

3. インターネットのアプリケーション

1. Man-in-the-Middleで途中にCache/Proxyを置ける
2. コピーも置ける

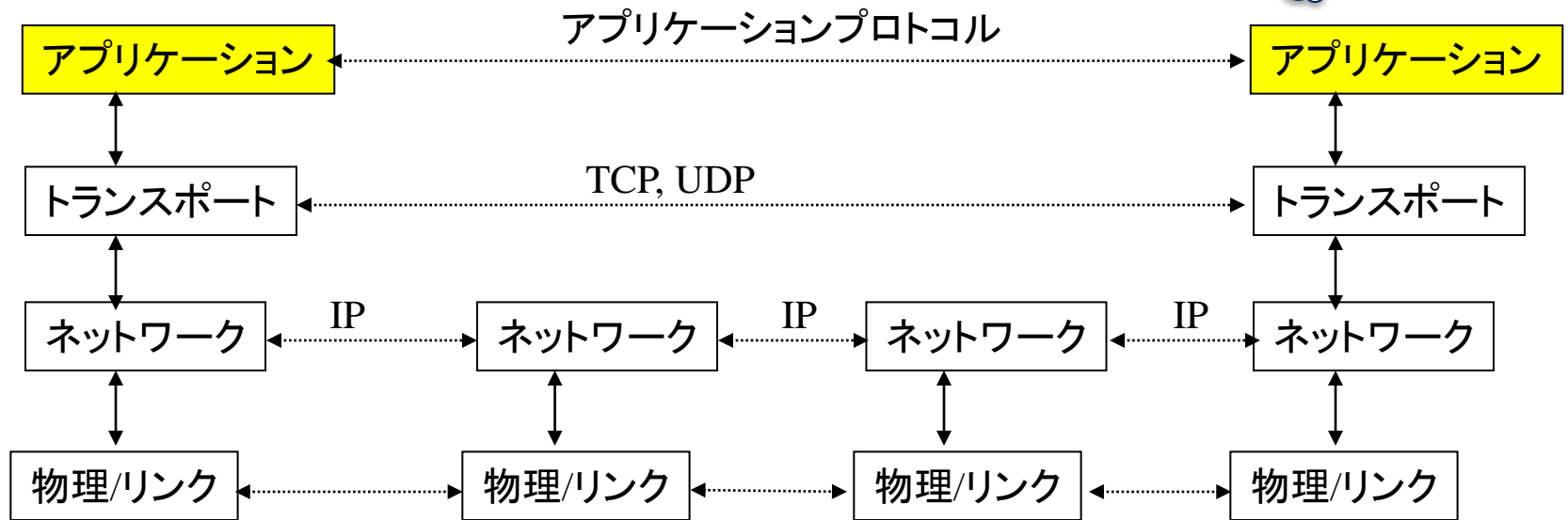


図1-12 TCP/IPの4レイヤモデル

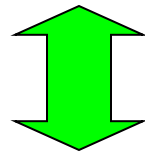
3.1 ディレクトリサービス(DNS)

DNS over UDP

- 
1. TCPに Fall-Backも
 2. 最近は、DoH/DoTも
 - DNS over https
 - 1.1.1.1 by Cloudflare
 - 8.8.8.8 by Google
 - DNS over TLS

ディレクトリサービス

人間が理解しやすい“文字列” = FQDN/URL



変換サービス = ディレクトリサービス

計算機が利用しやすい“文字列” = IPアドレス

- (*) 文字列には、なんらかの意味付けが行われる
- ・ 最近多言語化が行われた。
 - ・ ドメイン名の売買も行われている。。。。

ディレクトリサービス

- 広義のディレクトリサービス
レポジトリやレジストリの検索
レポジトリ(repository) : 貯蔵庫
レジストリ(registry) : 登録所
- インターネットにおけるディレクトリサービス
DNS正引き : FQDN → IPアドレス
DNS逆引き : IPアドレス → FQDN

Google と Yahoo の違い

- どちらも、検索エンジンからスタート
- Yahoo は、物を売ることが目的
 - 物流の仲介
 - ロングテールの法則をデジタルネットワークが覆した
- Google は、情報を集めることが目的
 - 情報の流通と加工による、価値の創出
 - コストゼロの情報をコストゼロで集める
 - 自作のコンピュータは、売らなくてもよい

Domain Name : JPRS/JPNIC が管理 (in Japan)

Host Name : システム運用者が決める

FQDN (Fully Qualified Domain Name)

= {host name} + {domain name}

URL (Uniform Resource Locator)

= {scheme} :// {FQDN}/{path}/{file name}

FQDN

scheme host name domain name

http://www.hongo.wide.ad.jp/index.html

URL

ドメイン名申請時の注意

- 時間制限有り
 - 登録後、1年以内の接続が必要
- 時間切れの場合
 - 当該ドメインが2ヶ月間凍結される
 - 利用されていないドメインとして取り扱われる

日本語ドメイン名

- ドメイン名の国際化

~~—まだ実験的な意味合いが強い~~

将来的に使えるかもしれない

使えないかもしれない

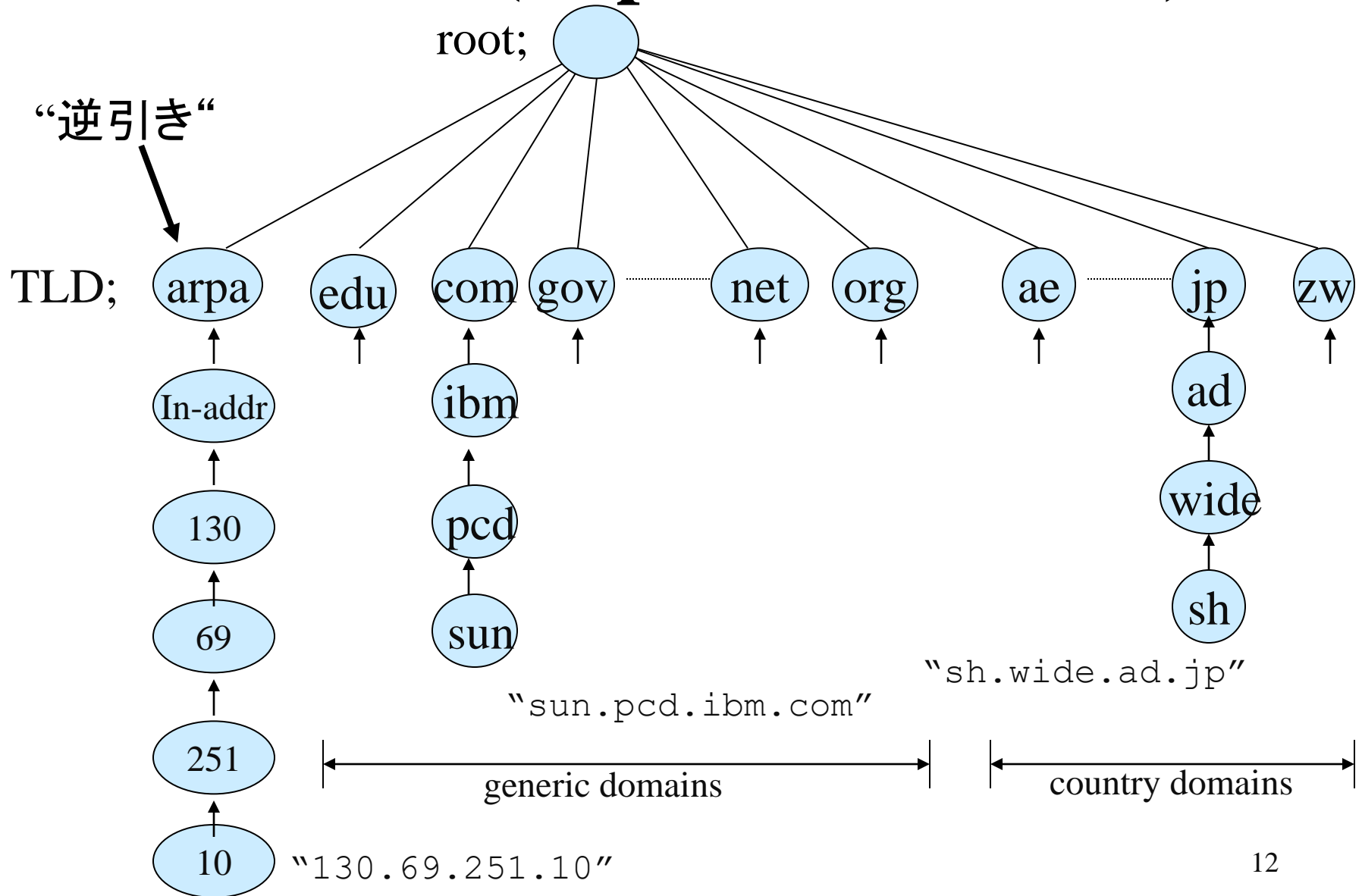
- 基本的に早い者勝ち

– おさえておくと良いかも...

