

Module 10

Data Processing and Text



3,587,520,067

Internet Users in the world



1,162,381,089

Total number of Websites



82,167,078,187

Emails sent [today](#)



1,819,868,567

Google searches [today](#)



1,699,641

Blog posts written [today](#)



231,740,961

Tweets sent [today](#)



2,104,290,745

Videos viewed [today](#)
on YouTube



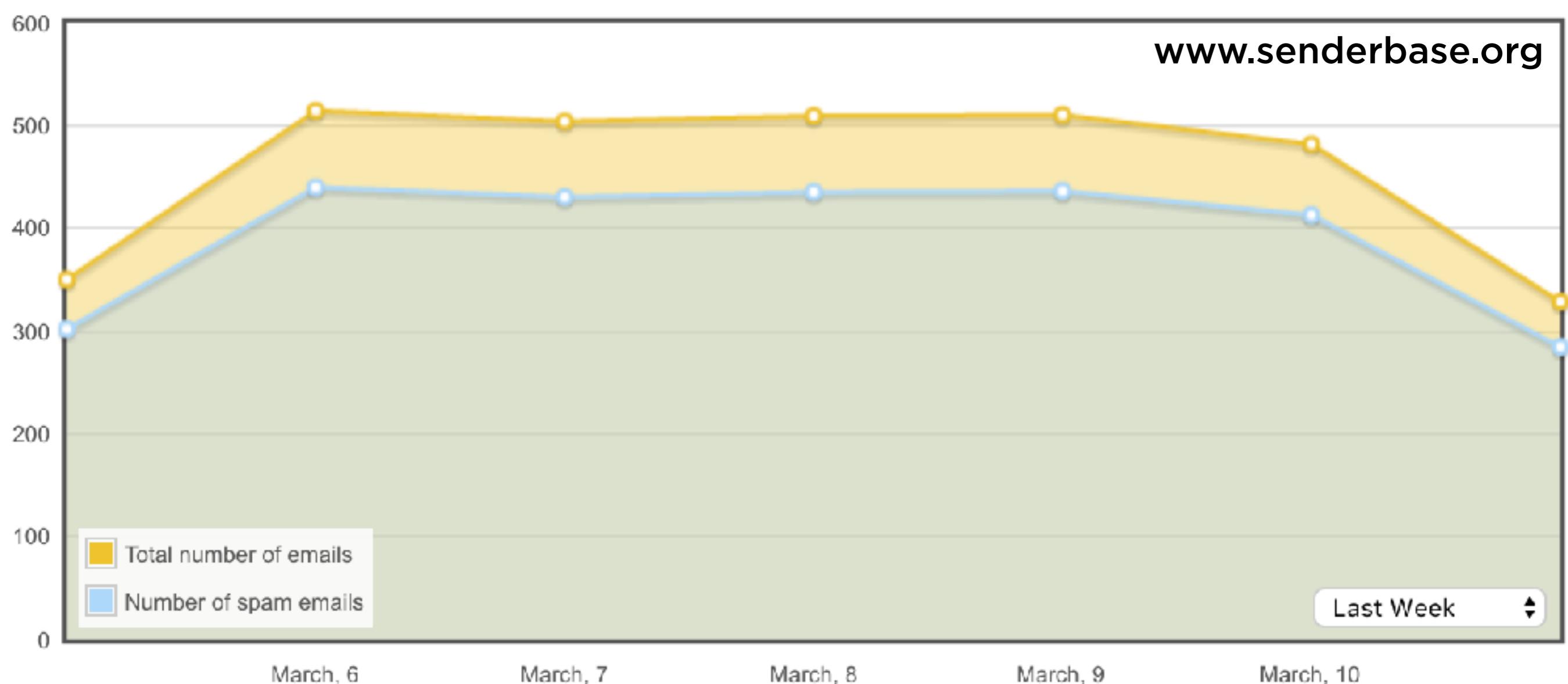
23,682,137

Photos uploaded [today](#)
on Instagram



37,451,716

Tumblr posts [today](#)



Date	Average Daily Spam Volume (Billions)	Spam Volume Change
March, 5	301.96	-2% ▼
March, 6	439.80	+46% ▲
March, 7	430.39	-2% ▼
March, 8	435.23	+1% ▲
March, 9	436.14	0%
March, 10	412.95	-5% ▼
March, 11	284.47	-31% ▼

TRAFFIC FROM MOBILE & ONLINE MESSAGING TO REACH 438 BILLION PER DAY BY 2019

Hampshire, UK: 6th July 2015: New data from Juniper Research has shown that mobile and online messaging traffic will reach 160 trillion per annum by 2019, up from 94.2 trillion this year – equating to approximately 438 billion messages sent and received by users on a daily basis by 2019. These figures incorporate SMS, MMS, IM (Instant Messaging), Social Media and Email.

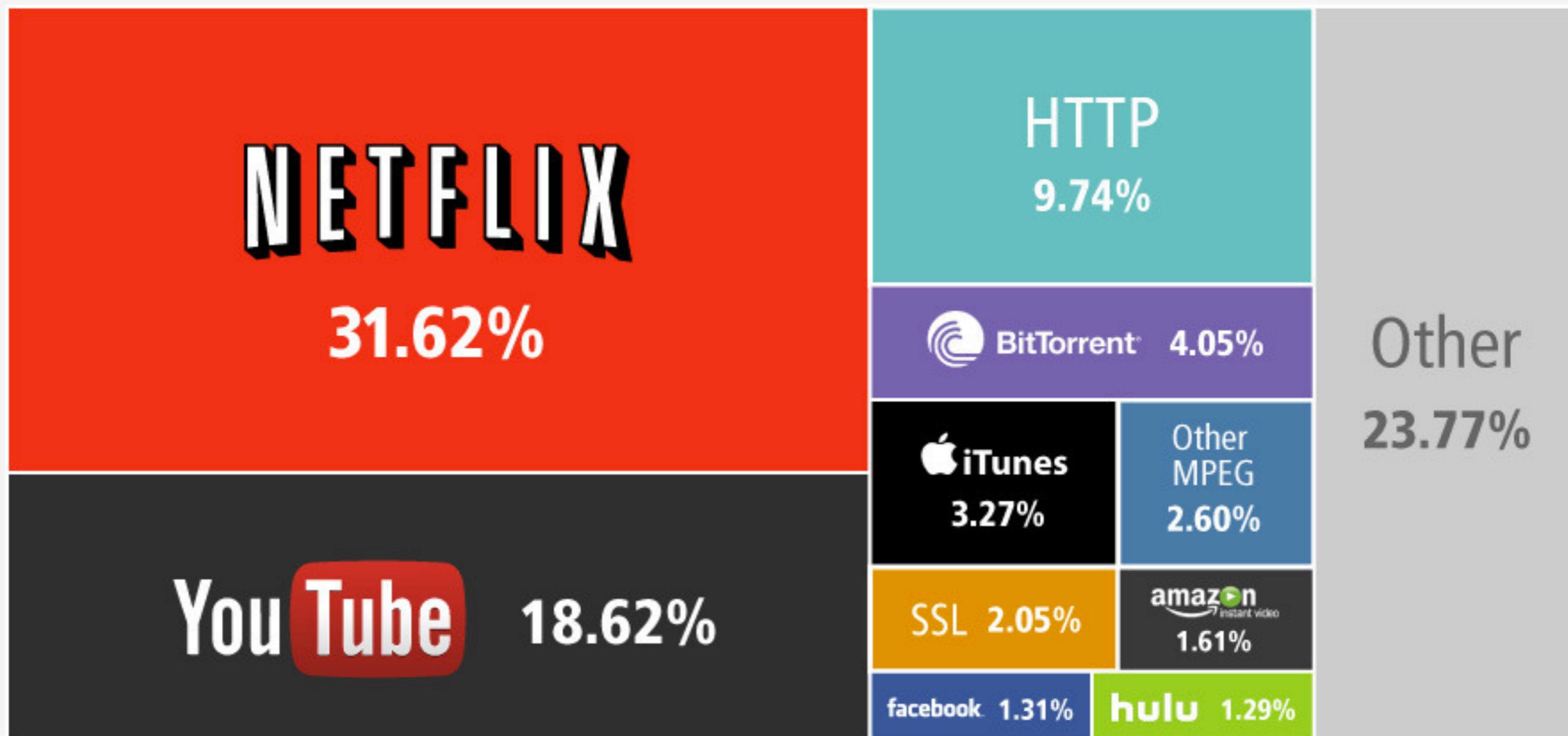
Last year, email accounted for the largest share of traffic, at around 35 trillion messages per year – although almost 80% of this figure (28 trillion) can be categorised as spam. However, within the next 12 months IM will overtake email generating almost 43 trillion messages annually.

