視カメラなど、情報を取得する装置、測定器）やアクチュエータ（ロボットアームを動かすモーターや温度の調整を行う空調機など、実効行為を行う装置）のことです。そうした物に小さな通信装置をつけてインターネットに接続することで世界中のどこからでも測定器の測定結果を見たり、離れたところにいるロボットを動かしたりできるということがIoTで想定されています。

IOTの特徴

インターネットは、もともと、コンピュータ同士の通信に使うことが想定されているので、それを物（物についた通信装置）との通信でうまく使うことができるのでしょうか。また、これまでのコンピュータ同士の通信とどこが大きく変わるのでしょうか。IoTの特徴として言われていることは、以下のようなことです。

１）数が非常に多い
既に、2019年で84億個の物と接続しているという報告があります[1]。今後ますます物との接続数が増加することが想定されます。コンピュータの数より物の数の方が圧倒的に多いので、それだけ多くの物を接続することが可能なのでしょうか。

[1] https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016
Access: 2021/3/2

IOTの特徴

２）通信機の能力が低い
インターネットに接続されるコンピュータの多くは、高性能なCPUと多くのメモリを有する装置です。ところが物につけられる通信装置の多くは、安価で小型な分、プロセッサの能力も低く、メモリも少ないです。通信速度も低いことも多いでしょう。このような通信装置で多様なアプリケーションを効率的に実現しなければなりません。

３）低消費電力化が必要な場合が多い
センサーはいろいろな場所に設置されます。電池駆動が必要な場合も、数多く想定されます。人が行きにくい場所に設置されたセンサーについてはもちろん、電池交換が容易でも非常に数が多い場合の面倒を考えると、消費電力を低く抑え、電池寿命を長持ちさせる必要があります。

センサーネットワークとは

IoTでは、センサーに通信装置を付けて通信をすることで様々な状況を測定、監視できることが期待されています。IoTでは、センサーにつけた通信装置がIPアドレスを持ち、IPを用いて通信することになります(*)。しかしながら、これ以外の方法で通信装置付きセンサーと通信し、様々な状況の測定、監視は可能です。一般的に、こうしたセンサーとの通信を行うネットワークをセンサーネットワークと呼びます。

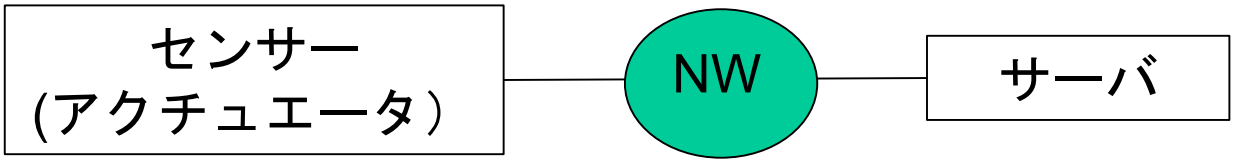
IoTも一種のセンサーネットワークとして捉えることができるので、IoTの特徴、「膨大な数」「低能力通信ノード」「低消費電力要求」は、そのままセンサーネットワークの特徴となります。そしてこれらの特徴は、そのまま、センサーネットワークが抱える課題となります。

センサーが計測したデータ（センサーデータ）は、センサーネットワークにより収集されます。

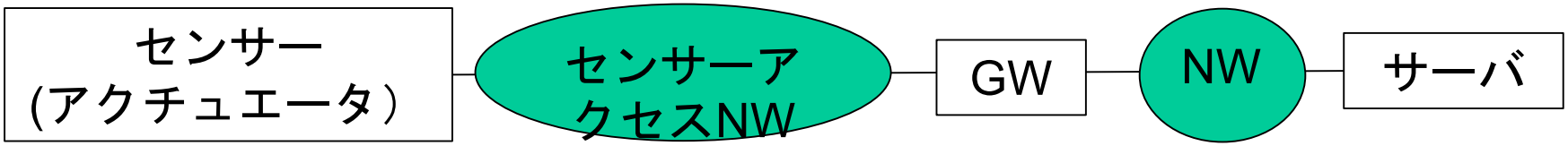
(*)そうでない場合も、IoTと呼んでいるケースもあるようです。

センサーネットワークアーキテクチャ

センサーネットワークの構成は、ゲートウェイ装置(GW)の介在あり、無し、で大別されます。以下の図で、ネットワーク(NW)の典型例はインターネット、あるいは、携帯電話網です。