新しいドメイン名

- ドット日本 (.日本)
 - `http://工学部.東京大学.日本`
- 地理的名称 g-TLD
 - `.tokyo`

DNS TLD(Top Level Domain)



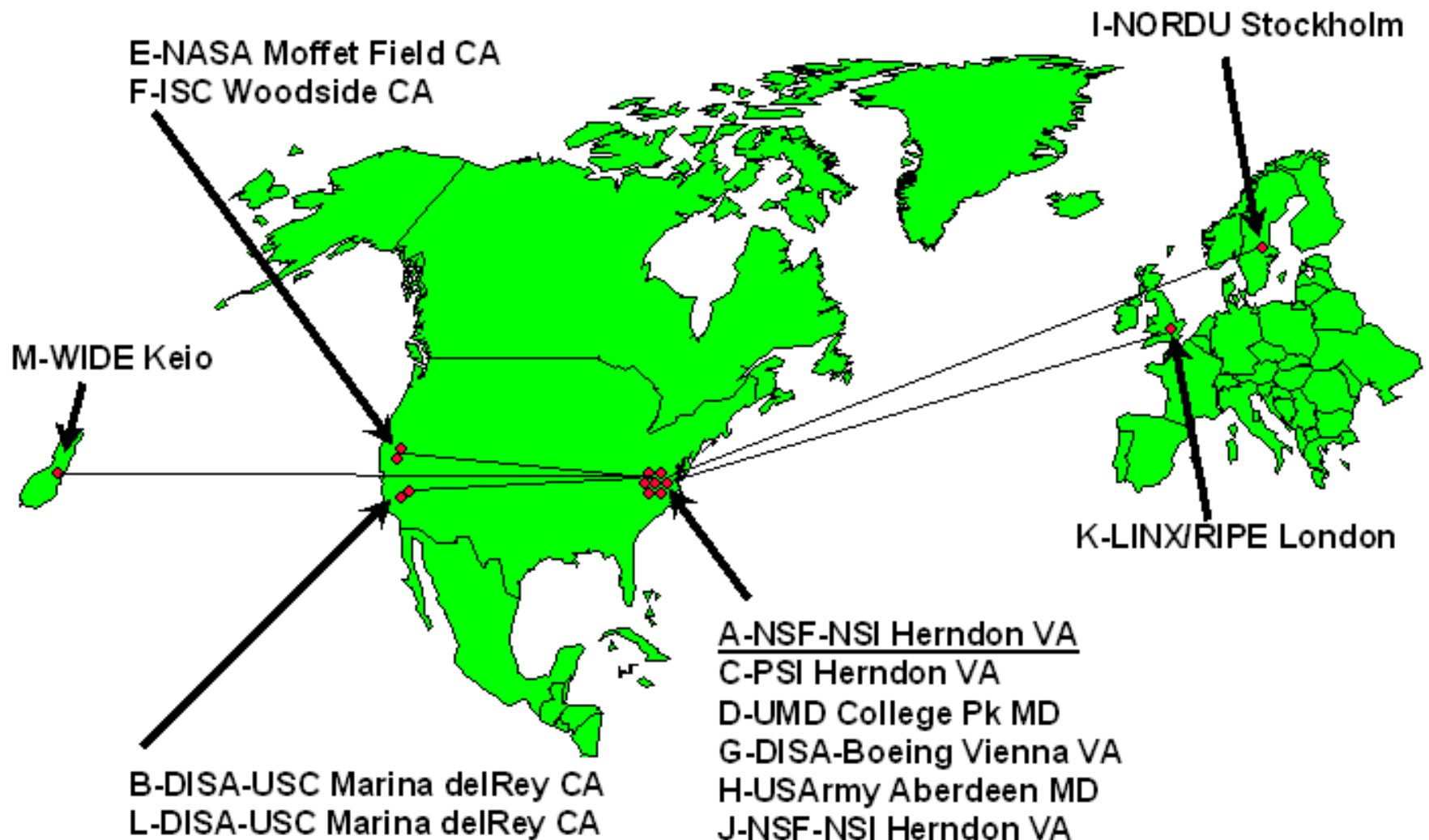
新しいトップレベルドメイン

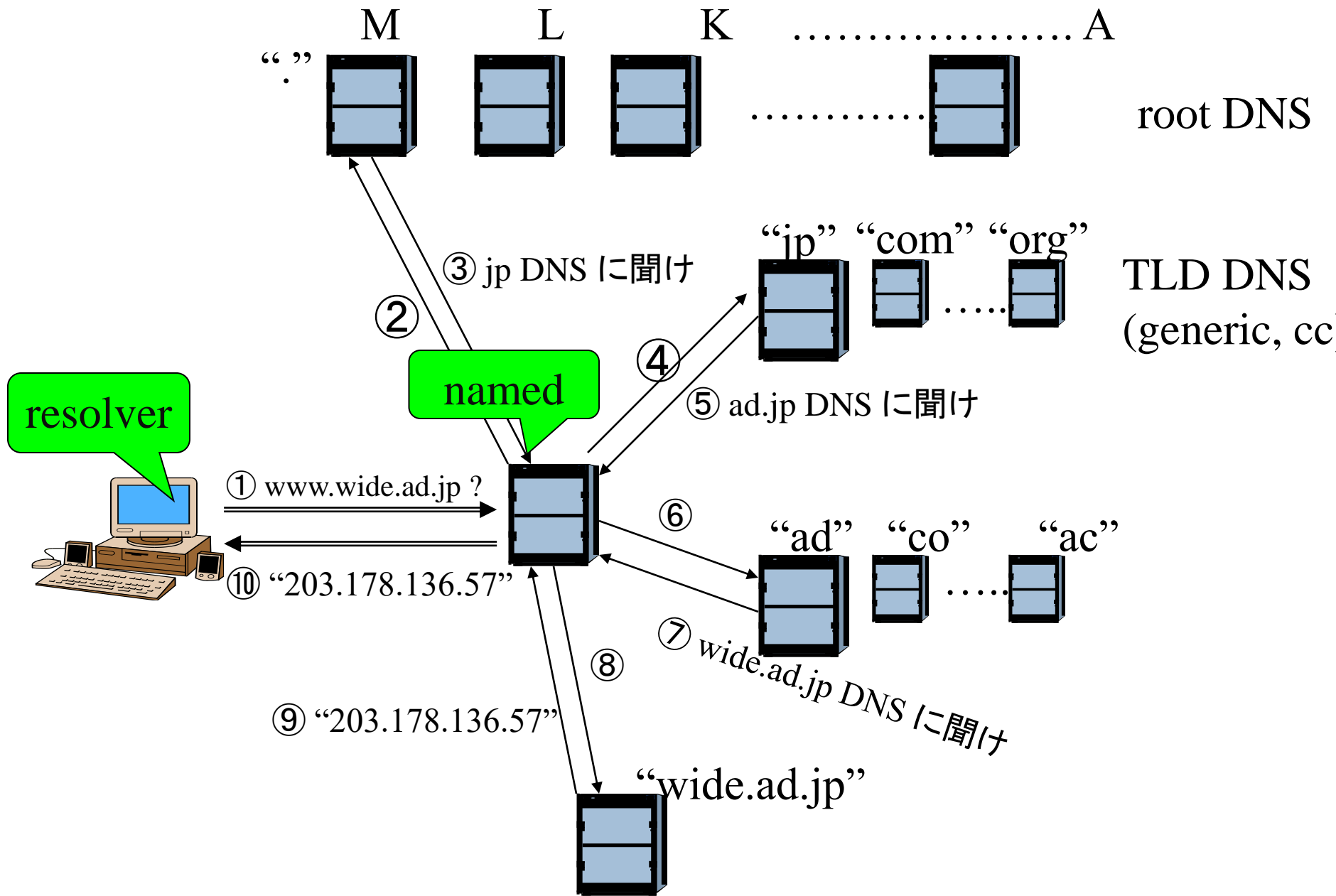
- gTLD(generic Top Level Domain)
 - **firm** for businesses, or firms
 - **shop** for businesses offering goods to purchase
 - **web** for entities emphasizing activities related to the WWW
 - **arts** for entities emphasizing cultural and entertainment activities
 - **rec** for entities emphasizing recreation/entertainment activities
 - **info** for entities providing information services
 - **nom** for those wishing individual or personal nomenclature
- <http://www.gtld-mou.org/>

DNS Root Servers

1 Feb 98

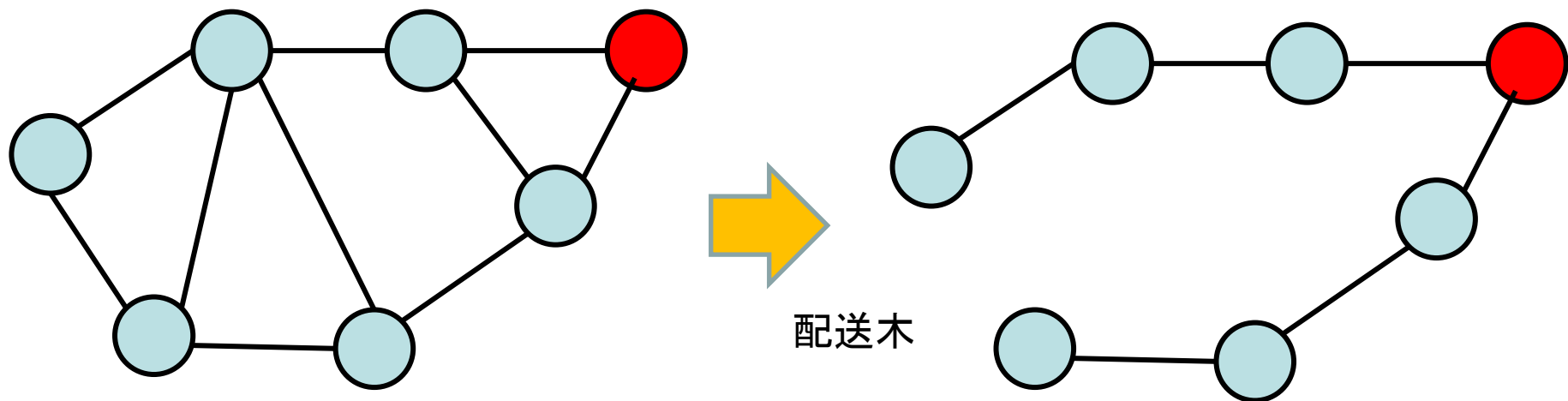
Designation, Responsibility, and Locations