YouTube Company Statistics	Data
Total number of people who use YouTube	1,325,000,000
Hours of video uploaded to YouTube every minute	300 hours
Number of videos viewed on YouTube everyday	4,950,000,000
Number of unique visits to YouTube every month	900,000,000
Total number of hours of video watched on YouTube each month	3.25 billion hours
Number of YouTube videos that have generated over 1 billion views	10,113
Percent of YouTube visitors that come from outside the U.S.	70 %
Number of countries with localized versions of YouTube	42
Total number of languages YouTube is broadcast in	54
User submitted video with the most views - "Charlie bit my finger"	829,000,000
Average number of mobile YouTube video views per day	1,000,000,000
Average time spent on YouTube per mobile session	40 minutes
Average YouTube partner channel payout per 5,000 views	\$0.32

YouTube Company Statistics	Data
Total number of people who use YouTube	1,325,000,000
Hours of video uploaded to YouTube every minute	300 hours
Number of videos viewed on YouTube everyday	4,950,000,000
Number of unique visits to YouTube every month	900,000,000
Total number of hours of video watched on YouTube each month	3.25 billion hours
Number of YouTube videos that have generated over 1 billion views	10,113
Percent of YouTube visitors that come from outside the U.S.	70 %
Number of countries with localized versions of YouTube	42
Total number of languages YouTube is broadcast in	54
User submitted video with the most views - "Charlie bit my finger"	829,000,000
Average number of mobile YouTube video views per day	1,000,000,000
Average time spent on YouTube per mobile session	40 minutes
Average YouTube partner channel payout per 5,000 views	\$0.32

Netflix and YouTube Are America's Biggest Traffic Hogs

Share of peak period downstream traffic in North America, by application*



* September 2013. Fixed access only.



Data challenges

- Creating it
- Storing it
- Moving it around
- Keeping it private

Data challenges

- Creating it
- Storing it
- Moving it around
- Keeping it private
- **Making sense of it**

Searching, indexing



Collecting, Correlating, Recommending

The image displays three screenshots from mobile applications demonstrating data handling:

- Left Screenshot (Yelp):** Shows a search results page for "Sklepy z zabawkami war..." (Toys stores). It lists six local businesses with their names, ratings, addresses, and price ranges. Below the list are navigation buttons for "Strona główna", "W pobliżu", "Wyszukaj", "Zakłady", and "Zamówienia".
- Middle Screenshot (Yelp):** Shows a detailed view for "Toster". It includes a map showing the location on Warszawska 14, Kraków, MA 31-155, Śródmieście. Below the map are buttons for "Wskazówki dojazdu" and "Więcej danych".
- Right Screenshot (Spotify):** Shows the "Discover Weekly" playlist interface. It features a grid of mood and genre cards such as Mood, Party, Pop, Workout, Focus, Rock, Indie/Alternative, EDM/Dance, Chill, Sleep, Hip Hop, R&B, Country, Folk & Americana, Metal, Soul, Travel, 80s, Decades, and Jazz. On the right side, it shows a list of songs from various artists.

Collecting, Correlating, Recommending



[All](#)[News](#)[Images](#)[Videos](#)[Books](#)[More ▾](#)[Search tools](#)

About 570,000,000 results (0.64 seconds)

Online Advertising - Convert Shoppers with Relevant Ads

Ad www.criteo.com/Mobile ▾

Make More Sales with Criteo Today.

10,000 brands · 130 countries

Services: Transparent CPC Pricing, Unparalleled Technology, Dynamic Creative, Cross ...

Contact Us

Tell Us A Little About Yourself.
Find a Local Office Near You.

What We Do

Driving Better Marketing Results.
Learn About Our Technology Today

Advertising Online - Marketing360.com

Ad www.marketing360.com/OnlineAdvertising ▾ +1 855-773-8169

#1 Marketing Platform® For Advertising Online. Free Demo!

Online Advertising - Reach More Customers Online

Ad www.rogersoutrank.com/Client_Leads ▾

Demo OutRank's Powerful Platform.

Online advertising - Wikipedia, the free encyclopedia

https://en.wikipedia.org/wiki/Online_advertising ▾

Online advertising, also called online marketing or Internet advertising or web advertising, is a form of marketing and advertising which uses the Internet to ...

Display advertising - Web banner - Mobile advertising - Paywall

Rose repeats to Hammerbacher—who's a founder of data analytics company Cloudera—a line from [an interview he gave](#) *Businessweek* back when he was an early employee hustling stats for Harvard bud Zuckerberg at Facebook:

“The best minds of my generation are thinking about how to make people click ads.”

And Rose, in his politeness, left off the last part of the line:
“That sucks.”

Patterns, trends, predictions

proofpoint>

Find Not Spam Release Allow Sender Options ▾
Username: csk@uwaterloo.ca

Quarantine			
Spam - Quarantined			
	Score	From	Subject
<input type="checkbox"/>	99	vanzari@aramisfeeling.ro	Cadoul tau e la un cl
<input type="checkbox"/>	99	Aden@phwwemoty.org	*****SPAM***** To I
<input type="checkbox"/>	99	rzljqzoaeqywpx@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	fymvsmqxcof@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	mwhnksic@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	amy-manges@goodsnfits.com	*****SPAM***** See
<input type="checkbox"/>	99	vcxtuprei@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	syxlhxsg@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	vlvorxrcdk@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	rjtirt@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	sjwsyf@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	nxrvgohxqbewg@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	sdsjocuiv@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	laryfhakexa@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	wcjhpjmopxvk@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	mmprig@cgl.uwaterloo.ca	*****SPAM*****
<input type="checkbox"/>	99	wtlhx@cgl.uwaterloo.ca	*****SPAM*****

Mailboxes Inbox VIPs Drafts (95) Sent (1) Flagged

Sort by Date

Hang w/	2015-03-15
Welcome To Hang w/ - Private Con Man Access...	
Humble Bundle	2015-09-10
Your Humble Bundle order is ready	
Con Man Productions	2016-04-18
😂 Con Man Commentary Tracks with Alan Tudyk...	
Con Man Productions	2016-05-18
🥳 Stretch Goal Reward: Free Spectrum Issue #0!	
Cye Waldman	2016-07-04
🔗 [The Tiling List] Voderberg rant	
XVideos	2016-11-27
Your account on Xnxx.com	
Official Pokémon GO Team	2017-01-06
Videos: Ditto, New Pokémon and Fun Facts for T...	
Ubisoft Account Support	2017-01-17
Welcome to Ubisoft	
FirstMediaX Info	2017-01-22
Informasi mengenai Link Account 9995805904...	
CanadianHelps	2017-02-07
Your charitable tax receipt for 2016 is ready!	
Kitchener-Waterloo Symphony	2017-03-04
Renew your 2017/18 subscription by March 17 fo...	
billing@canadianwebhosting.com	2017-03-09
Customer Invoice	

32 95 1

Mailboxes Inbox VIPs Drafts (95) Sent (1) Flagged

Sort by Date

Hang w/	2015-03-15
Welcome To Hang w/ - Private Con Man Access...	
Humble Bundle	2015-09-10
Your Humble Bundle order is ready	
Con Man Productions	2016-04-18
😂 Con Man Commentary Tracks with Alan Tudyk...	
Con Man Productions	2016-05-18
🥳 Stretch Goal Reward: Free Spectrum Issue #0!	
Cye Waldman	2016-07-04
🔗 [The Tiling List] Voderberg rant	
XVideos	2016-11-27
Your account on Xnxx.com	
Official Pokémon GO Team	2017-01-06
Videos: Ditto, New Pokémon and Fun Facts for T...	
Ubisoft Account Support	2017-01-17
Welcome to Ubisoft	
FirstMediaX Info	2017-01-22
Informasi mengenai Link Account 9995805904...	
CanadianHelps	2017-02-07
Your charitable tax receipt for 2016 is ready!	
Kitchener-Waterloo Symphony	2017-03-04
Renew your 2017/18 subscription by March 17 fo...	
billing@canadianwebhosting.com	2017-03-09
Customer Invoice	

32 95 1

Official Pokémon GO Team
Videos: Ditto, New Pokémon
To: ezemoney126441
Reply-To: Official Pokémon



Welcome To [MalaysianCupid.com!](http://MalaysianCupid.com)



MalaysianCupid.com

Dear Member

Congratulations on joining MalaysianCupid.com!

This email contains your logon information and important information that will help you get the most out of your membership and help you to find your perfect partner.

YOUR LOGIN DETAILS:

Login Email: azriel@gmail.com

Your Password is: **ayang78**

Please save this email for future reference!



Thanks for your payment, processed on June 02, 2015.



Hello, AZRIEL TEPPER

American Express Card
81009



We processed your
scheduled payment.



\$987.91

PROCESSED ON

June 02, 2015

It's processed today - but give us 24 -
36 hours for your payment to appear
online.