(a) GWが無い場合

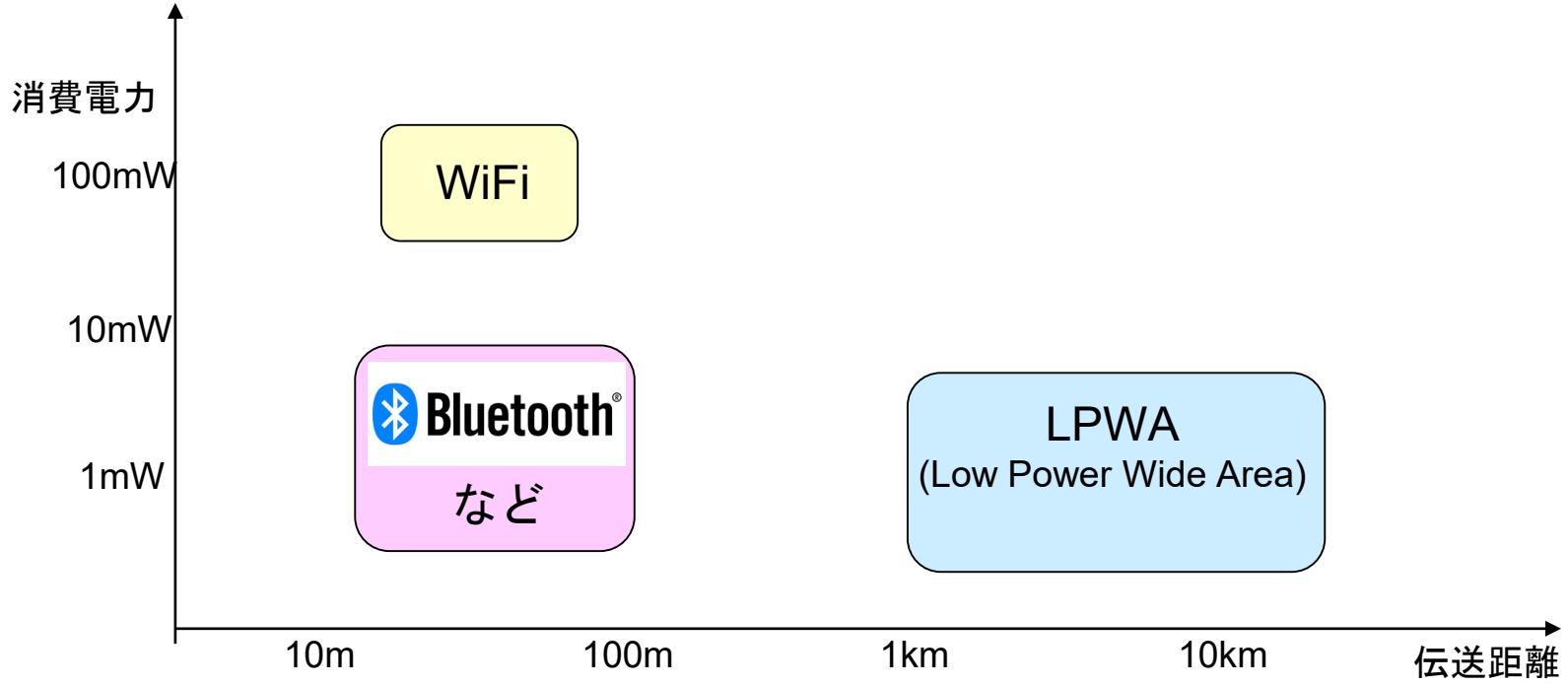


(b) GWがある場合

(a)の典型例がセンサーについた通信機がIPアドレスを使う場合です。
(b)の場合の「GW-NW-サーバ」部分は、通常のコンピュータ間接続などと同様ですので、次ページ以降では、センサーアクセスNW部分を中心に見てみましょう。

センサーアクセスネットワーク用無線通信規格

センサーアクセスネットワークには、イーサネットなどの有線ネットワークも使われますが、多くの場合、無線のネットワークが使用され、その用途に応じて、多くの規格があります。その特徴を消費電力と通信距離で分類したものが以下の図になります。WiFiやBluetoothは皆さんも聞いたことがあるでしょう。それに加えて、最近では、LPWAと呼ばれる通信規格に属するものが提案され、利用され始めています。