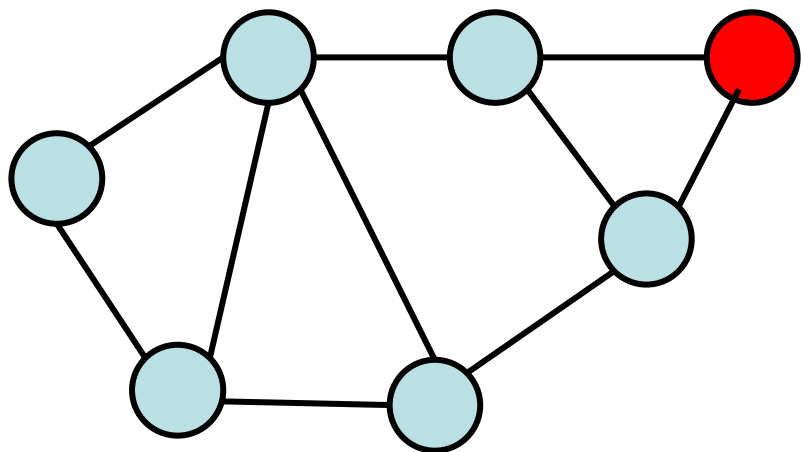


Anycast Routing

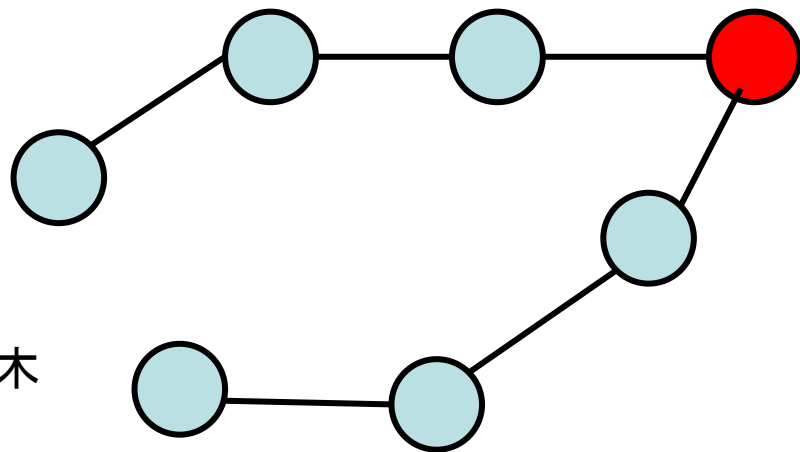
- 同じルーティングエントリー(ネットワーク情報)を、インターネット上の別の場所から広告する。
 - “自動的”に、配送木が分割される。
 - (1) “自動的”負荷分散、(2) “自動的”障害対策



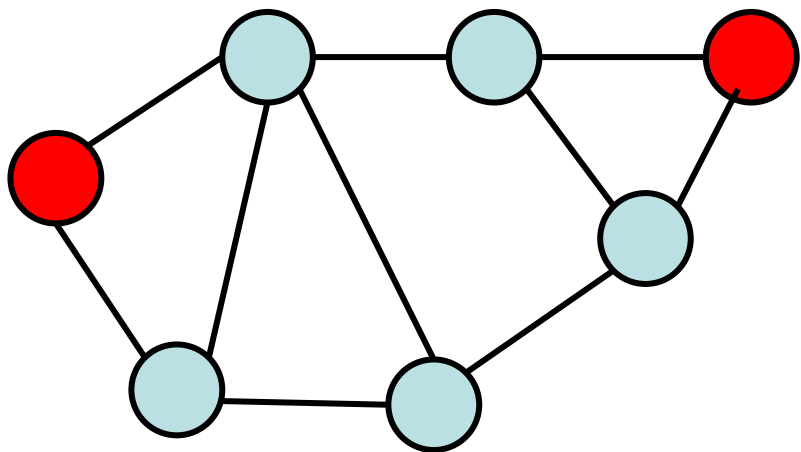
Anycast Routing



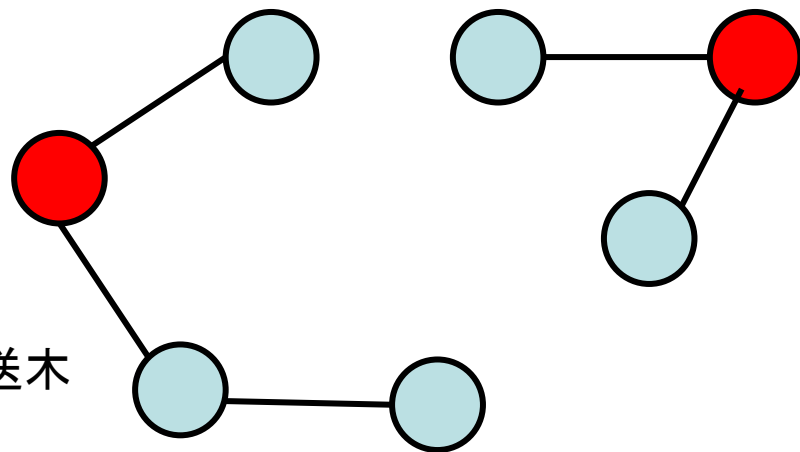
配送木

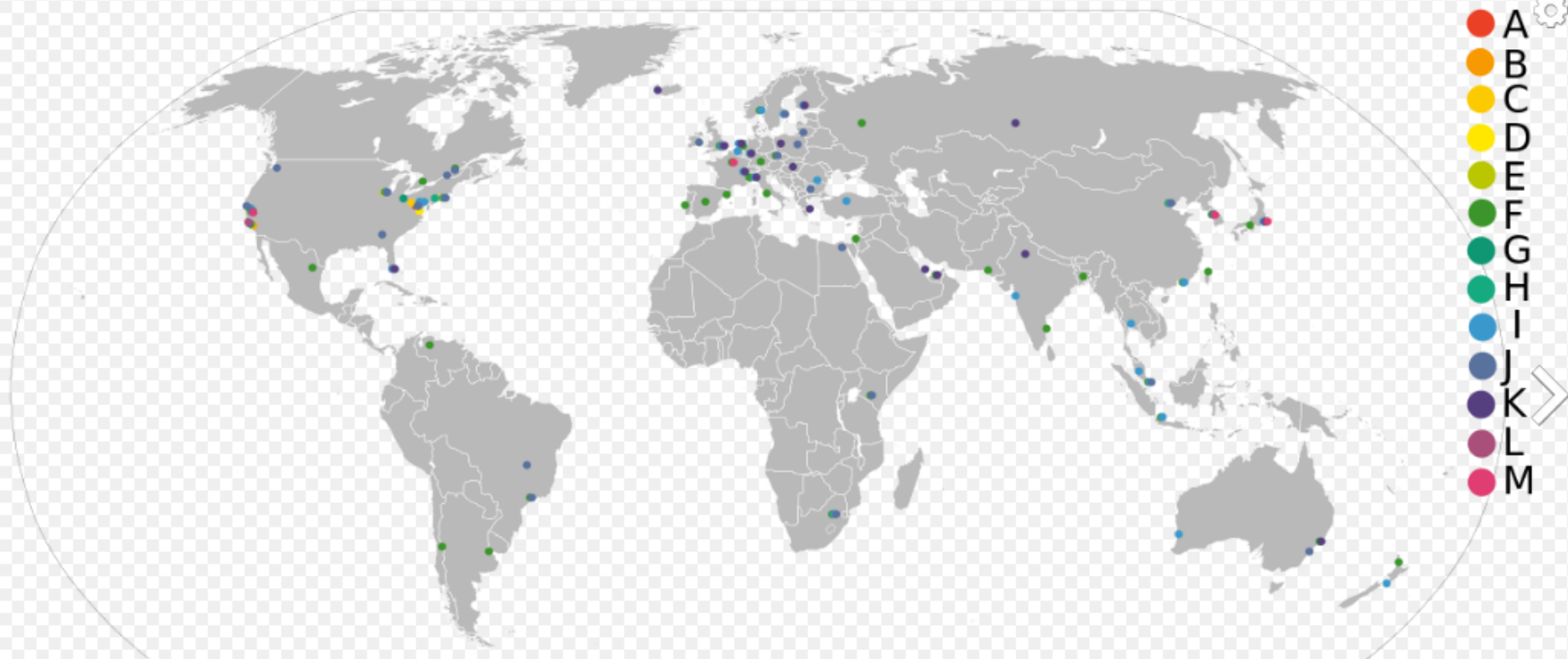


Anycast Routing



配送木





Anycast Routing を用いて、Load Distribution と Resiliency を実現している

RR(Resource Record)

- A (IPv4 Address) : 正引きIPv4
- AAAA (quad-A IPv6 Address) : 正引きIPv6
- NS (Name Server) : DNSサーバ
- MX (Mail eXchange)
- CNAME (Canonical Name) : 別名
- SOA(Start of Authority)
- PTR (Domain name Pointer) : 逆引き
- HINFO (Host Information)

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\hiroshi>nslookup
Default Server: mail.hongo.wide.ad.jp
Address: 203.178.139.66

> www.cnn.com
Server: mail.hongo.wide.ad.jp
Address: 203.178.139.66

Non-authoritative answer:

Name: cnn.com
Addresses: 64.236.16.52, 64.236.16.84, 64.236.16.116, 207.25.71.5
207.25.71.20, 207.25.71.25, 207.25.71.29, 64.236.16.20
Aliases: www.cnn.com

> www.wide.ad.jp
Server: mail.hongo.wide.ad.jp
Address: 203.178.139.66

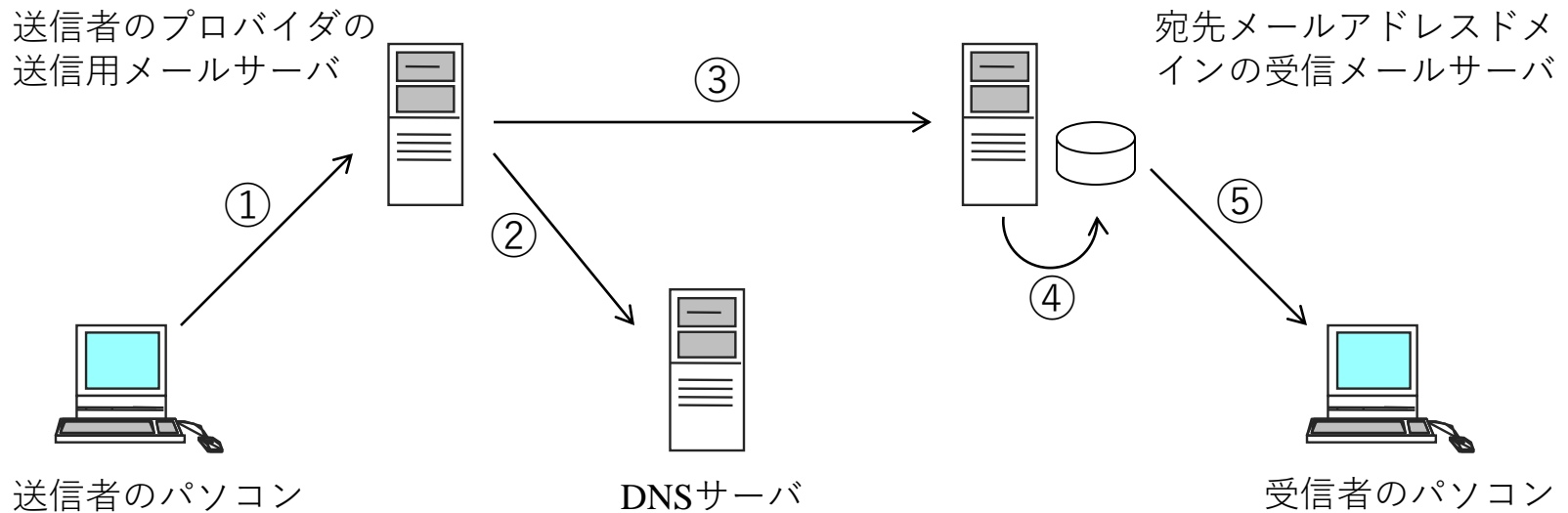
Non-authoritative answer:

Name: www.wide.ad.jp
Address: 203.178.136.57

>

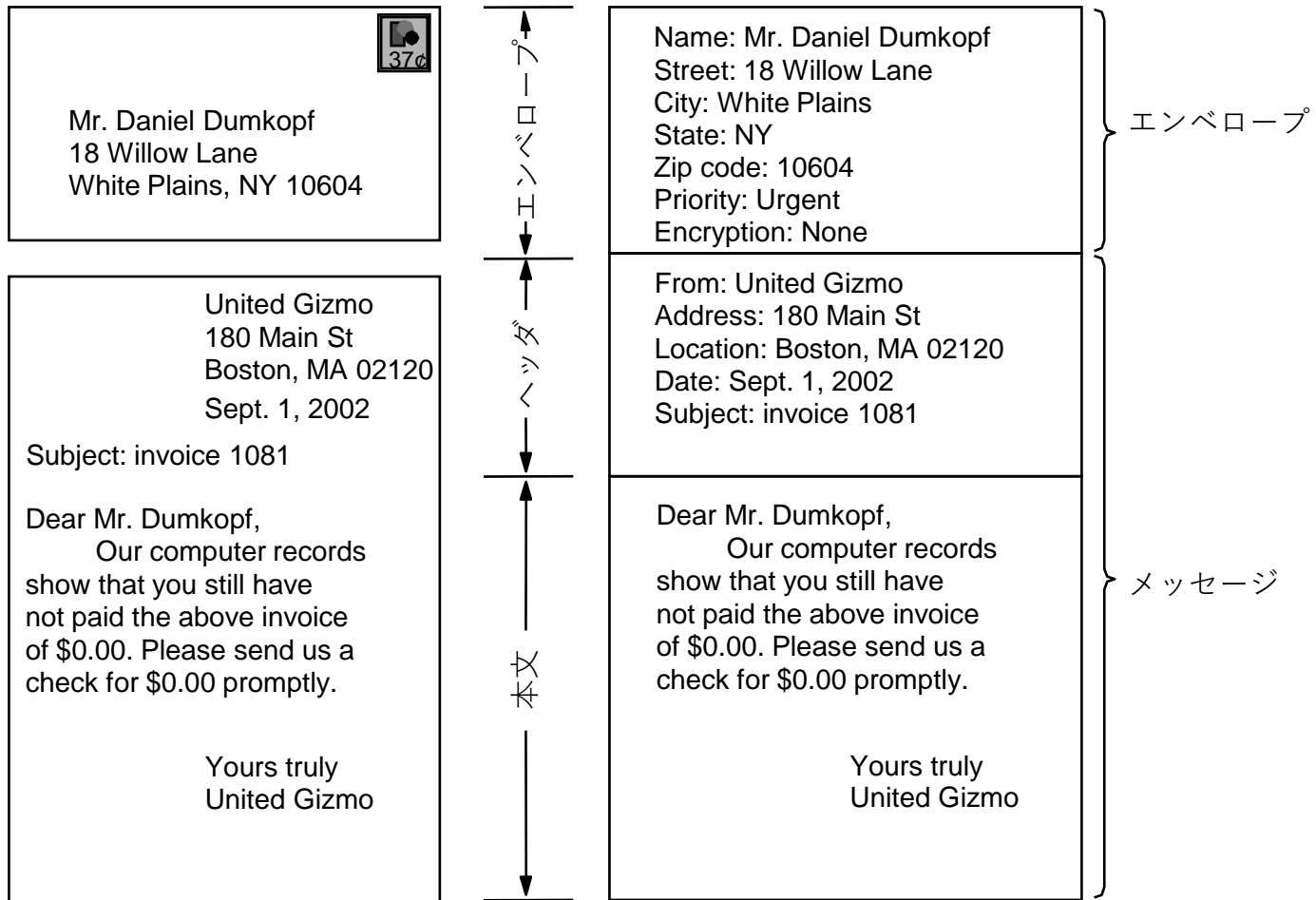
3.2 電子メール

電子メールが届くまで



- ① 送信者は加入しているプロバイダの送信メールサーバにメールを送る (SMTP)
- ② 送信メールサーバは宛先メールアドレスのドメインを担当するメールサーバをDNSで見出す
- ③ 送信メールサーバは宛先メールアドレスの受信メールサーバにメールを転送 (SMTP)
- ④ 受信メールサーバは受信したメールをディスクに格納
- ⑤ 受信者が受信メールサーバに接続して届いたメールを読み出す (POPまたはIMAP)

ヘッダ、本文、エンベロープ



(a)

(b)