[View your account.](#)

Thanks for your Card Membership,
American Express Customer Care

Was this e-mail helpful? [Give us your feedback](#)

STAY CONNECTED



[@AskAmex](#)

Amex Customer Care, at your service.



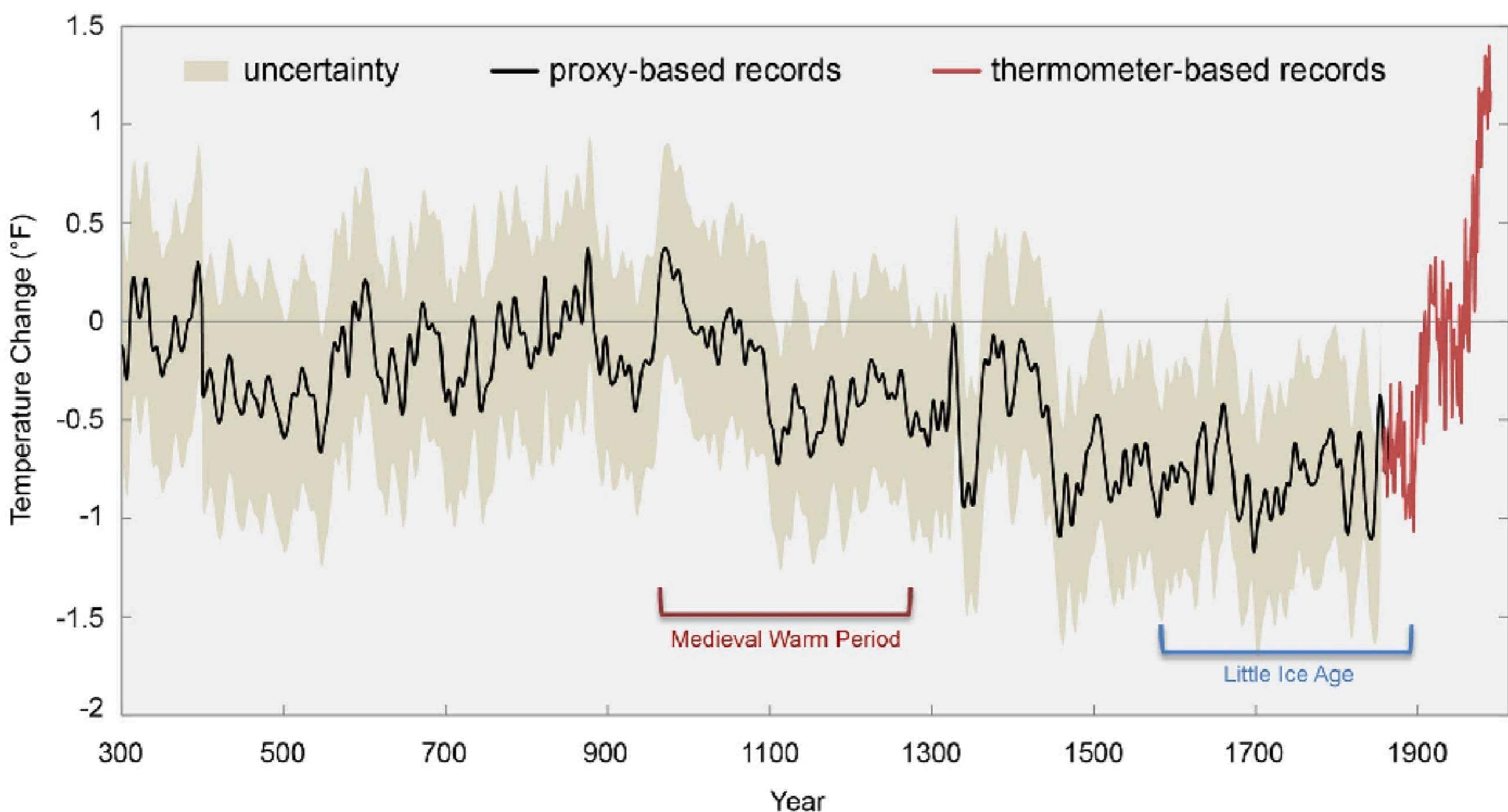
[Community_@Amex](#)

Your questions. Your interests. Your
community.



Amex Mobile available on the [App Store](#)SM or [Google Play](#)TM

1700 Years of Global Temperature Change from Proxy Data





TRUMPING

The Data That Turned the World Upside Down



HG

HANNES GRASSEGGER AND MIKAEL KROGERUS

Jan 28 2017, 9:15am

Psychologist Michal Kosinski developed a method to analyze people in minute detail based on their Facebook activity. Did a similar tool help propel Donald Trump to victory? Two reporters from Zurich-based Das Magazin went data-gathering.

An earlier version of this story appeared in Das Magazin in December.



WikiLeaks

Leaks News About Partners Search



Shop

Donate

Submit

Vault 7: CIA Hacking Tools Revealed



Releases ▾

Documents ▾

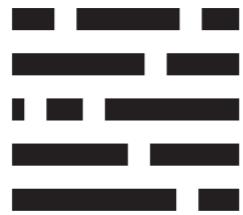
Contents

- Press Release
- Analysis
- Examples

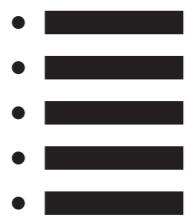
The shape of data

How is your information organized? How do the parts relate to each other?

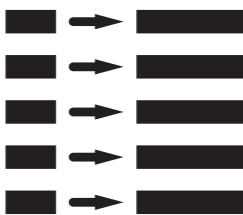
These questions profoundly affect the tools you use and the code you write.



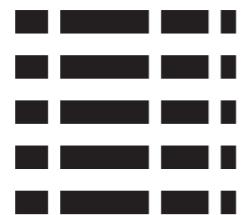
Raw text



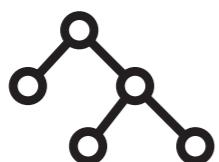
Sequence



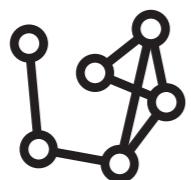
Dictionary



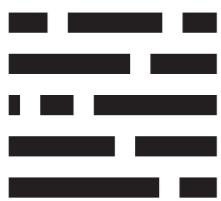
Table



Tree

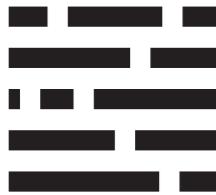


Graph



Raw text

Call me Ishmael. Some years ago—never mind how long precisely —having little or no money in my purse, and nothing particular to interest me on shore, I thought I would sail about a little and see the watery part of the world. It is a way I have of driving off the spleen and regulating the circulation. Whenever I find myself growing grim about the mouth; whenever it is a damp, drizzly November in my soul; whenever I find myself involuntarily pausing before coffin warehouses, and bringing up the rear of every funeral I meet; and especially whenever my hypos get such an upper hand of me, that it requires a strong moral principle to prevent me from deliberately stepping into the street, and methodically knocking people's hats off—then, I account it high time to get to sea as soon as I can. This is my substitute for pistol and ball. With a philosophical flourish Cato throws himself upon his sword; I quietly take to the ship. There is nothing surprising in this. If they but knew it, almost all men in their degree some time



Raw text

TRUMP: Chief Justice Roberts, President Carter, President Clinton, President Bush, President Obama, fellow Americans and people of the world, thank you.