横谷哲也、IoTと通信ネットワーク技術、電子情報通信学会誌、102、5、pp. 383-387、2019の図４をもとに作成

センサーアクセスネットワーク用無線通信規格

LPWAグループの通信規格は低消費電力で広い範囲(数km)をカバーすることを目的とします。通常、速度は遅いため、大量の情報を一度に送出するのではなく、少しずつ出すような通信に向いています。低消費電力であるため、電池で動作し、電池交換なしで長く使われるような状況に適しています。2012年ごろからLoRa、SIGFOXなどの規格が商用利用されるようになりました。また、 携帯電話網のLTE(Long Time Evolution)をセンサーネットワークに用いるためにNB-IoT (Narrow Band-IoT)が開発されました。LTEは高速広帯域ですが上記のような用途には消費電力が大きすぎ高価格です。NB-IoTにより低速狭帯域ですが低消費電力、低価格を実現しています。NB-IoTの利用にはライセンスが必要ですので、携帯電話会社から提供される通信装置を使う必要がありますが、干渉などが管理されています。一方、LoRa、SIGFOXはアンライセンスですのでWiFiのように自由に設置可能ですが、干渉などによる性能低下の懸念は生じます。

センサーアクセスネットワーク用無線通信規格
近距離無線

Bluetoothは非常に良く使われています。2019年には42億台が出荷されました[1]。主には、スマホやパソコンとイヤホンやマウスなどの周辺機器の接続が典型的な使用例です。ペアリングと呼ばれる動作によって２つの装置間が結ばれます。片方がマスター、もう一方がスレーブとなり、スレーブはマスターの指示により動作するだけで、スレーブ同士の直接通信はできません[2]。パソコンやスマホが通常マスターになります。それらの機器はGWになり得ますので、これでマウスなどのセンサーとGW間のセンサーアクセスネットワークができたことになります。Bluetoothの新しい規格では中継機能が追加され、センサー同士でネットワークが構成可能になっています。

[1] https://www.bluetooth.com/wp-content/uploads/2020/03/BMU_2020-JPN.pdf
Access: 2021/3/2
[2]アンドリュー・S・タネンバウム／デイビッド・S・ウエザロール、コンピュータネットワーク、日経BP社、2013

エッジデバイス、エッジコンピューティング

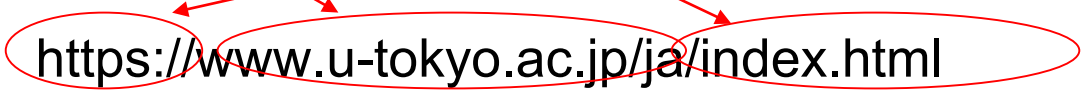
クラウドコンピューティングが遠隔地にあるクラウド(2-1参照）上のコンピュータを利用して計算を実行するのに対して、センサーなどデータを発生させるデバイス（エッジデバイス）に比較的近い位置にあるコンピュータで計算を行うことをエッジコンピューティングと言います。このコンピュータとエッジデバイスが一体化したケースもあります。エッジコンピューティングはクラウドコンピューティングに比し通信遅延を抑制することができるため、プラントの制御や車両の自動運転などの遅延に対する要求が厳しい処理に適しています。

アプリケーションプロトコル（HTTP)

HTTP (Hyper Text Transfer Protocol)はWebサーバーとクライアント（サーバと接続する相手）がWebコンテンツをやりとりするためのプロトコルです。Webコンテンツの記述言語であるHTML (Hyper Text Markup Language)で書かれたテキストや画像をメタデータを含めてやりとりすることができます。
クライアントのHTTPリクエストに対してサーバがHTTPレスポンスを返すことが基本です。HTTPリクエストは、画面上の特定の場所をクリックしたり、URL(後述)を指定したりすることで送出される場合が多く、HTTPメソッド、対象までのパス、などからなります。
HTTPメソッドには、「やりたい動作」などの情報が含まれます。最も良く使われる「やりたい動作」はGETで、これは「パスの指定先から取ってくる」ことを意味します。HTTPレスポンスは、リクエストが成功したか否かなどを表すstatus codeと呼ばれる回答を含むヘッダと本体（コンテンツ取得時は、取得したコンテンツ）が主たる内容になります。
HTTPでは、通信内容が暗号化されていないため、セキュリティ上の懸念があることから、最近では、暗号化を行うHTTPであるHTTPSの利用が増えてきています。