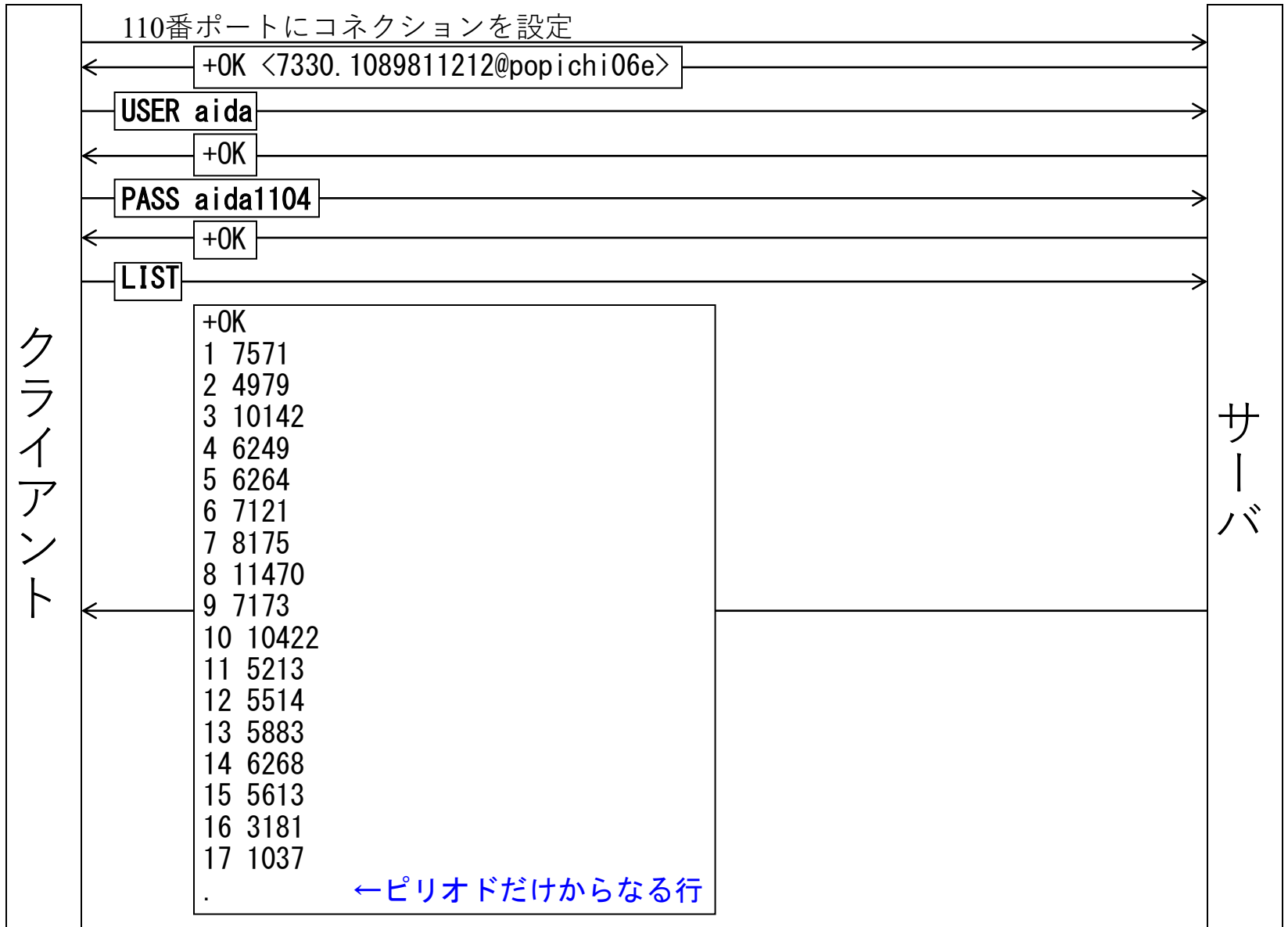
各ヘッダの意味

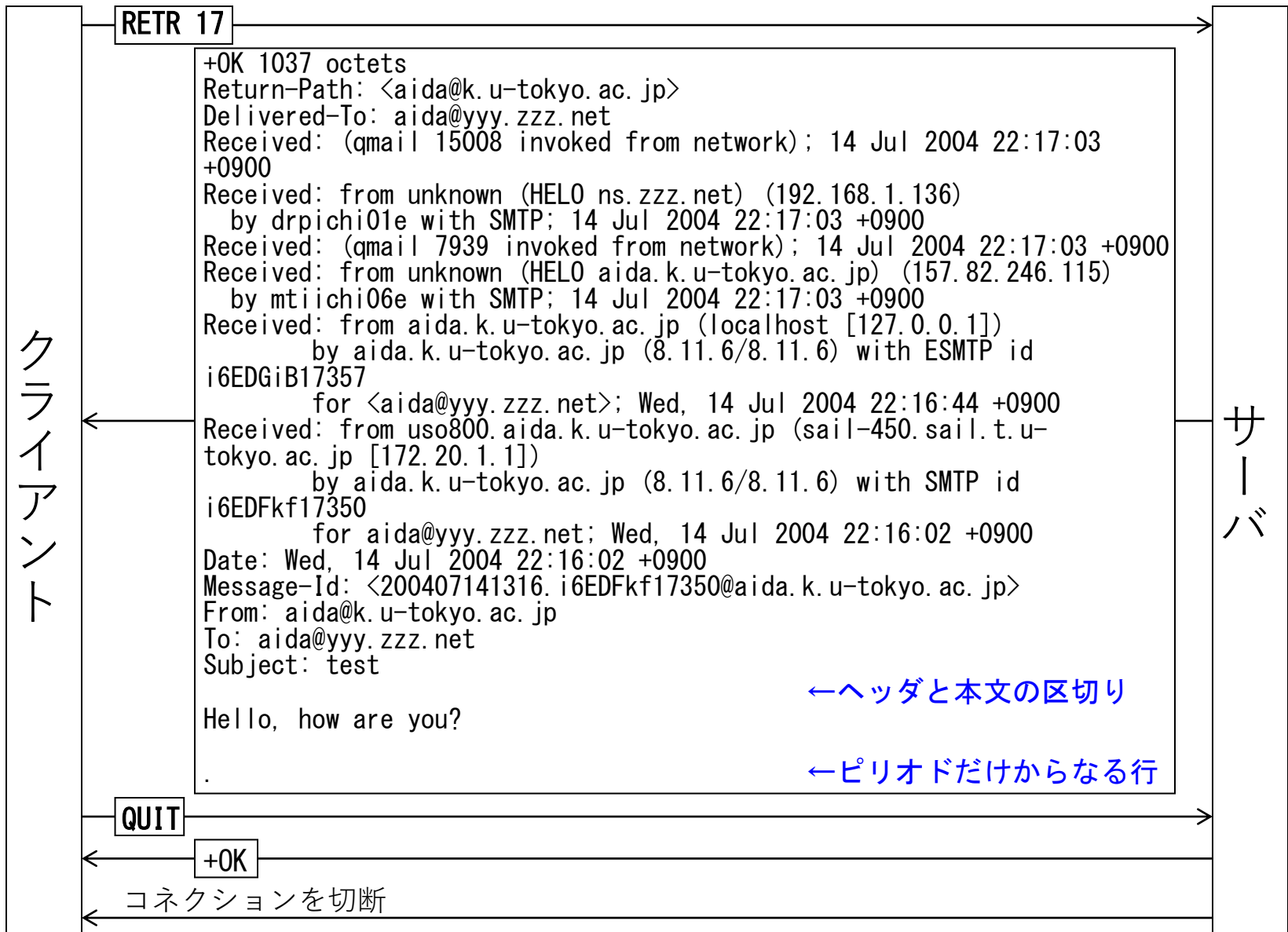
ヘッダ	意味
From:	送信元
To:	主たる宛先
Cc:	「カーボンコピー」の宛先
Bcc:	「ブラインド・カーボン・コピー」の宛先
Date:	送信日時
Subject:	題名
Sender:	実際の送信者
Reply-To:	返信先
Message-Id:	メッセージ識別子
In-Reply-To:	返信のもととなったメッセージの識別子
Received:	メールの転送履歴

SMTP, POPプロトコル

- クライアントからサーバに1行コマンドを送信するとサーバからクライアントに1行以上の応答が返ってくる
- 返答の先頭にステータスコード
- 複数行からなるデータを送るときにはピリオドだけからなる行で終わりを表す







注意すべき点

- パスワード・本文はそのまま送られる
 - APOP
 - POP over SSL
- MAIL FROM・メッセージヘッダのFromは本人確認されているわけではない
 - POP before SMTP
 - SMTP AUTH
 - PGP, S/MIME

3.3 マルチメディアデータ形式

MIME (Multipurpose Internet Mail Extensions)

- ASCII文字以外のデータを送るための仕様
 - ヘッダ中の非ASCII文字
 - 本文中の非ASCII文字 (Transfer-Encoding)
 - テキスト以外のデータ (Content-Type)
 - 複数部分からなる本文 (添付ファイル等)

Content-Typeの例

- text/plain
- text/enriched
- image/gif
- image/jpeg
- audio/basic
- video/mpeg
- application/octet-stream
- application/postscript
- multipart/mixed
- multipart/alternative

MIME Encoding-Type

Type	意味
7 bit	7ビットデータ
8 bit	8ビットデータ
binary	バイナリ-データ
BASE64	BASE64でエンコード

- ・ BASE64

6 bits 分割 → 10進表記 → 文字化

(2進表記) 01010011 00011001 01111111 ; 3 Bytes

↓
(6ビット分割) 010100 110001 110001 111111

↓
(10進表示) 20 49 49 63 ; 0~63

↓
(文字化) U x x / ; 4 Bytes

日本語の文字コード; RFC1468

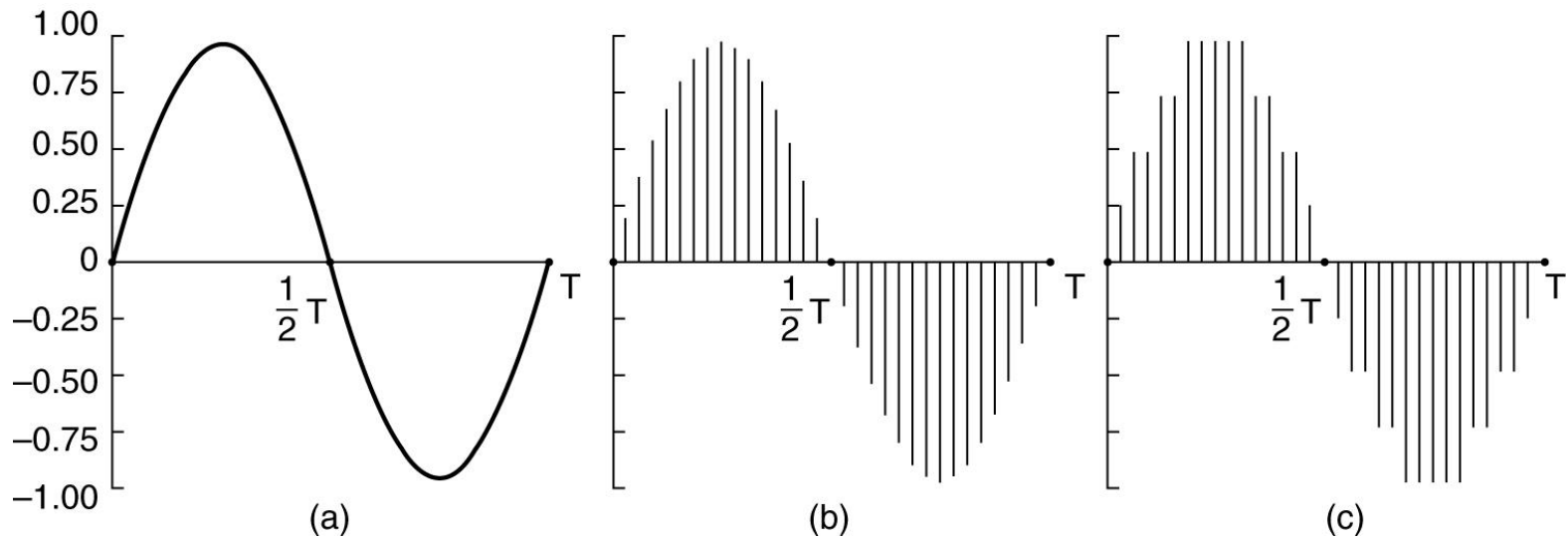
- ISO-2022-JP (JISコード)
RFC1468で定義されたコード;
電子メール/インターネットでの標準コード
(*) JUNETコード、7 bits JISコード

Content - Type: text/plain; charset=ISO-2022-JP

- EUCコード
Extended UNIX Code (UNIX系システム)
- Shift-JISコード
JIS X 0208-1990 をいくつかシフトしたコード体系

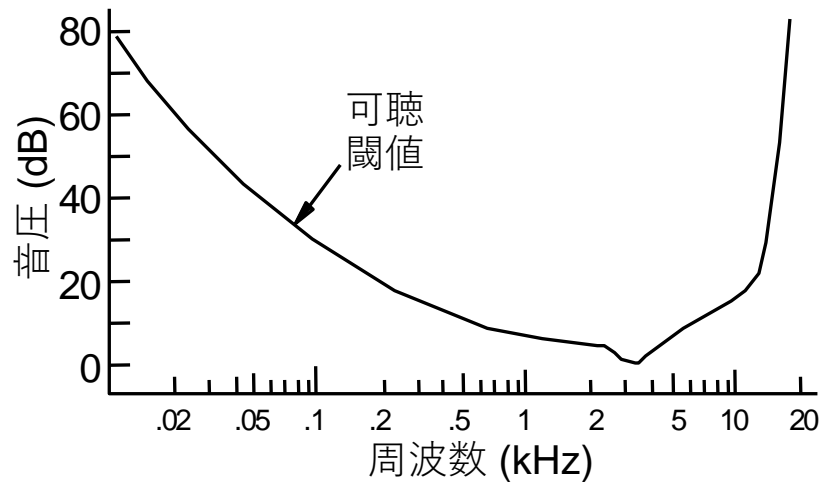
デジタル化

- 量子化
 - 必ず情報が失われる
- 標本化(サンプリング)
 - 最高周波数の2倍以上高い周波数で標本化すれば情報は失われない(ナイキストの原理)