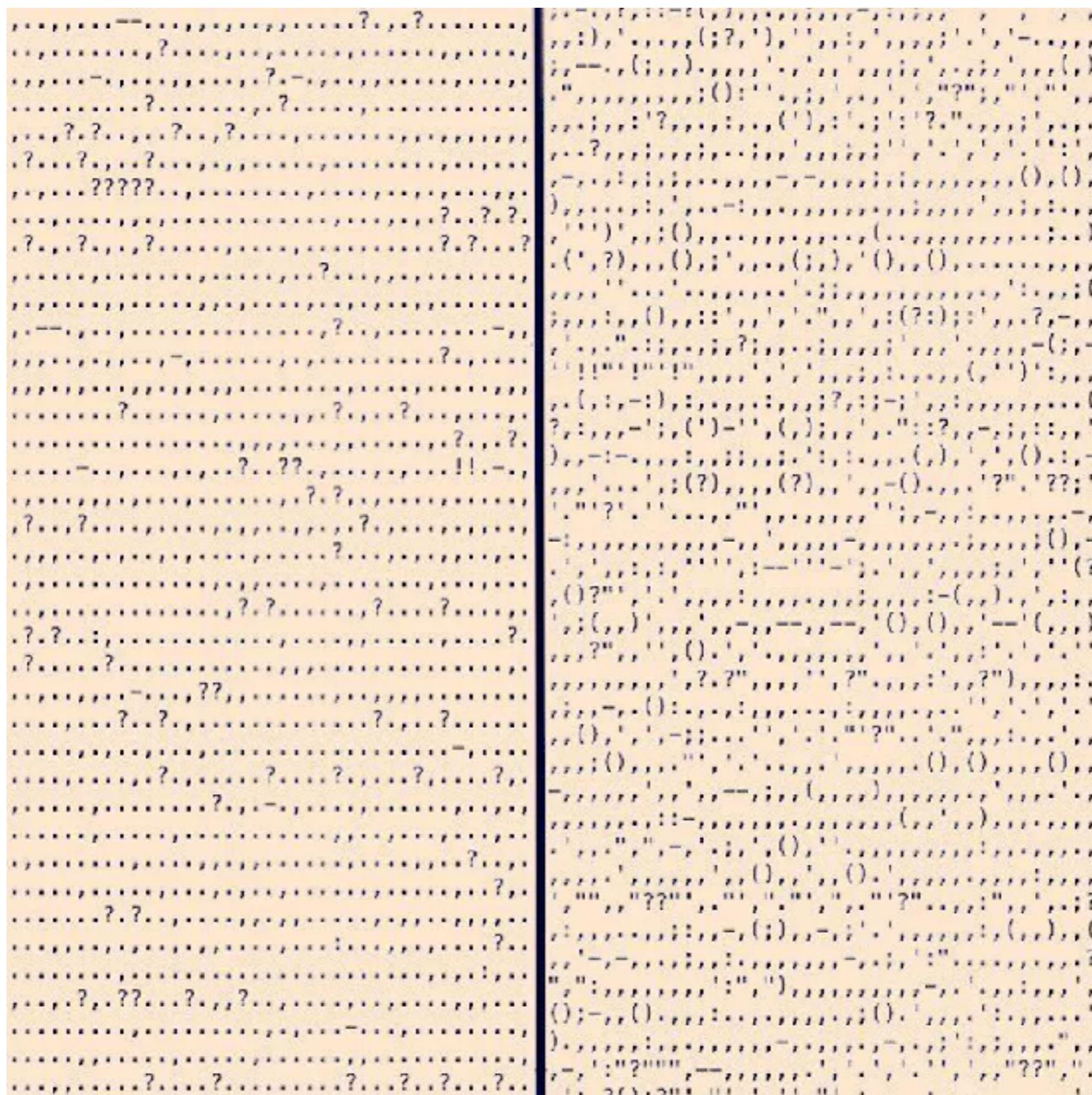
(APPLAUSE) We, the citizens of America, are now joined in a great national effort to rebuild our country and restore its promise for all of our people.

(APPLAUSE)

Together, we will determine the course of America and the world for many, many years to come. We will face challenges, we will confront hardships, but we will get the job done.

Every four years, we gather on these steps to carry out the orderly and peaceful transfer of power, and we are grateful to President Obama and First Lady Michelle Obama for their gracious aid throughout this transition. They have been magnificent. Thank you.

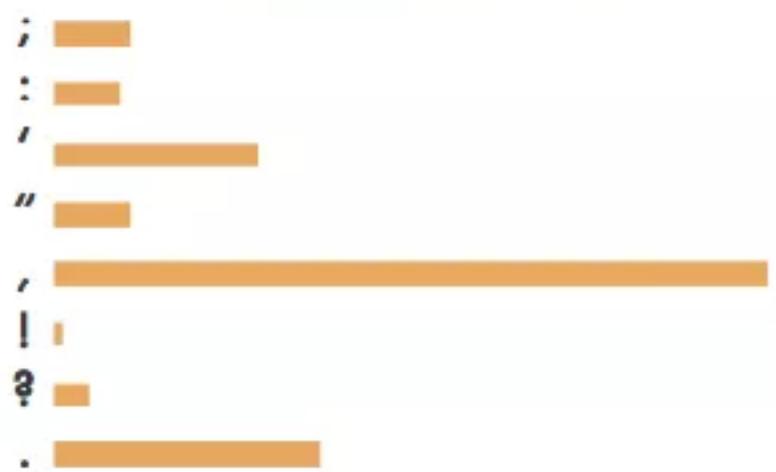
(APPLAUSE)



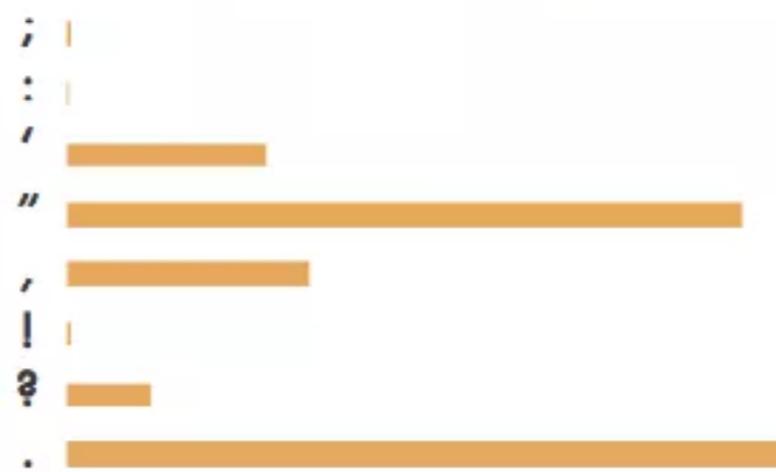
McCarthy

Faulkner

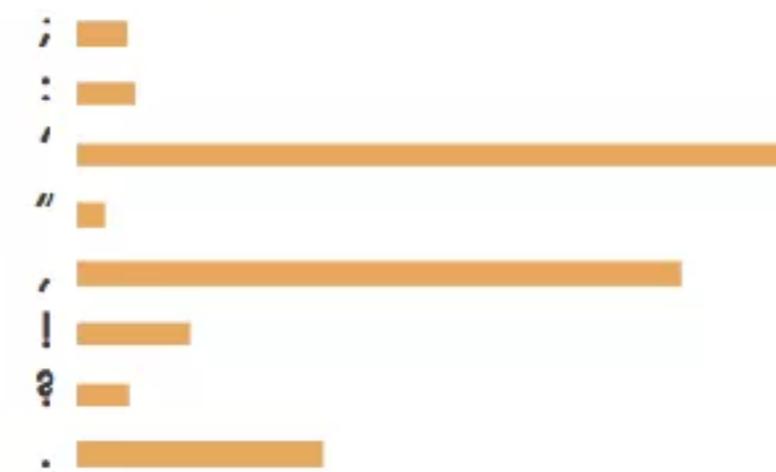
Absalom, Absalom!



A Farewell To Arms



Alice in Wonderland



Blood Meridian



Frankenstein



Great Expectations



Huckleberry Finn



Pride and Prejudice



Ulysses



Received: from CONNMBX02.connect.uwaterloo.ca ([129.97.149.109]) by
connhub1.connect.uwaterloo.ca ([129.97.149.101]) with mapi id 14.03.0319.002;
Tue, 17 Jan 2017 15:57:38 -0500

From: Rishabh Moudgil <rishabh.moudgil@uwaterloo.ca>

To: Craig Kaplan <csk@uwaterloo.ca>

CC: Kevin Harrigan <kevinh@uwaterloo.ca>, Kristina Bayda
<kbayda@uwaterloo.ca>, Travis Bartlett <travis.bartlett@uwaterloo.ca>

Subject: A01 Marking Scheme

Thread-Topic: A01 Marking Scheme

Thread-Index: AdJw/+DUxNKRR1CRRK0Zfc2CQLKSng==

Date: Tue, 17 Jan 2017 20:57:36 +0000

Message-ID: <748888CA42FDF349AF07A8978DDED060281C9EC0@connmbx02>

Accept-Language: en-CA, en-US

Content-Language: en-CA

X-MS-Exchange-Organization-AuthAs: Internal

X-MS-Exchange-Organization-AuthMechanism: 04

X-MS-Exchange-Organization-AuthSource: connhub1.connect.uwaterloo.ca

X-MS-Has-Attach:

X-MS-Exchange-Organization-SCL: -1

X-MS-TNEF-Correlator:

Content-Type: multipart/alternative;
boundary=_000_748888CA42FDF349AF07A8978DDED060281C9EC0connmbx02_

MIME-Version: 1.0

--_000_748888CA42FDF349AF07A8978DDED060281C9EC0connmbx02_

Content-Type: text/plain; charset="Windows-1252"

Content-Transfer-Encoding: quoted-printable

Hi Craig,

Network Working Group
Request for Comments: 2822
Obsoletes: 822
Category: Standards Track

P. Resnick, Editor
QUALCOMM Incorporated
April 2001

Internet Message Format

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

Abstract

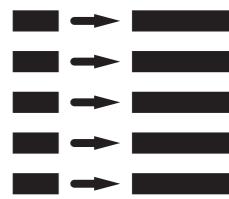
This standard specifies a syntax for text messages that are sent between computer users, within the framework of "electronic mail" messages. This standard supersedes the one specified in Request For Comments (RFC) 822, "Standard for the Format of ARPA Internet Text Messages", updating it to reflect current practice and incorporating incremental changes that were specified in other RFCs.