URL、DNS

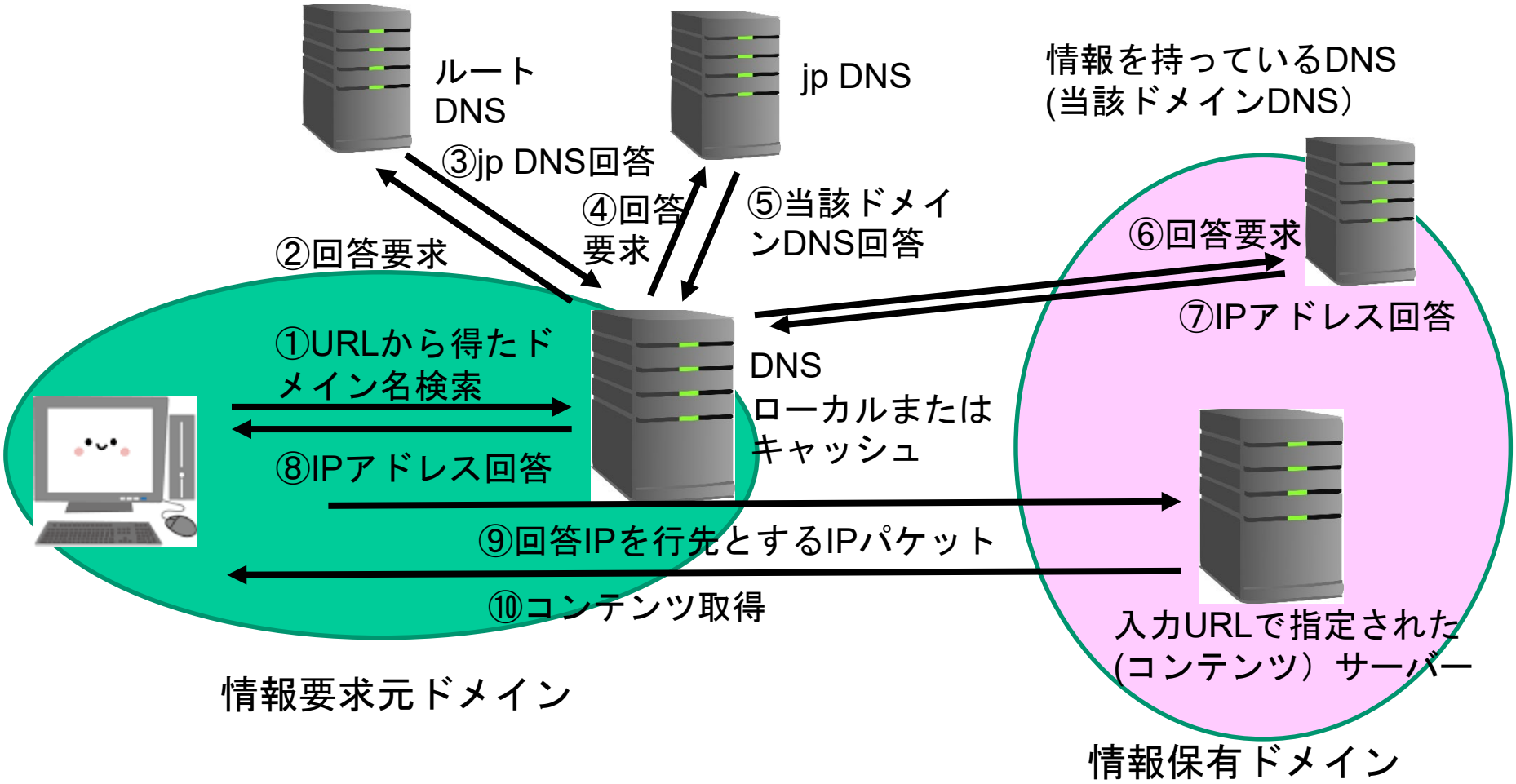
HTTPをはじめアプリケーションプロトコルの中には、コンテンツなどの位置を一種のアドレスであるURL(universal resource locator)で指定するものがあります。URLの構造はプロトコル、コンテンツのあるコンピュータのドメイン名、パス名からなります。