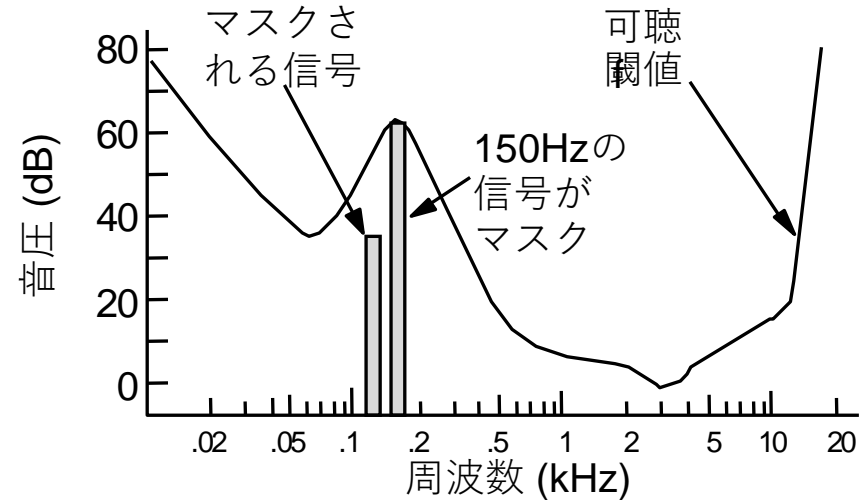


オーディオ

- CELP: 音質あまり良くない→携帯電話
- MP3: 人間の聴覚特性を利用した圧縮



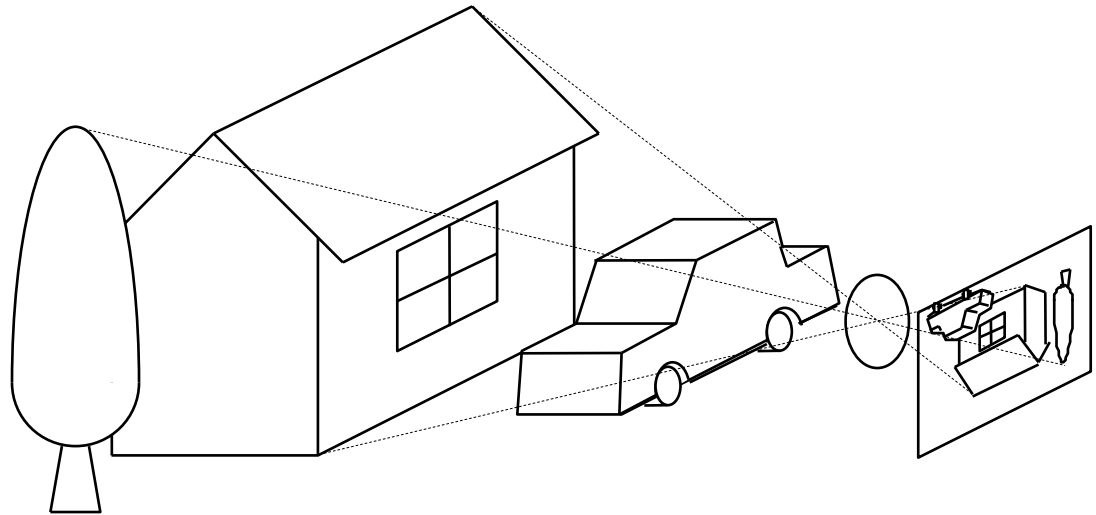
(a)



(b)

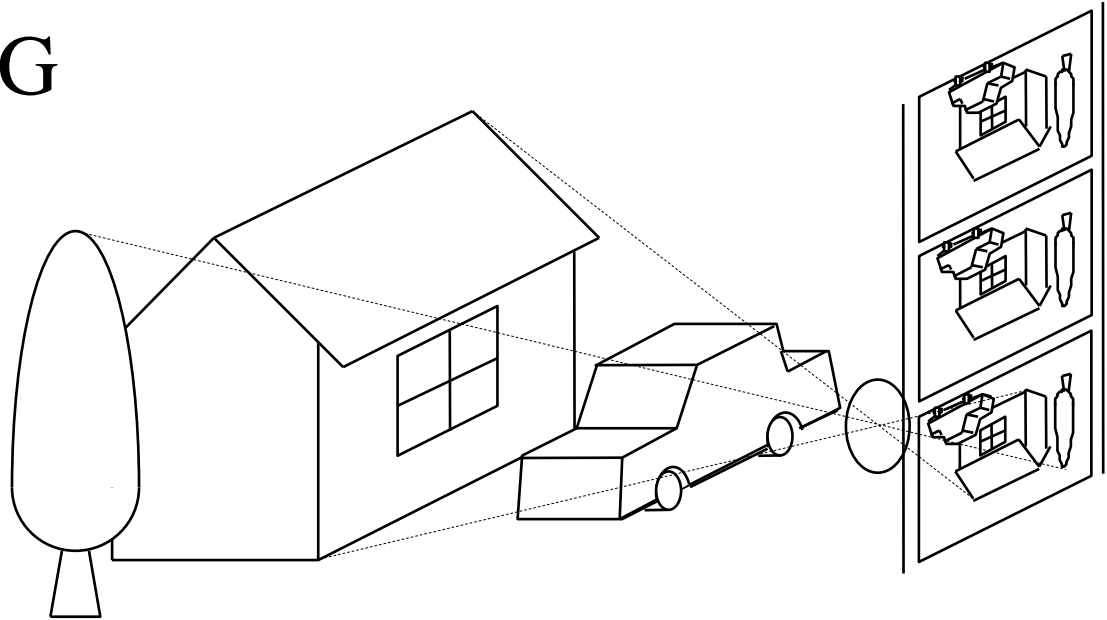
静止画像

- BMP
- GIF
- PNG
- JPEG
- JPEG-2000



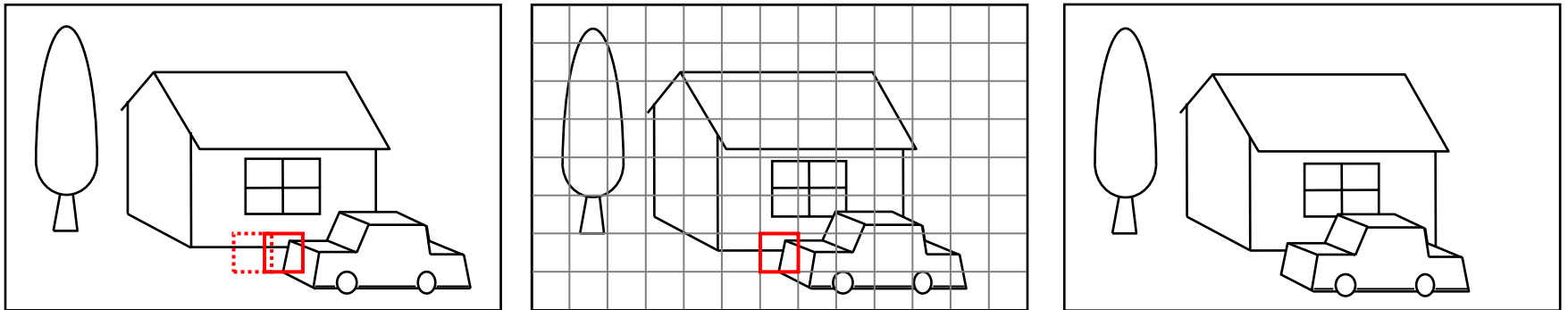
動画像

- Motion JPEG
- MPEG-1
- MPEG-2
- H.264



動き補償

- 16×16 画素のマクロブロックごとに参照フレーム内で最も似ている位置(画素または半画素単位)を探し、その変位(動きベクトル)と差分画像を符号化



ピクチャ形式

- Iピクチャ
 - 全てのブロックをJPEGと同じ方式で符号化
- Pピクチャ
 - 直前のIまたはPピクチャから予測符号化
- Bピクチャ
 - 直前のIまたはPピクチャと直後のIまたはPピクチャの両方から予測符号化

GOP (Group of Pictures)

- 定期的にIピクチャを挿入することで伝送誤りや頭出し等に対処
- I B B P B B P B B P B B P B B
- I B B P B B P B B P B B P B B P
- I P P P P P P P P P P P P P P P

グラフィックス

- Postscript
- PDF
- Flash

この方々は、ビットマップではなく、オブジェクト指向のプログラムなのです。

最近の話題.....

- ダイナミックレンジの問題
 - (DOLBY Vision)
- メディアのオブジェクト化



富士山、
見えても
撮れない

気が長い
で景色を撮影。
(川崎 川崎市)

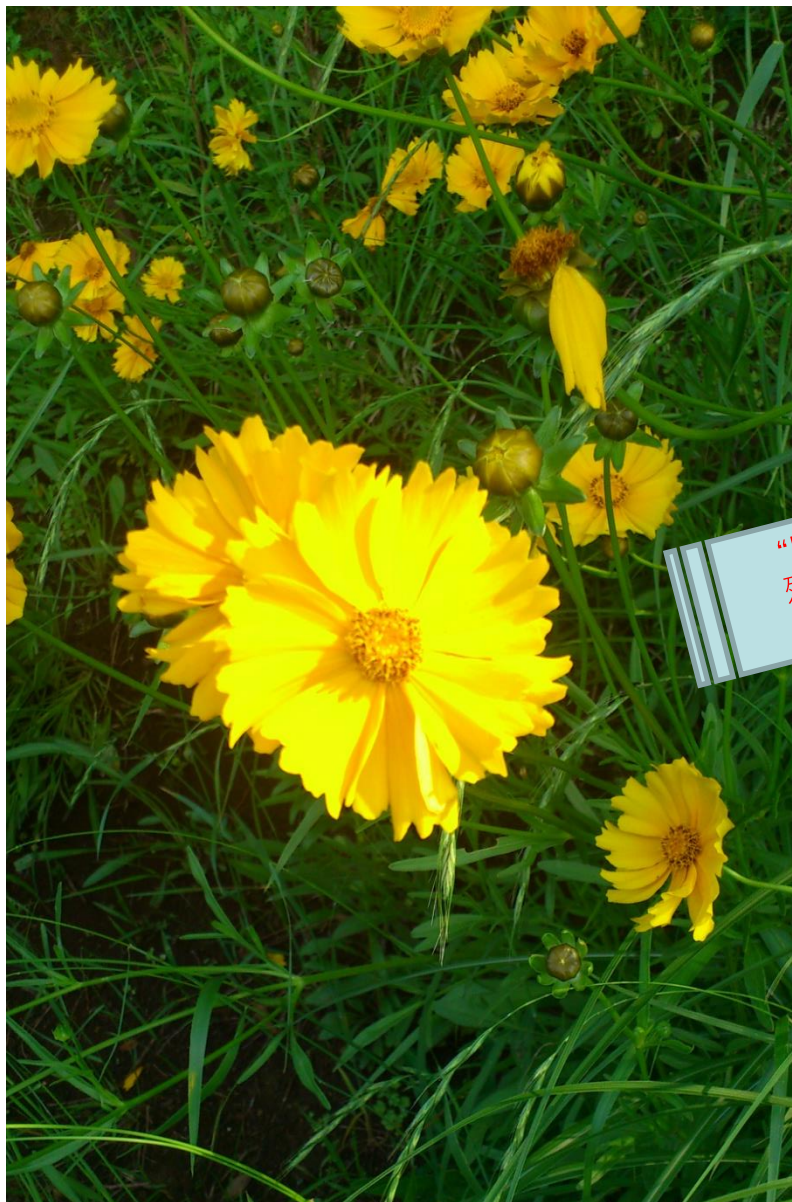
撮影範囲を
変えると
富士山は、
撮れますが,,,



**ダイナミックレンジは、
仕様には、書かれていない。。。**

2014年5月18日
EXPERIA SO-01C, Android 2.3.4,
810万画素(720p)

散歩中(May 31, 2014)
“際立つ” 綺麗な華 !!!