Sequence

46.12	47.88	46.32	45.27	44.32	43.87	44.23	42.95	41.74	40.69
41.68	40.73	40.75	40.55	39.39	39.27	40.89	41.22	.	40.57
40.43	40.58	39.93	41.08	40.00	37.64	37.46	37.16	36.76	35.65
36.31	37.32	35.55	34.98	34.72	34.55	36.12	36.76	37.62	.
36.36	37.88	36.59	37.13						

The Right Honourable Justin Trudeau
The Right Honourable Stephen Harper
The Right Honourable Paul Edgar Philippe Martin
The Right Honourable Joseph Jacques Jean Chrétien
The Right Honourable A. Kim Campbell
The Right Honourable Martin Brian Mulroney
The Right Honourable John Napier Turner
The Right Honourable Pierre Elliott Trudeau
The Right Honourable Charles Joseph Clark
The Right Honourable Pierre Elliott Trudeau
The Right Honourable Lester Bowles Pearson
The Right Honourable John George Diefenbaker
The Right Honourable Louis Stephen St-Laurent
The Right Honourable William Lyon Mackenzie King
The Right Honourable Richard Bedford Bennett
The Right Honourable William Lyon Mackenzie King
The Right Honourable Arthur Meighen
The Right Honourable William Lyon Mackenzie King



Dictionary

Associate a set of *keys* with a set of *values*. Ask for the value associated with any key without examining every other key/value pair.

1896	Athens, Greece	1968	Mexico City, Mexico
1900	Paris, France	1972	Munich, West Germany
1904	St. Louis, United States	1976	Montréal, Canada
1908	London, United Kingdom	1980	Moscow, Soviet Union
1912	Stockholm, Sweden	1984	Los Angeles, United States
1920	Antwerp, Belgium	1988	Seoul, South Korea
1924	Paris, France	1992	Barcelona, Spain
1928	Amsterdam, Netherlands	1996	Atlanta, United States
1932	Los Angeles, United States	2000	Sydney, Australia
1936	Berlin, Germany	2004	Athens, Greece
1948	London, United Kingdom	2008	Beijing, China
1952	Helsinki, Finland	2012	London, United Kingdom
1956	Melbourne, Australia	2016	Rio de Janeiro, Brazil
1960	Rome, Italy	2020	Tokyo, Japan
1964	Tokyo, Japan		

Discover Weekly



Your weekly mixtape of fresh music. Enjoy new discoveries and deep cuts chosen just for you. Updated every Monday, so save your favourites!

Table

Created by: Spotify • 30 songs, 2 hr 36 min

PAUSE

FOLLOWING

...

FOLLOWER

1

Filter

Download

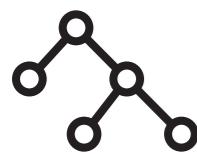


SONG	ARTIST	ALBUM		Download
+ Ways To Go - Margot Mix	Weval, Margot	Weval Remix	11 hours ago	7:11
+ Death Is A Girl	Mini Mansions	The Great Pretend	11 hours ago	4:36
+ Jumbo	Underworld	Beaucoup Fish	11 hours ago	6:58
+ Bug Powder Dust	The Mysterons	Meandering	11 hours ago	4:27
+ ...To Have No Answer	Flock of Dimes	If You See Me, Say	11 hours ago	3:49
+ I'll Cut You Down	Uncle Acid & The	Blood Lust	11 hours ago	5:02
+ L'enfer ce n'est pas les autres c'est moi	The Eye Of Time	Myth I: A Last Da	11 hours ago	5:46
+ Terrain	pg.lost	Key	11 hours ago	5:29



2:46

3:54



Tree

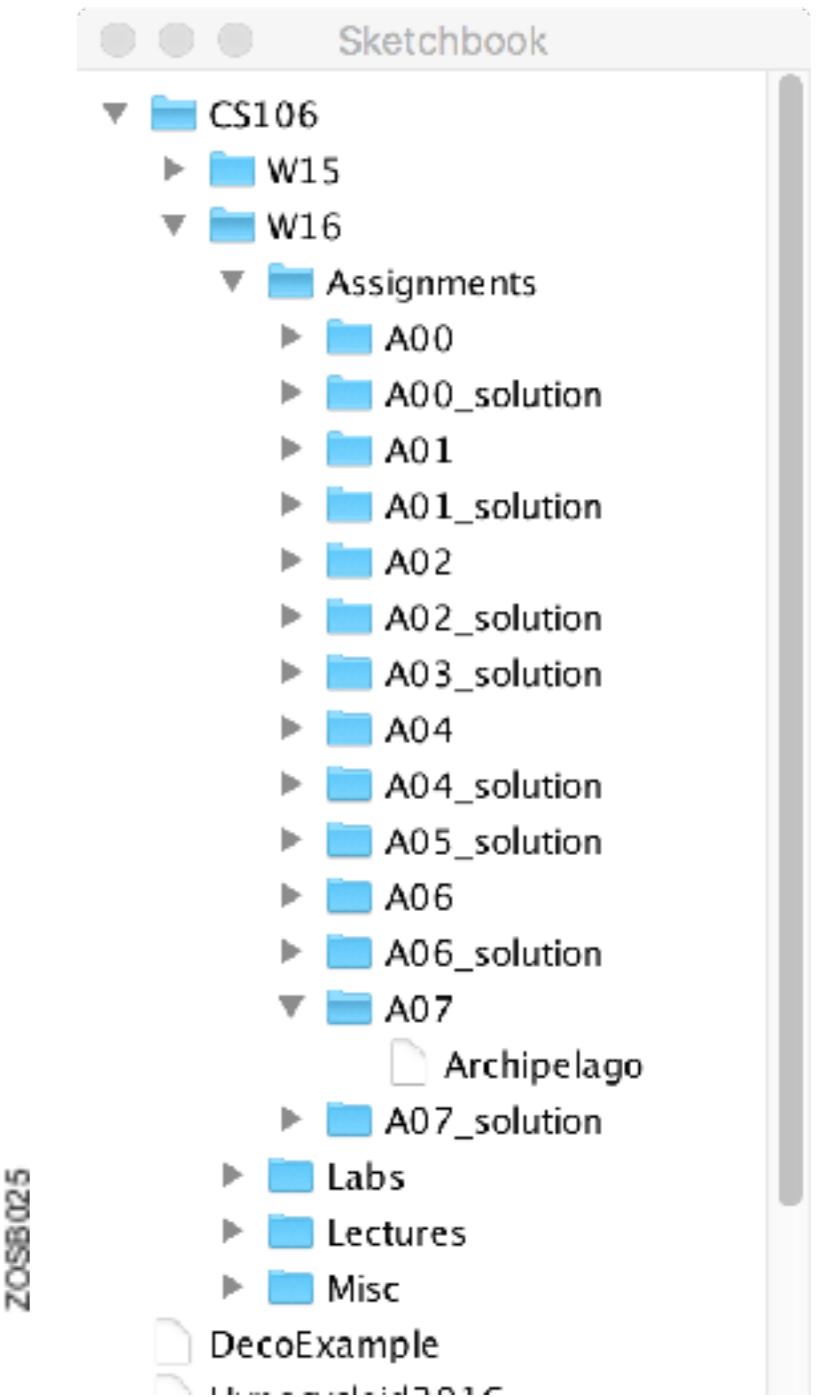
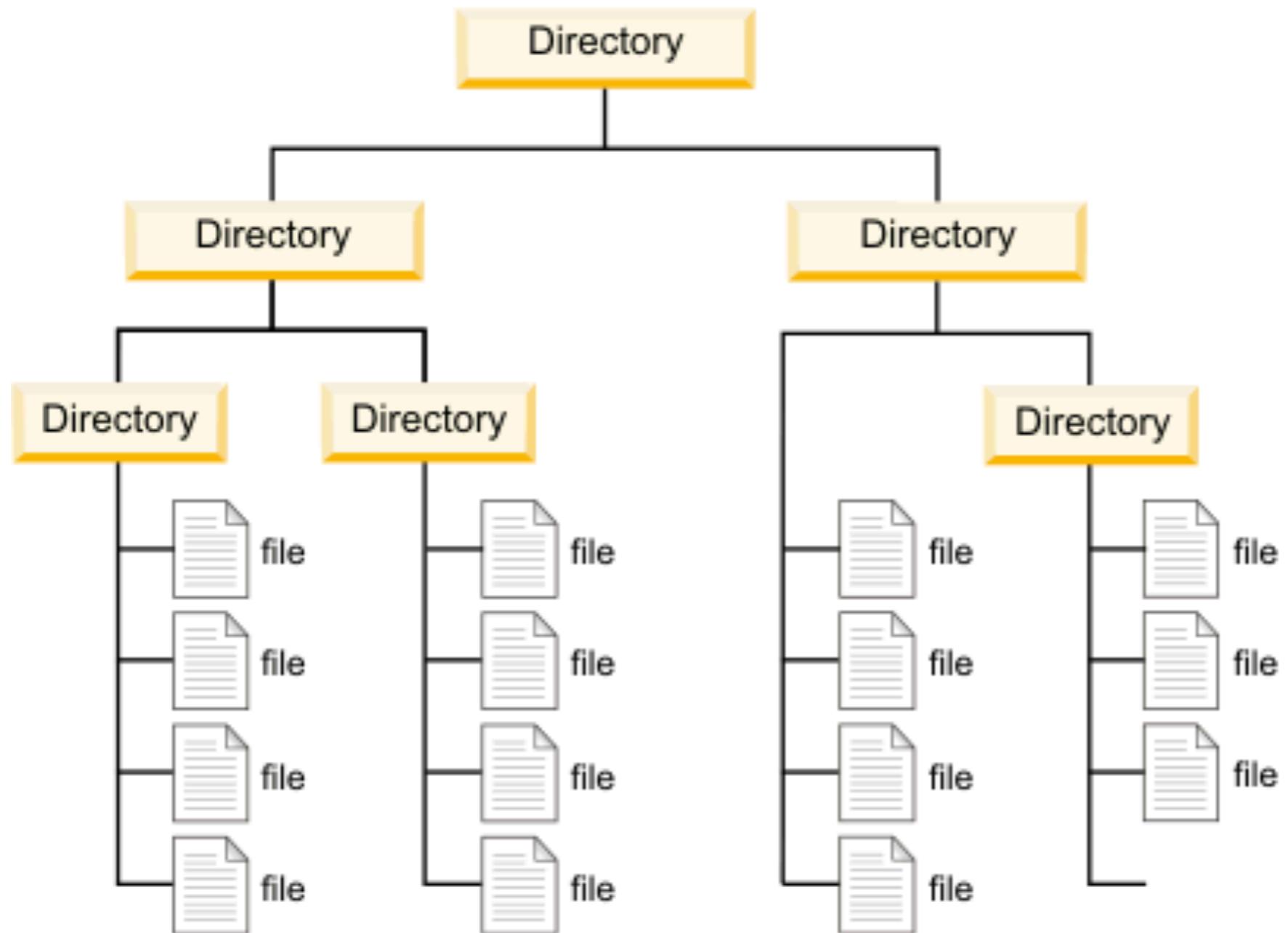