一方、指定された位置に到達することはアプリケーション層ではなくネットワーク層の機能です。しかし、インターネットの場合のネットワーク層のプロトコルであるIPはURLを解釈できないため、URL（から得たドメイン名）をIPアドレスに変換する必要があります。その仕組みはDNS(domain name system)と呼ばれます。DNSは階層化され、上位DNSは下位DNSを知っており、該当ドメインDNSが所属ホストの情報を持っているという仕組みになっています。そのため、世界中のドメインとIPアドレスを対応づけることができます。

アドレス変換から見たコンテンツ取得までの流れ

jp傘下のドメインの場合



アプリケーションプロトコル（FTP)

FTP (File Transfer Protocol)は、クライアント（サーバを使う人）とサーバの間でファイルをやりとりするためのプロトコルです。FTPは制御用のコネクションと実際にデータ（ファイル）を転送するためのコネクションを別に用意します。２つの見えない線をクライアントとサーバ間に用意すると思ってください。制御用コネクションを通じて、データ転送用のポート番号と呼ばれる通信のための番号をやりとりします。決定されたポート番号を用いて、データをデータ転送用コネクションで転送します。

FTPは、認証のためのパスワードなどの情報やデータ自体が暗号化されずに送られるため、セキュリティ上の懸念があります。その場合は、次ページで勉強するSSHなどの仕組みを利用するなど、FTPを改良したプロトコルが使われ始めています。

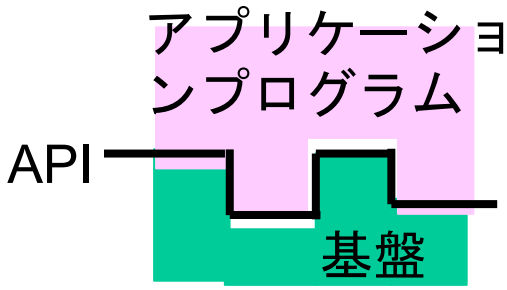
アプリケーションプロトコル（SSH)

SSH (Secure Shell)は、サーバなどネットワークに接続している機器を離れたところにいるサーバ管理者などが安全に操作するためのプロトコルです。悪意をもった人がサーバにloginしてしまうと非常に影響が大きいので、それを防ぐため、通信はすべて暗号化された状態で行われます。また、認証も、パスワード認証もありますが、より堅牢な公開鍵認証（2-6参照）が推奨されています。遠隔にあるコンピュータを操作するアプリケーションプロトコルとしては、これまでtelnetが多く使用されてきましたが、SSHの利用により安全性が大幅に向上しています。
認証後、サーバ管理者が直接使っているPCから、遠隔にあるサーバに、sshコマンドを用いて接続して、loginして、サーバでの操作を行うという使い方が一般的です。

クライアント技術（API)

API (Application Programming Interface)は、あるソフトウエア基盤上に第３者がアプリケーションプログラムを構築するための切り口のことです。この切り口に合わせてプログラムを作れば、この基盤が所定の動作をします。この結果、多くの第三者が(APIに準拠する限り）基盤の中身の詳細を知らなくとも自由にアプリケーションを開発できますし、基盤提供者は多くのアプリケーションに使ってもらうことができることになります。実際のAPIはアプリケーションプログラムを書くためのルール集です。コマンド集あるいはファイル群で提供される場合もあります。

APIを提供する基盤には多くのものがあります。AmazonやGoogleなど多くのプラットフォーム事業者は自社のWebサービスを使ってもらうためにAPIを提供しています。



クライアント技術（SDK)

SDK (Software Development Kit)は特定の基盤上のソフトウエア開発のためのツール類（APIに関するものを含む、マニュアルやサンプルコード、開発支援用プログラムなど）を言います。通常、Micrsoftなどのコンピュータオペレーティングシステム事業者、プログラミング言語メーカ、ハードウェア基盤事業者によって提供されます。SDKを利用することで、当該基盤を利用したソフトウエアの開発が容易になり、提供者にとっては、その基盤のユーザが増えることになります。

オープンデータ

- ウェブには，行政・研究機関，企業，個人などの第三者が提供・発信するさまざまなデータがあります.
- 例えば，政府統計のポータルサイトである「e-Stat」では，国内の様々な統計データを検索して入手することができます.
  - e-Stat：https://www.e-stat.go.jp/
- 機械（コンピュータ）の読み取りに適したデータ形式で，二次利用が可能な利用ルールで公開されたデータをオープンデータと呼びます.
- 政府のカタログサイト「DATA.GO.JP」では，二次利用可能な様々なオープンデータを検索して入手することができます.
  - DATA.GO.JP: https://www.data.go.jp/