“際立ち”
残したい
が。。。





これ一枚に
収めたい
けども。。。

DOLBY Vision

- ダイナミックレンジの問題
 - なぜ、狭いダイナミックレンジにしたのか？
 - ➔ 帯域幅と記憶容量が小さかったから
 - (*) 映像の場合には、ディスプレイの能力

- じゃあ、こうしたくなるなあ。

5G? WiFi6も同じ。。

- ➔ 帯域幅を大きくすると？
 1. 非圧縮(圧縮度を下げて)にして、低遅延に！！
 2. Beyond Human's Capacityのセンサーを利用
 - ① 人には見えないものが見えるようにする
 - ② 情報処理構造を変える。

視覚機能

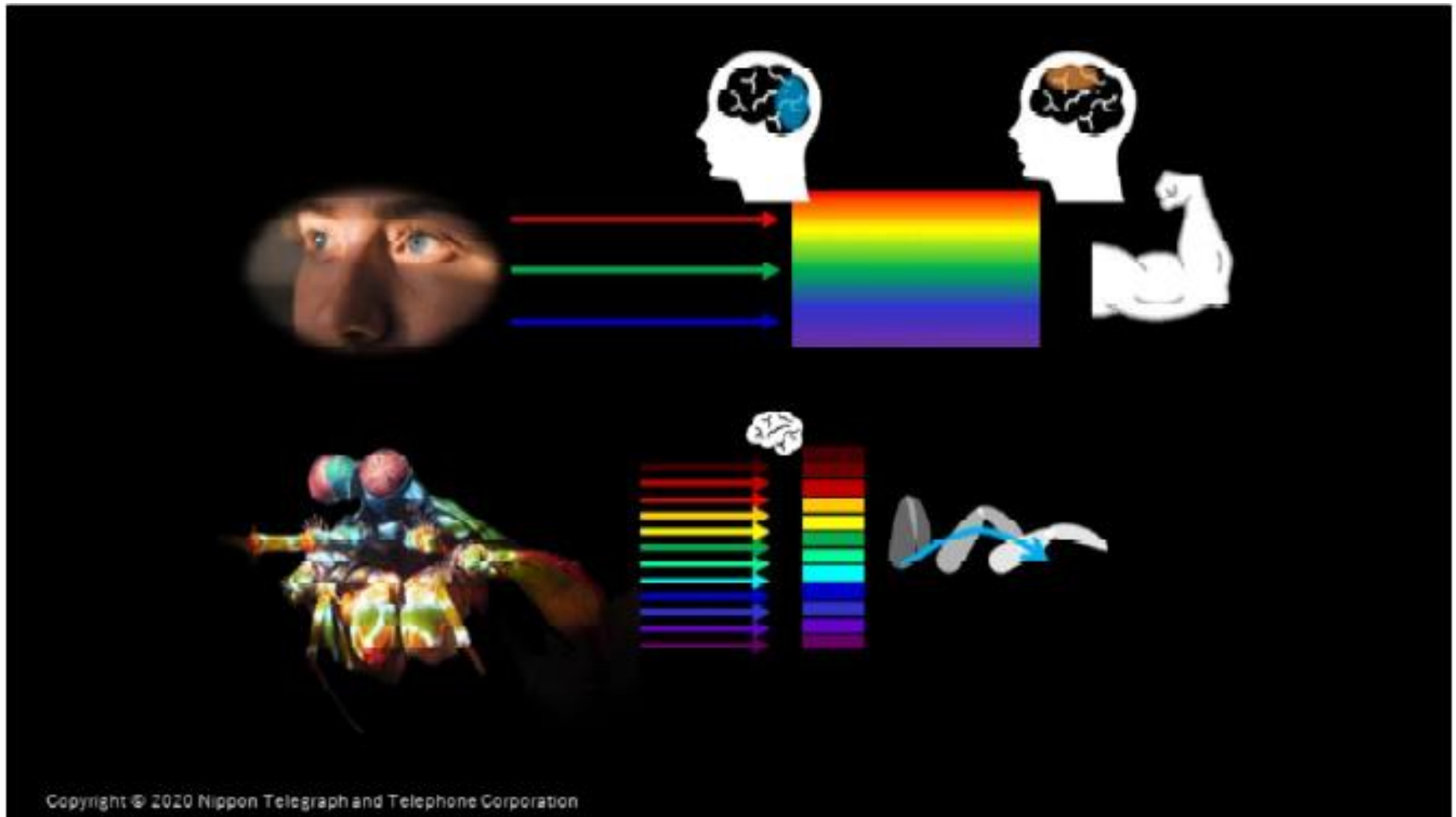




ノルウェーの科学者ビョルン・ロスレット氏が別の生物の目を想定して撮影した同じサクラソウの画像



シャコは、生物界最強の視覚システムを持っていると言われていますが、なんと12色を検知する受容体を持っています。



人が、赤青緑の3種類の受容体と脳による処理で様々な中間色を見分けているのに対し、シャコは12色の受容体と最低限の情報処理によって、非常に速い反応速度で知覚を行っています。