Image
save()
saveFrame()

Files
beginRaw()
beginRecord()
createOutput()
createWriter()
endRaw()
endRecord()
PrintWriter
saveBytes()
saveJSONArray()
saveJSONObject()
saveStream()

a.ref-link | 204x18

saveTable()
saveXML()
selectOutput()

Transform

applyMatrix()
popMatrix()
printMatrix()
pushMatrix()
resetMatrix()
rotate()
rotateX()
rotateY()
rotateZ()
scale()
shearX()
shearY()
translate()

Calculation
abs()
ceil()
constrain()
dist()
exp()
floor()
lerp()
log()
mag()
map()
max()
min()
norm()
pow()
round()
sq()
sqrt()

Trigonometry

acos()
asin()
atan()
atan2()
cos()
degrees()
radians()
sin()
tan()

Random

noise()
noiseDetail()
noiseSeed()
random()
randomGaussian()
randomSeed()

Elements Console Sources Network Timeline > ▲ 1 : X

```
<a href="beginRaw.html" class="ref-link">beginRaw()</a>
<a href="beginRecord.html" class="ref-link">beginRecord()</a>
<a href="createOutput.html" class="ref-link">createOutput()</a>
<a href="createWriter.html" class="ref-link">createWriter()</a>
<a href="endRaw.html" class="ref-link">endRaw()</a>
<a href="endRecord.html" class="ref-link">endRecord()</a>
<a href="PrintWriter.html" class="ref-link">PrintWriter</a>
<a href="saveBytes.html" class="ref-link">saveBytes()</a>
<a href="saveJSONArray.html" class="ref-link">saveJSONArray()</a>
<a href="saveJSONObject.html" class="ref-link">saveJSONObject()</a>
<a href="saveStream.html" class="ref-link">saveStream()</a>
<a href="saveStrings.html" class="ref-link">saveStrings()</a>
<a href="saveTable.html" class="ref-link">saveTable()</a>
<a href="saveXML.html" class="ref-link">saveXML()</a>
<a href="selectOutput.html" class="ref-link" href="https://processing.org/reference/saveXML_.html" ut()</a>
</div>
▶<div class="category">...</div>
▶<div class="category">...</div>
</div>
▶<div class="ref-col">...</div>
</div>
<!-- ===== FOOTER
```

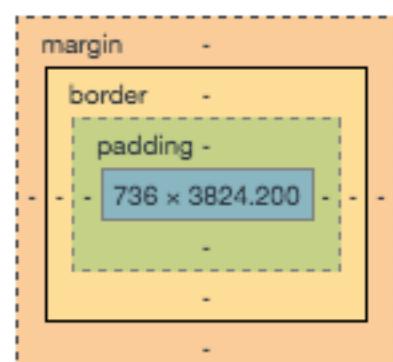
html.js.no-touch body#language

Styles Event Listeners DOM Breakpoints Properties

Filter :hov .cls +

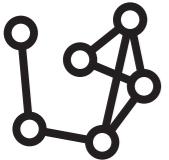
```
element.style { }
body {
    margin: 0;
    padding: 0;
    overflow-y: scroll;
    background-color: #ddd;
    font-family: 'theSerif', 'Enriqueta',
        georgia, times, serif;
    -webkit-font-smoothing: antialiased;
    -webkit-text-size-adjust: none;
    font-size: 100%;
    font-size: 0.79em;
    font-weight: normal;
    line-height: 1.5em;
    color: #252525;
}
```

body { user agent stylesheet }

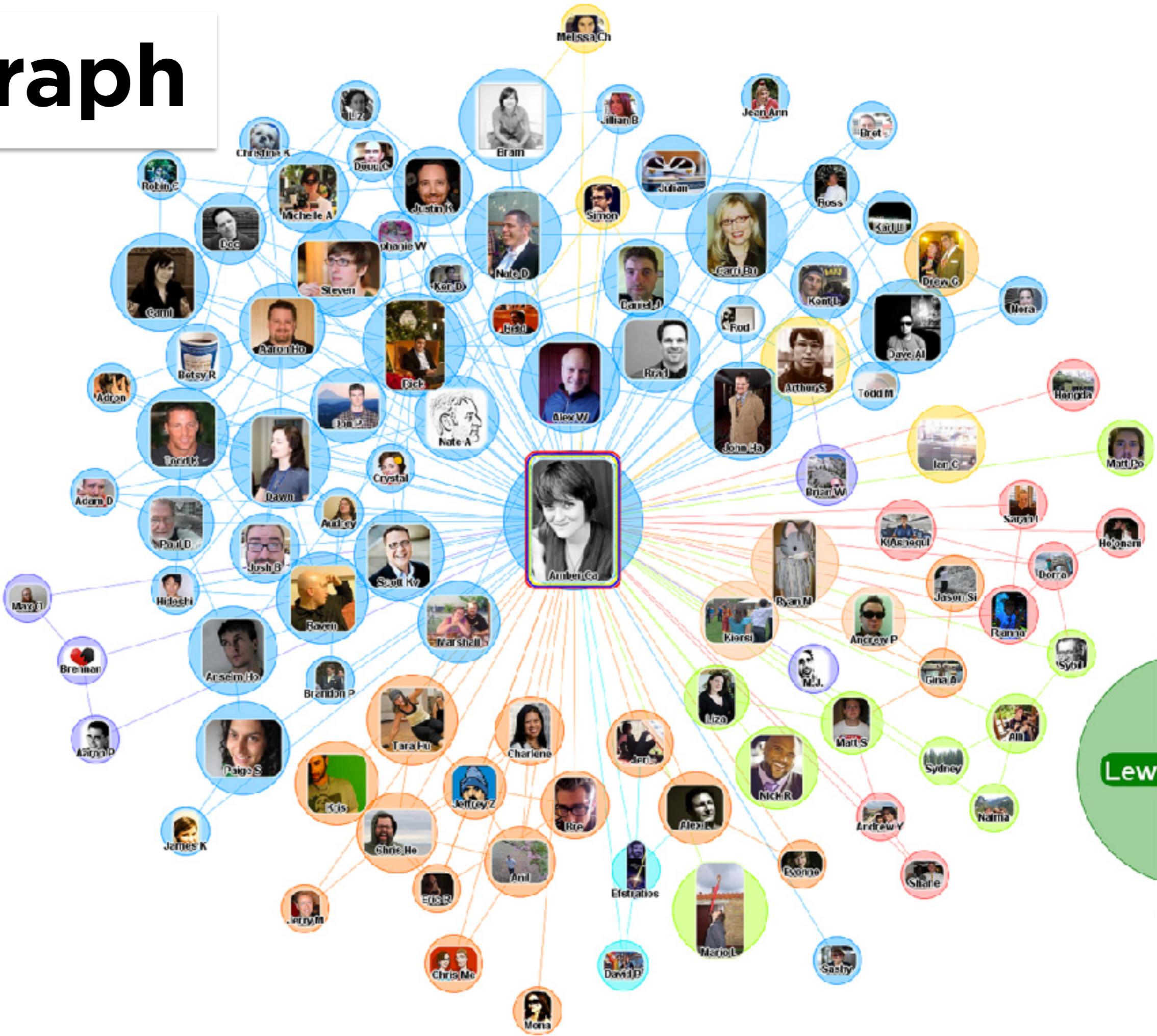


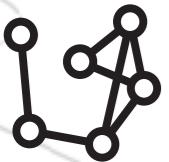
Filter □ Show all

► background-color... □ rgb(2...
► color ■ rgb(3...
► display block
► font-family theSer...
► font-size 12.64px