オープンデータ

e-Statのサイト

出典：政府統計の総合窓口(e-Stat)
（https://www.e-stat.go.jp/）

オープンデータ

- 教育用標準データセットは，データ分析のための汎用素材として，
e-Statの統計データを元に作成され公開されているデータセットです.
  - 教育用標準データセット： https://www.nstac.go.jp/SSDSE/

- データセットは表計算ソフト形式またはCSV形式で公開されており，
以下のデータが含まれています.
  - A. 市区町村別データ
    - 1741市区町村，125項目の統計データ
  - B. 都道府県別・時系列データ
    - 47都道府県，12年分，107項目の統計データ
  - C. 都道府県庁所在市別・家計消費データ
    - 都道府県庁所在市，227項目の統計データ

オープンデータ

教育用標準データセット A. 市町村別データ（表計算ソフト形式）

	A	B	C	D	E	F	G	H	I	J	K	L
1	code	prefecture	municipality	A1101	A110101	A110102	A1102	A110201	A110202	A1301	A130101	A130102
2	year	年度	年度	2015	2015	2015	2015	2015	2015	2015	2015	2015
3	地域コード	都道府県	市区町村	総人口	総人口（男）	総人口（女）	日本人人口	日本人人口（男）	日本人人口（女）	15歳未満人口	15歳未満人口（男）	15歳未満人口（女）

https://www.nstac.go.jp/SSDSE/data/2020/SSDSE-2020A.xlsx

データの形式

- オープンデータなどで公開されているデータには以下のような代表的な形式があります.
  - 表計算ソフト形式
    - 表計算ソフトで読み込み可能な形式
  - CSV形式
    - データの値をカンマ（，）で区切って表したもの
    - 拡張子は.csv
  - XML形式
    - データの値をその種類を表すタグとともに表したもの
    - 拡張子は.xml
  - JSON形式
    - JavaScriptのオブジェクト表記法を元にデータを記述したもの
    - 拡張子は.json

- XML形式やJSON形式はWeb APIを用いてウェブサービスからデータを取得する際にも利用されます.

ウェブからのデータ収集における留意点

- データの信ぴょう性
  - ウェブに公開されているデータを収集する際には，元データの公開元，元データ自体の収集方法や内容，などの検証を十分に行い，データの信ぴょう性を確認する必要があります.

- バイアス
  - 収集したデータにはバイアス（偏り）が含まれる可能性があることに留意する必要があります.
    - 選択バイアス：データを集める際に観測したものと観測しなかったものの間の性質の差によって生じるバイアス
    - 情報バイアス：観測者の先入観や観測対象の過少申告や過剰反応によって生じるバイアス

- 個人情報の扱い
  - データを収集する際には，個人情報の扱いに十分に留意する必要があります．収集したデータが個人情報を含む場合は，あらかじめ利用目的を公表しておくか，または取得後速やかに利用目的を本人に知らせなければいけません.

ウェブクローラ

- 一般にウェブブラウザは以下のような流れでウェブページを取得しています.
  - ウェブブラウザのようなクライアントはウェブサーバへhttpリクエストをおくる
    - この時，http://から始まるURL (universal resource locator) を指定する
  - ウェブサーバはこのリクエストに答えてURLで指定されたコンテンツ（HTMLで記述されたファイル）を返す

- ウェブクローラは，ウェブブラウザが行うようなウェブサーバへのコンテンツのリクエストと受信をウェブのリンクを辿りながら逐次的に行うことで自動的にウェブのコンテンツを取得し蓄積するプログラムです.
  - ウェブクローラはボットやスパイダーとも呼ばれます.

- 検索エンジンではウェブクローラを用いて膨大な数のウェブページを自動で収集し索引付け（インデキシング）を行い管理しています.