資料：NTT 常務執行役員 川添 雄彦 氏

「集中力」

のテストを試してみましよう。

【テスト①】





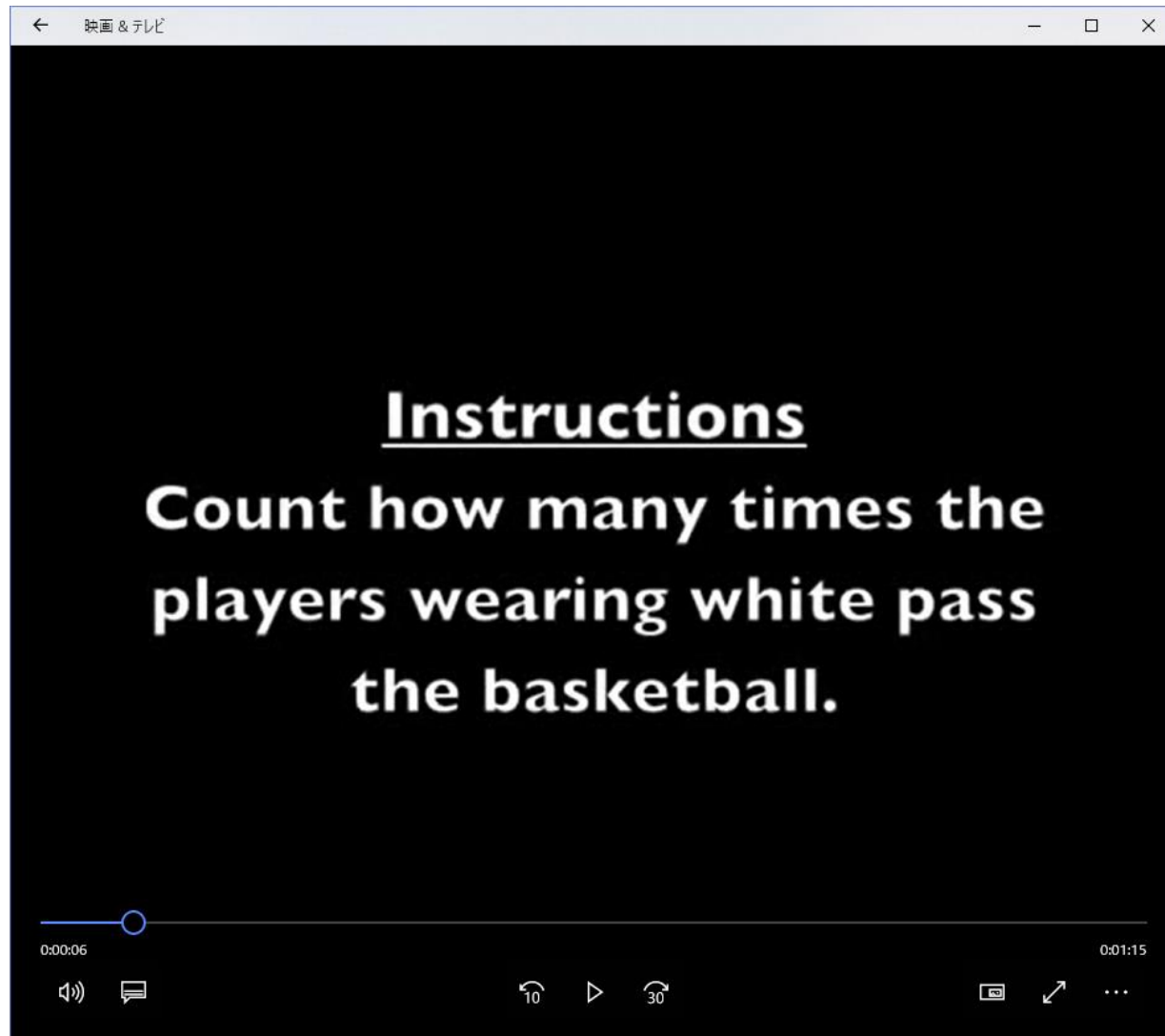


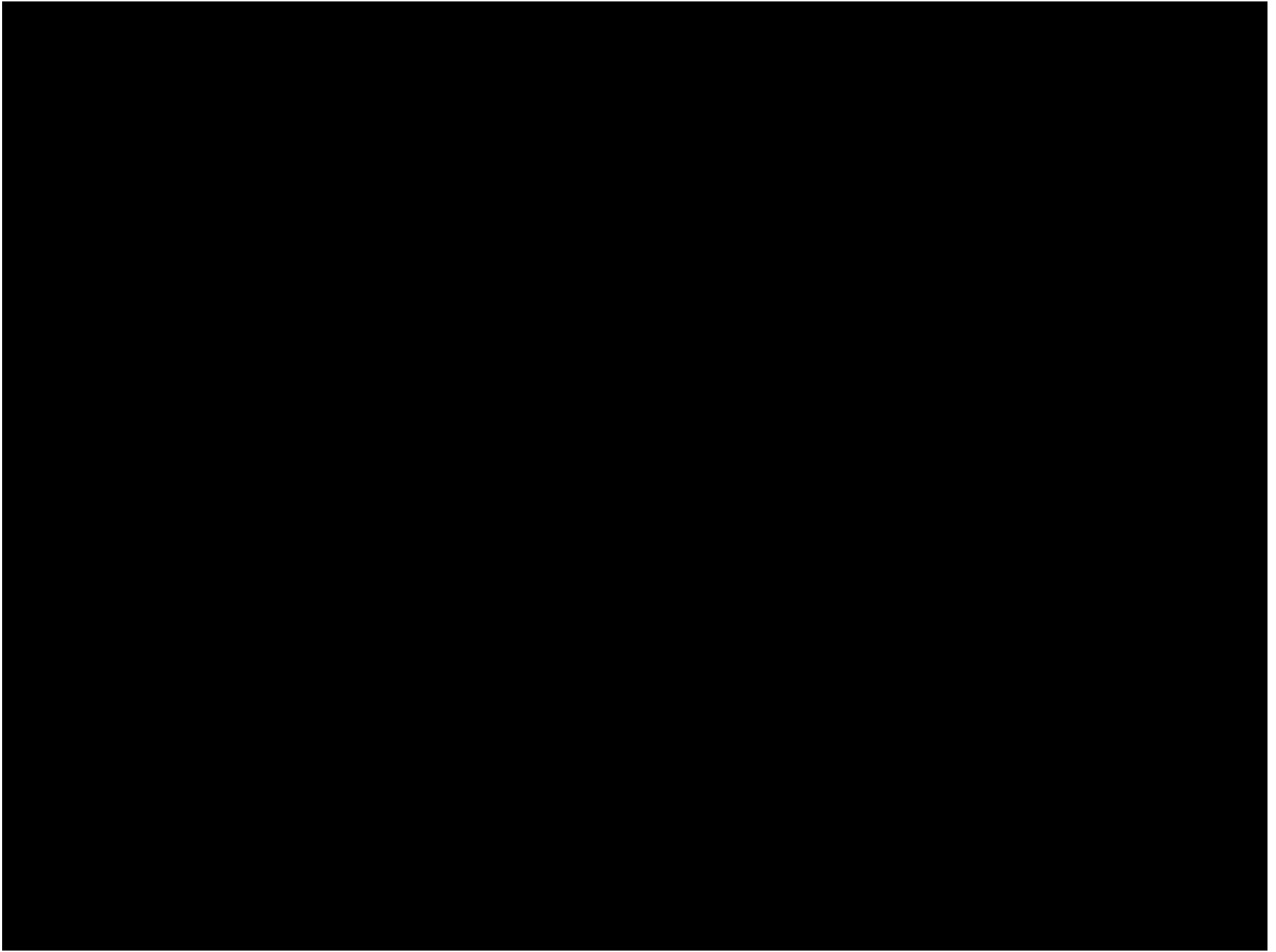
【集中力のテスト②】

白いシャツを着た人(複数)は、
何回 パスをする？

【テスト①】

<https://www.youtube.com/watch?v=P-PP35A0vHw>

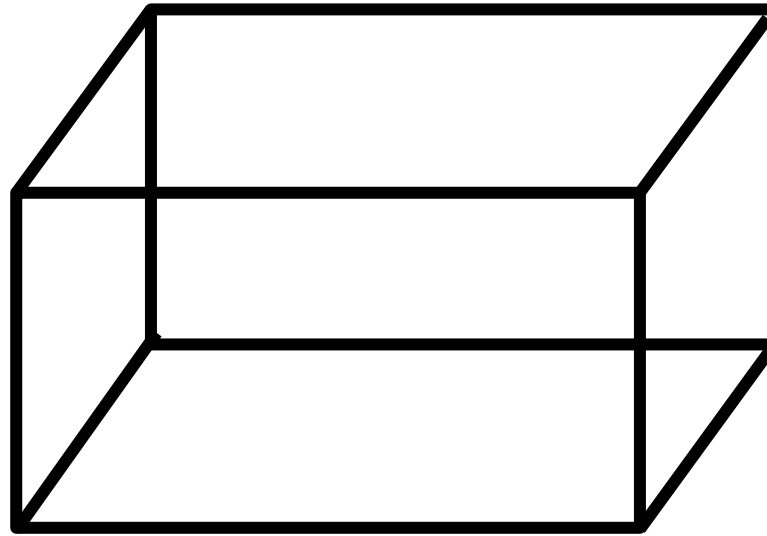




「集中力」

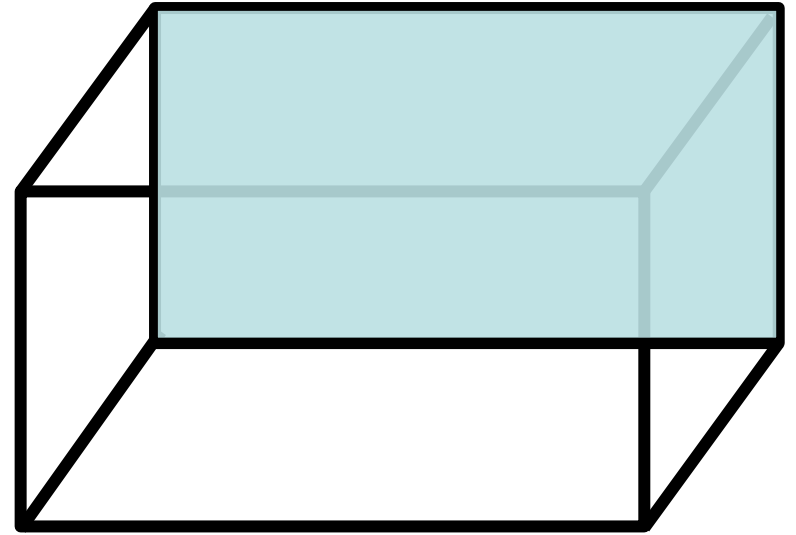
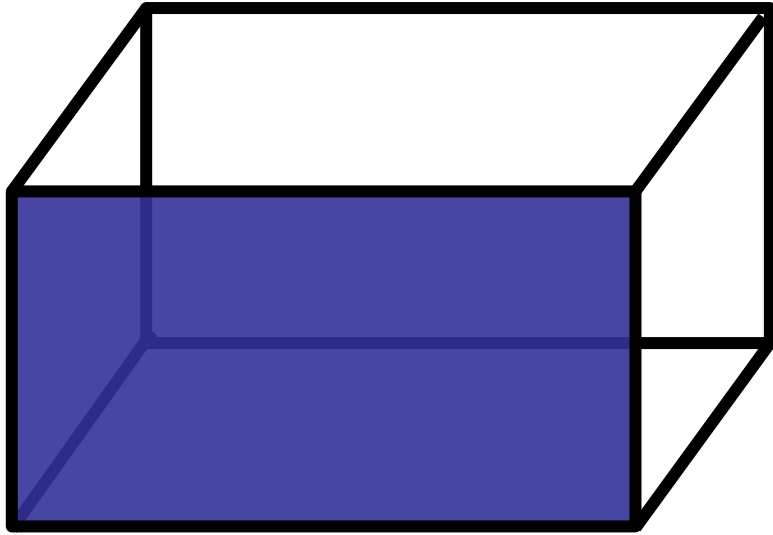
人間の脳って、不思議ですね。

【テスト③】



- 2つの立方体が見える。
- 同じものでも、見方を変えると違うものになる。
- 「利己的」な行為が、「他利的」に見えることがある。
- 「利他的」な行為は、実は「利己的」行為の結果なのである。

【テスト③】



- こういう工夫をすると、誤解が消える。

最近の話題・・・

- ダイナミックレンジの問題
 - (DOLBY Vision)
- **メディアのオブジェクト化**

最近、起こっていること。

- ビットマップ → Native Digital

- お金の (仮想化) デジタル化

- 商取引の PDF化

- 紙は、容易に改竄可能 !!!

- (*) 紙と電子ファイルどっちが改竄しやすいか？

- 契約書のプログラム化

1. アルゴリズムの自然言語化

- ① 口約束 (脳神経に音(言語)でプリントアウト)

- ② 契約書 (紙に文字でプリントアウト)

2. アルゴリズムで表記

- (*) シミュレーション(=Debugging)が可能になる。

人類が発明した情報化

1. 言語の発明 : 情報は消えてしまうもの
2. 文字の発明 : 情報は保存可能
3. お金の発明 : モノに非依存な仲介者
4. 紙の発明 : 情報は持ち歩き可能 **not digital**
5. 印刷の発明 : 情報をコピー(複製)可能 **not digital**
6. デジタルコピーの発明 : コピーしても劣化なし
7. デジタル伝送の発明 : 媒体からの解放
8. プログラム(code)の発明 : “Software Defined”

さて、、、お金(≠通貨)って???



◆そもそも、通貨は、「**価値**」を**抽象化(=デジタル化)**したもの。

◆昔は、実際に価値がある、金/銀/銅。・・・「**金本位制**」



✓ 「通貨」の合計は、全体の価値のほんの一部(≪ CF)

◆ある時  から **数字(=デジタル化)** になった。



◆ **落書きできない通貨** vs. **落書きし放題のデジタル(仮想通貨)**



◆その昔：Transactionの情報は、あまり価値がなかったけど、、、

最近：**Transactionの情報に価値が出てきたあ!!!!**



➔ **どんな金の使い方をしているのか?**

1. 要は、BS、PL、CFの情報 == 「信用情報」
2. 何には、お金をだすか? == 「購買意欲」

なんだあ、、、

Cyber/Digital First で考えると、、、今の物理実態

(Physical Instance, e.g., 貨幣)は、単なる仮の姿なんですね。

Software Defined って ?

機能(ソフトウェア)と物理的なモノ(ハードウェア)が Unbundle(アンバンドル) される。

つまり、

1. 機能が“アップグレード&追加”される
2. ハードウェアの“呪縛から解放”される

が起こる。

8. プログラム(code)の

こうなると、
One-Time の
ID、Token に進んでいる。
(*暗号化の話を思いだして見て。。。)