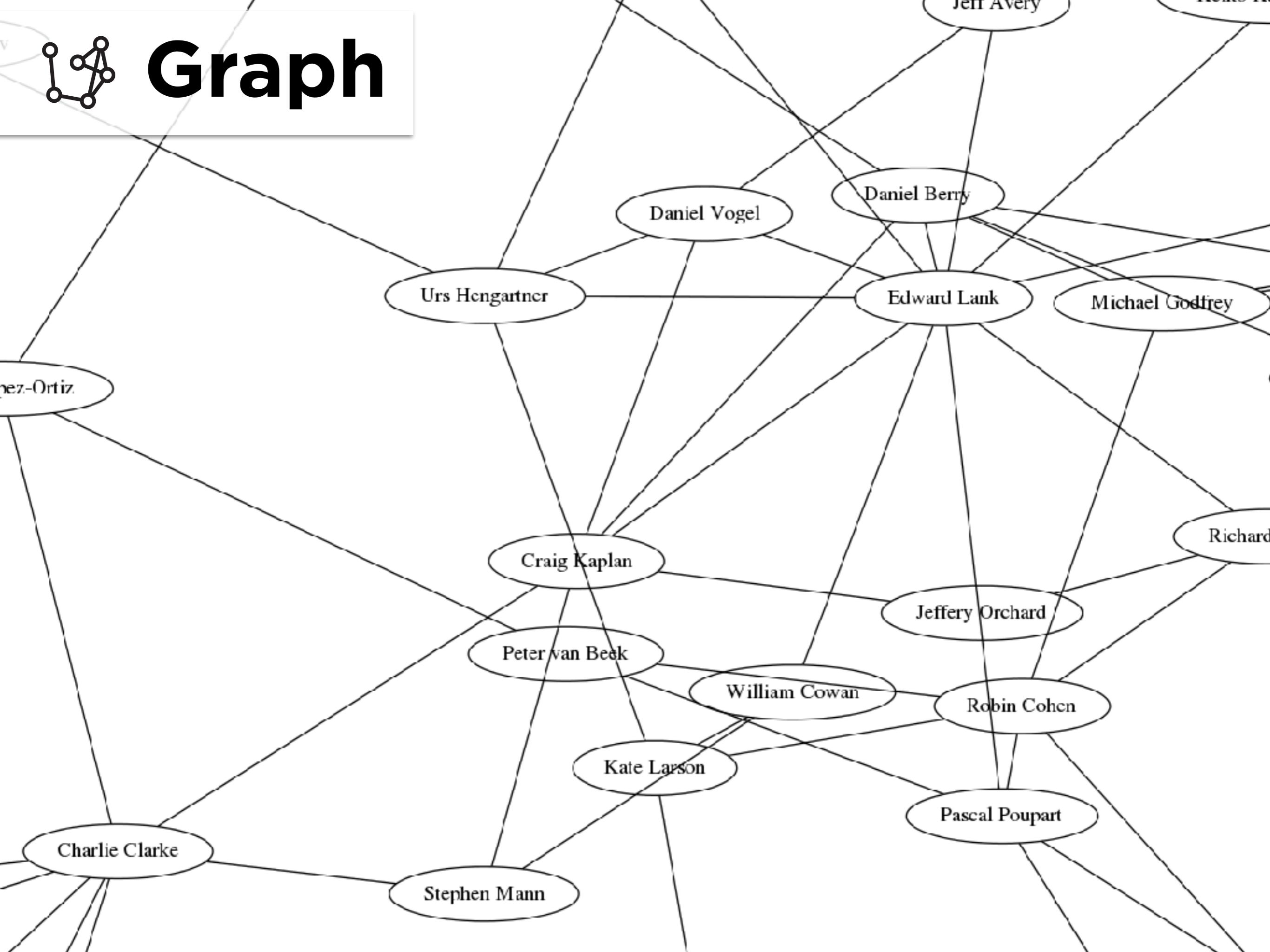


Graph





Graph



String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```

String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```

Initialize a variable
from a string literal

String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```

Count the number of
characters in a string

String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```

Extract a character from a string. Like accessing an array

String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```

Glue two strings
together

String operations

```
String wd = "...";  
int len = wd.length();  
char c = wd.charAt(2);  
  
String str3 = str1 + str2;  
if( str1.equals( str2 ) ) { ... }  
  
String[] words = splitTokens( str1 );
```

Check if two strings have
the same characters

String operations

```
String wd = "...";
```

```
int len = wd.length();
```

```
char c = wd.charAt(2);
```

```
String str3 = str1 + str2;
```

```
if( str1.equals( str2 ) ) { ... }
```

```
String[] words = splitTokens( str1 );
```



Break a string into words
by looking for whitespace

Messier text

```
String[] splitTokens( String text, String delims ) { ... }
```

Break the long string text into “words”, where the characters in delims (and not whitespace) are treated as breakpoints.

```
String trim( String text ) { ... }
```

Return a copy of text with any excess whitespace removed from the start and end.

Example: the Region of Waterloo's list of reserved street names

FullStreetName	Municipality
Abbey Glen	Kitchener
Aberle	Woolwich
Abeth	Kitchener
Abitibi	Cambridge
Able	Cambridge
Abram Clemens St	Kitchener
Accobee	Cambridge
Adair	Cambridge
Adcock	Region of Waterloo
Addlev	Cambridge

Reading the dictionary

A

a

aa

aal

aalii

aam

Aani

aardvark

aardwolf

Aaron

Aaronic

Aaronical

Aaronite

Aaronitic

Aaru

Ab

aba

Ababdeh

Ababua

abac

abaca

abacate

abacay

abacinate

abacination

abaciscus

Reading the dictionary

A
a
aa
aal
aalii
aam
Aani
aardvark
aardwolf
Aaron
Aaronic
Aaronical
Aaronite
Aaronitic
Aaru
Ab
aba
Ababdeh
Ababua
abac
abaca
abacate
abacay
abacinate
abacination
abaciscus

Find the longest word
Find all words with three or more Ys
Find all words ending with MT
Find all words starting with TM
Find all words ending with DOUS
Find all words containing UFA
Find all words ending in GRY
Find all palindromes
Find words with three consecutive double letters
Find the longest word whose letters are in alphabetical order
Find the longest word with no vowels

Finding things in strings

```
if( str.contains( "abc" ) ) { ... }
```

Check if the string str has the substring “abc” anywhere inside of it.

```
if( str.startsWith( "def" ) ) { ... }  
if( str.endsWith( "ghi" ) ) { ... }
```

Look for a substring specifically at the start or end of a string.

Writing a spellchecker

With the dictionary at our disposal, it's easy to check if a given string is a word.

```
String[] dict;
```

```
void setup() {  
    dict = loadStrings( "words.txt" );  
}
```

```
boolean isWord( String word ) {
```

```
}
```

Writing a spellchecker

With the dictionary at our disposal, it's easy to check if a given string is a word.

```
String[] dict;  
  
void setup() {  
    dict = loadStrings( "words.txt" );  
}  
  
boolean isWord( String word ) {  
    for ( int idx = 0; idx < dict.length; ++idx ) {  
        if ( dict[idx].equals( word ) ) {  
            return true;  
        }  
    }  
    return false;  
}
```

```
boolean isWord( String word ) {  
    for ( int idx = 0; idx < dict.length; ++idx ) {  
        if ( dict[idx].equals( word ) ) {  
            return true;  
        }  
    }  
    return false;  
}
```

Looping over every word works, but it's painfully slow, especially when the word isn't there!