ウェブクローラ

Pythonのモジュール（urllib）を用いたウェブページ取得の例

```
import urllib
response = urllib.request.urlopen('https://www.u-tokyo.ac.jp/en/')
print(response.getcode()) # HTTPステータスコードの表示
print(response.info()) # HTTPヘッダーの表示
print(response.read()) # コンテンツの表示
response.close()
```

取得するウェブページのURLを指定

```
200
Date: Tue, 30 Mar 2021 07:03:08 GMT
Server: Apache
Accept-Ranges: bytes
Access-Control-Allow-Origin: http://cdn.pr.u-tokyo.ac.jp
Access-Control-Request-Headers: *
Connection: close
Transfer-Encoding: chunked
Content-Type: text/html


b'<!DOCTYPE html>\r\n<html lang="en">\r\n    <head>\r\n        <meta charset="UTF-
8">\r\n        <meta http-equiv="X-UA-Compatible" content="IE=edge">\r\n        <title>T
he University of Tokyo</title>\r\n        <meta name="viewport" content="width=device-
width,initial-scale=1,maximum-scale=2,minimum-scale=1" user-scalable="yes">\r\n
<meta name="description" content="The official website of the University of Tokyo. Fea
tures a general introduction to the University, its research and international activities, ad
missions and other information.">\r\n        <meta name="keywords" content="The Unive
rsity of Tokyo,UTokyo,\xe6\x9d\xb1\xe4\xba\xac\xe5\xa4\xa7\xe5\xad\xa6,\xe6\x9
```


ウェブスクレイピング

- ウェブページ（一般にはHTMLで記述されたファイル）から情報を抽出することをウェブスクレイピングと呼びます.
- ウェブクローラなどで自動で収集したウェブページからウェブスクレイピングにより情報の抽出を行うことで，ウェブから自動的にデータを収集することができます.
- ウェブスクレイピングでは非構造的なウェブページから情報を抽出し，データ分析に利用可能な構造的なデータに整理します.
- ウェブクローリング・スクレイピングを行う際はウェブサーバに負荷がかからないように十分に注意する必要があります．また，サイトによってはウェブクローリング・スクレイピングを禁止している（代わりにWeb APIの利用を求める）こともあるため，事前にサイトの規約をよく確認しておく必要があります.

ウェブスクレイピング

Pythonのモジュール（Beautiful Soup）を用いたウェブスクレイピングの例

```
import urllib
from bs4 import BeautifulSoup

response = urllib.request.urlopen('https://www.u-tokyo.ac.jp/en/')
soup = BeautifulSoup(response)
response.close()
print(soup.title.text) # ウェブページのtitleタグのテキストを表示
```

```
The University of Tokyo
```

実行結果

https://www.u-tokyo.ac.jp/en/

スクレイピングにより対象ウェブページのタイトル情報を抽出

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible"/>
<title>The University of Tokyo</title>
<meta content="width=device-width,initial-scale=1,maximum-scale=2,minimum-scale=
```

ウェブページのタイトル情報