Dictionaries

In programming, a *dictionary* is a mapping from a set of *keys* to a set of *values*. Any given key may have at most one associated value.

Year → Olympic host city

Name → Phone number

Student ID number → Exam seating code

Clicker ID → Student ID number

Server name → IP address

Dictionaries

Dictionary operations we might care about:

- Look up the value associated with a given key
- Ask if the dictionary has a given key
- Add a new key to the dictionary, with its associated value
- Remove a key and its value from the dictionary

Processing includes a few handy dictionary classes, where the keys are Strings:

- IntDict: **map Strings to ints**
- FloatDict: **map Strings to floats**
- StringDict: **map Strings to Strings**

Processing includes a few handy dictionary classes, where the keys are Strings:

- IntDict: map Strings to ints
- FloatDict: map Strings to floats
- StringDict: map Strings to Strings

Java allows more-or-less arbitrary mappings between keys and values:

- `java.util.TreeMap<K, V>`: map any key type K to any value type V.

```
IntDict myDict = new IntDict();
```

Create a new, empty dictionary

```
IntDict myDict = new IntDict();
```

Create a new, empty dictionary

```
myDict.set( "Kumquat", 13 )
```

```
myDict.set( "Durian", 19 );
```

**Add a new key to the dictionary, with
its associated value**

```
IntDict myDict = new IntDict();  
Create a new, empty dictionary
```



```
myDict.set( "Kumquat", 13 )  
myDict.set( "Durian", 19 );  
Add a new key to the dictionary, with  
its associated value
```



```
println( myDict.get( "Kumquat" ) );  
Look up the value associated with a  
given key
```

```
IntDict myDict = new IntDict();  
Create a new, empty dictionary
```



```
myDict.set( "Kumquat", 13 )  
myDict.set( "Durian", 19 );  
Add a new key to the dictionary, with  
its associated value
```



```
println( myDict.get( "Kumquat" ) );  
Look up the value associated with a  
given key
```



```
if( myDict.hasKey( "Rambutan" ) ) { ... }  
Ask if the dictionary has a given key
```

```
IntDict myDict = new IntDict();  
Create a new, empty dictionary
```



```
myDict.set( "Kumquat", 13 )  
myDict.set( "Durian", 19 );  
Add a new key to the dictionary, with  
its associated value
```



```
println( myDict.get( "Kumquat" ) );  
Look up the value associated with a  
given key
```



```
if( myDict.hasKey( "Rambutan" ) ) { ... }  
Ask if the dictionary has a given key
```



```
myDict.remove( "Durian" );  
Remove a key and its value from the  
dictionary
```

Writing a spellchecker

```
String[] dict;  
  
void setup() {  
    dict = loadStrings( "words.txt" );  
}  
  
boolean isWord( String word ) {  
    for ( int idx = 0; idx < dict.length; ++idx ) {  
        if ( dict[idx].equals( word ) ) {  
            return true;  
        }  
    }  
    return false;  
}
```

Writing a spellchecker

```
IntDict myDict;

void setup()
{
    String[] words = loadStrings("words.txt");
    for( int idx = 0; idx < words.length; ++idx ) {
        myDict.set( words[idx], 1 );
    }
}

boolean isWord( String word )
{
    return myDict.containsKey( word );
}
```

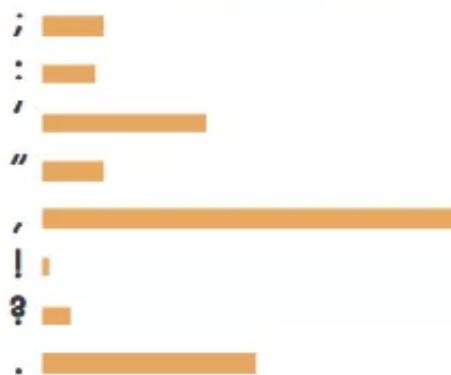
Writing a spellchecker

```
IntDict myDict;  
  
void setup()  
{  
    String[] words =  
    for( int idx = 0; idx < words.length; ++idx ) {  
        myDict.set( words[idx], 1 );  
    }  
}  
  
boolean isWord( String word )  
{  
    return myDict.containsKey( word );  
}
```

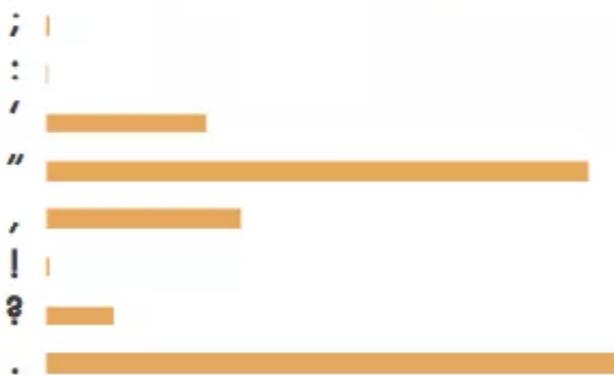
These are guaranteed
to be fast!

Counting things

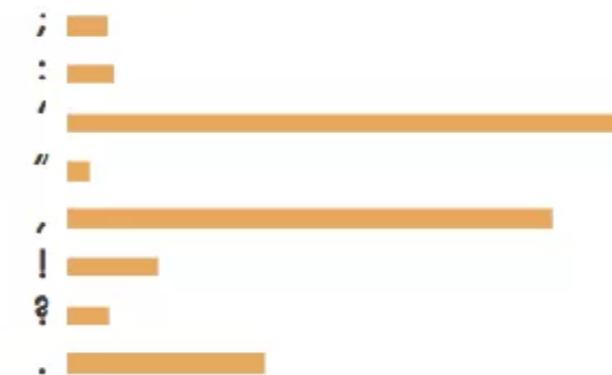
Absalom, Absalom!



A Farewell To Arms



Alice in Wonderland



Blood Meridian



Frankenstein



Great Expectations



Huckleberry Finn



Pride and Prejudice



Ulysses



Finding patterns

It's easy to search a string for a given phone number:

```
if( myString.contains( "(519) 888-4567" ) ) { ... }
```

But what if we wanted to find all the phone numbers in a string?

Finding patterns

Regular Expressions are a general tool for finding patterns in strings.

Finding patterns

Regular Expressions are a programming language for finding patterns in strings.

Finding patterns

Regular Expressions are a **cryptic** programming language for finding patterns in strings.



```
String[] match( String text, String pattern ) { ... }
```

**Look for an instance of the regular expression
pattern inside of the string text. If the answer
is not null, the pattern was found.**

Regular Expressions - Quick Reference Guide



Anchors	Literal Characters	Character group contents
\$^ start of line \$ end of line \\b word boundary \\B not at word boundary \\A start of subject \\G first match in subject \\z end of subject \\Z end of subject \\z or before newline at end	Letters and digits match exactly Some special characters match exactly Escape other specials with backslash	a x B 7 0 @ - = % \\ \\ \$ \\ [
Non-printing characters	Character Groups	Examples
\a alarm (BEL, hex 07) \cx "control-x" \e escape (hex 1B) \f formfeed (hex 0C) \n newline (hex 0A) \r carriage return (hex 0D) \t tab (hex 09) \ddd octal code ddd \hh hex code hh \{hhh..} hex code hhh..	Almost any character (usually not newline) Lists and ranges of characters Any character except those listed	[a-zA-Z0-9] [:alnum:]
Generic character types	Counts (add ? for non-greedy)	Comments (?#comment)
\d decimal digit \D not a decimal digit \s whitespace character \S not a whitespace char \w "word" character \W "non-word" character	0 or more ("perhaps some") 0 or 1 ("perhaps a") 1 or more ("some") Between "n" and "m" of Exactly "n", "n" or more	*
POSIX character classes	Alternation	?
alnum letters and digits alpha letters ascii character codes 0-127 blank space or tab only cntrl control characters digit decimal digits graph printing chars -space lower lower case letters print printing chars +space punct printing chars -alnum space white space upper upper case letters word "word" characters xdigit hexadecimal digits	Either/or	+
		{n,m}
		{n}, {n,}
	Lookahead and Lookbehind	
	Followed by NOT followed by Following NOT following	(?=) (?!) (?<=) (?<!)
	Grouping	
	For capture and counts Non-capturing Named captures	() (?:) (?<name>)
	Back references	
	Numbered Relative Named	\n \gn \g{n} \g{-n} \k<name>
	Conditional insertions	
	(?n:insertion) (?n:insertion:otherwise)	

Substring “ufa” anywhere in a word:

ufa

Word ending in “mt”:

mt\$

Word with three or more “y”s, on a line by itself:

y.*y.*y

An integer:

^(-?[1-9]+\d*)\$ | ^0\$

An email address:

\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.\[A-Z\]{2,}\b

A URL:

^(https?:\/\/)?([\da-z\.-]+)\.([a-z\.]{2,6})([\/\w\.-]*)*\?\$/