```
Tokyo. Features a general introd
tivities, admissions and other inf
ormation." name="description"/>
<meta content="The University of Tokyo,UTokyo,東京大学,東大,Todai,Japanese Universit
y" name="keywords"/>
<meta content="(C)The University of Tokyo" name="copyright"/>
<link href="/content/400132641.ico" rel="shortcut icon" type="image/x-icon"/>
<link href="/content/400130668.png" rel="apple-touch-icon" sizes="180x180"/>
<link href="/content/400132625.png" rel="icon" sizes="192x192" type="image/pn
g"/>
<meta content="The University of Tokyo" property="og:title"/>
```

スクレイピングの対象ウェブページのHTMLソース

アノテーション

- データに対してそれがどのようなデータであるかを示す情報を付加することを一般にアノテーションと呼びます．この時，付加される情報をタグまたはメタデータと呼びます．
  - XML形式のデータはXMLのタグによってアノテーションされたデータです．また，ウェブページは一般にHTMLのタグによってアノテーションされたデータとして見ることもできます．

- 例えば，オープンデータでは以下のようなメタデータがアノテーションされておりデータを検索，利用しやすくしています．
  - タイトル，組織名，作成者，タグ，公開・更新日，URL，データ形式，ファイルサイズ，使用言語，ライセンス

- 例えば，政府のデータカタログのメタデータは以下からダウンロードすることができます
  - https://www.data.go.jp/for-developer/for-developer

アノテーション

- ウェブ上の（ファイルからIoTの”モノ”に至るまでの）リソースのメタデータをアノテーションする枠組みとしてRDF（Resource Description Framework）があります.
- オープンデータをRDFなどで記述し構造化した上で相互にリンクさせて活用するための枠組みとしてLOD（Linked Open Data）があります.
- LODはウェブ上のデータを公開または利用する方式（公開されたデータそのものを指す場合もある）です．LODではデータの情報はRDFで記述され，データ間の関係を表すラベルが付与され，データ同士がリンクで結ばれたデータのウェブを形作るものです.
- LODの例として，Wikipediaから抽出した情報をLODとして公開しているDBpediaがあります.
  - DBpedia: http://ja.dbpedia.org/

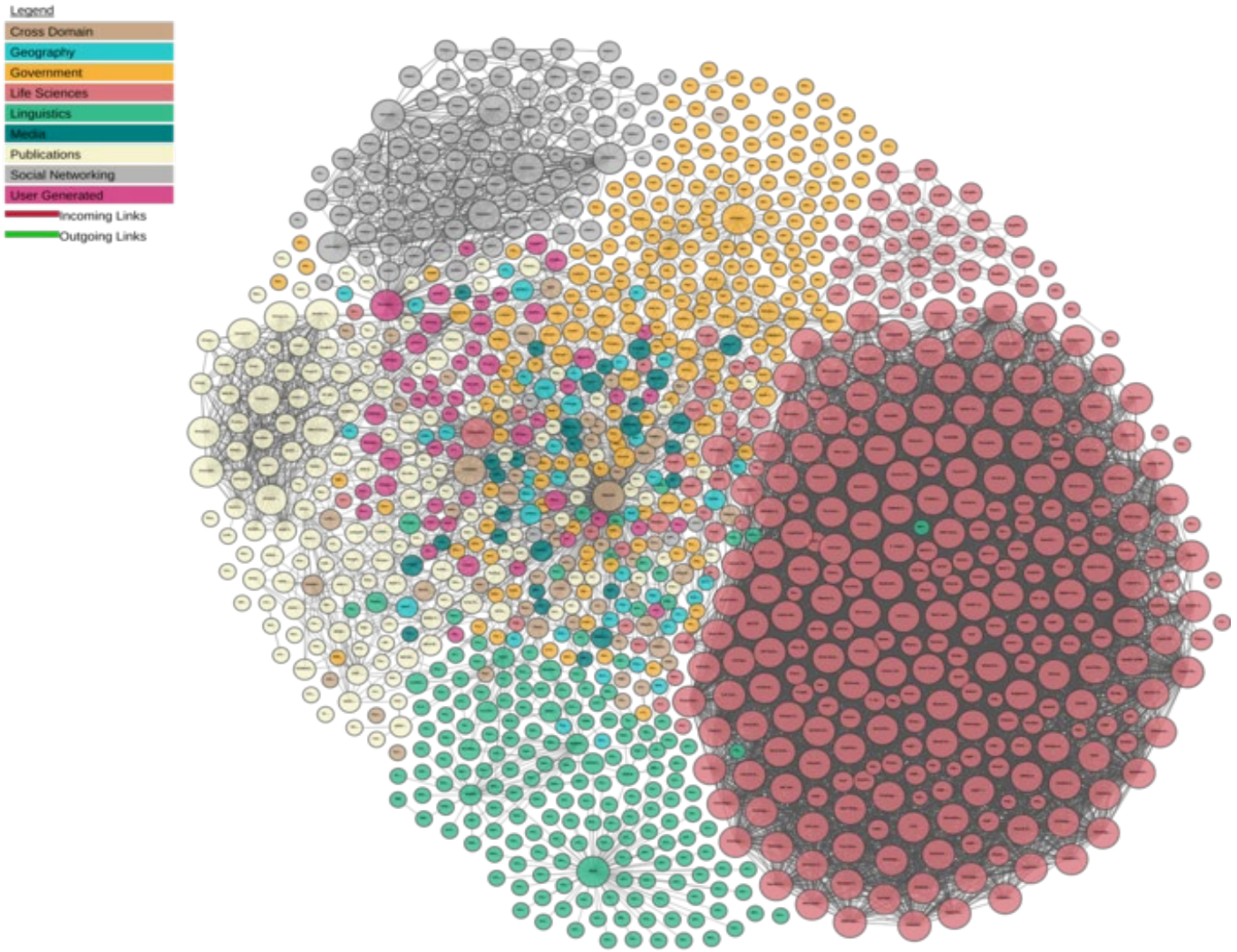
アノテーション

DBPediaの「東京大学」に関するページの例
　対象となるエンティティの情報はRDFで記述されている

http://ja.dbpedia.org/ (CC-BY-SA 3.0)

アノテーション

さまざまなLinked Open Data



https://commons.wikimedia.org/wiki/File:Lod-cloud_2017-02-20.png (CC BY-SA 3.0)