

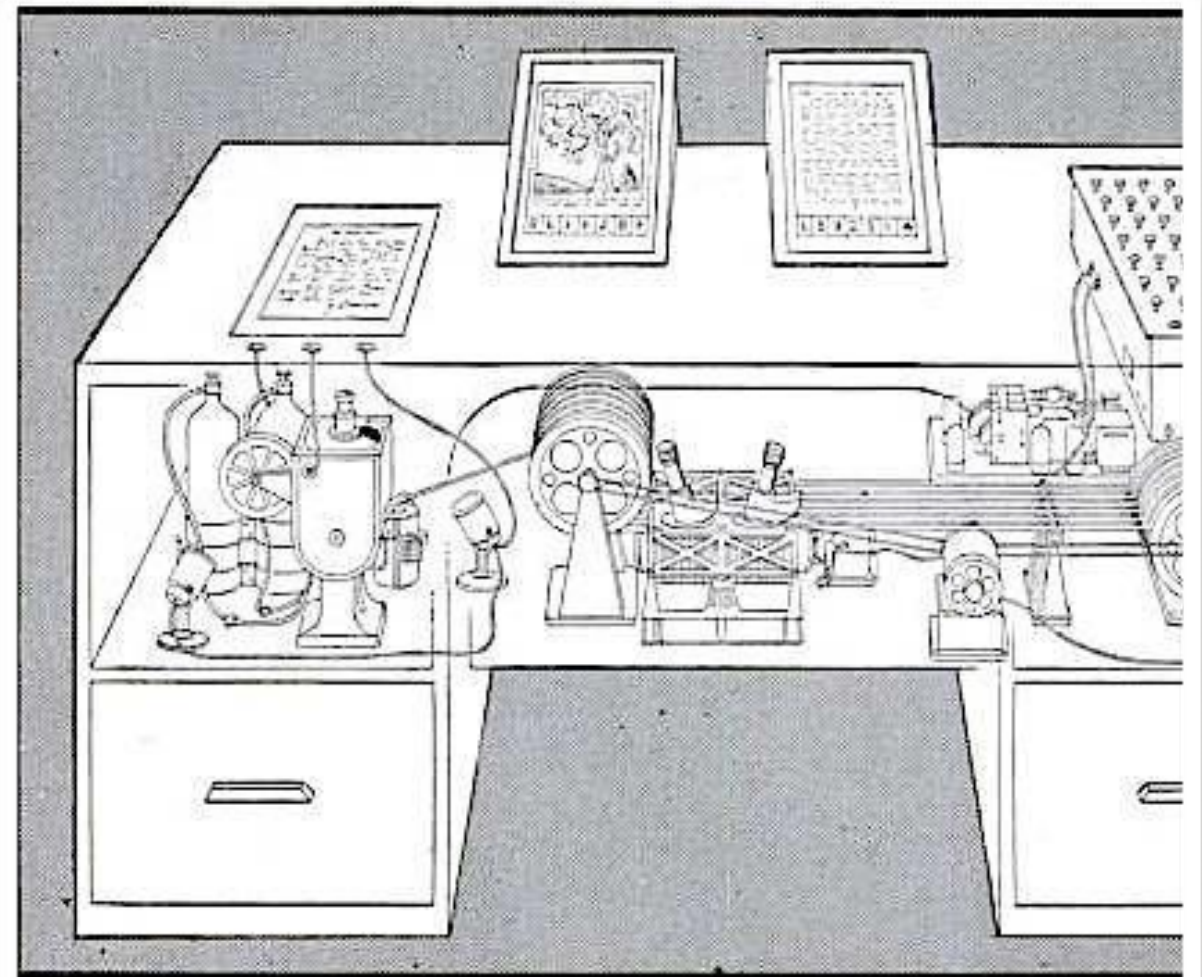
Hypertext Transfer Protocol

One Protocol to Rule Them All

Memex

“Wholly new forms of encyclopedias will appear, ready-made with a mesh of associative trails running through them, ready to be dropped into the memex and there amplified.”

—*Vannevar Bush, 1945*



MEMEX in the form of a desk would instantly bring files and material to the operator's fingertips. Slanting translucent viewing screens magnify film filed by code numbers. At left is a mechanism which automatically longhand notes, pictures and letters, then files them in the desk for future use.

AS WE MAY THINK CONTINUED



monday afternoon

december 9

3:45 p.m. / arena

Chairman:

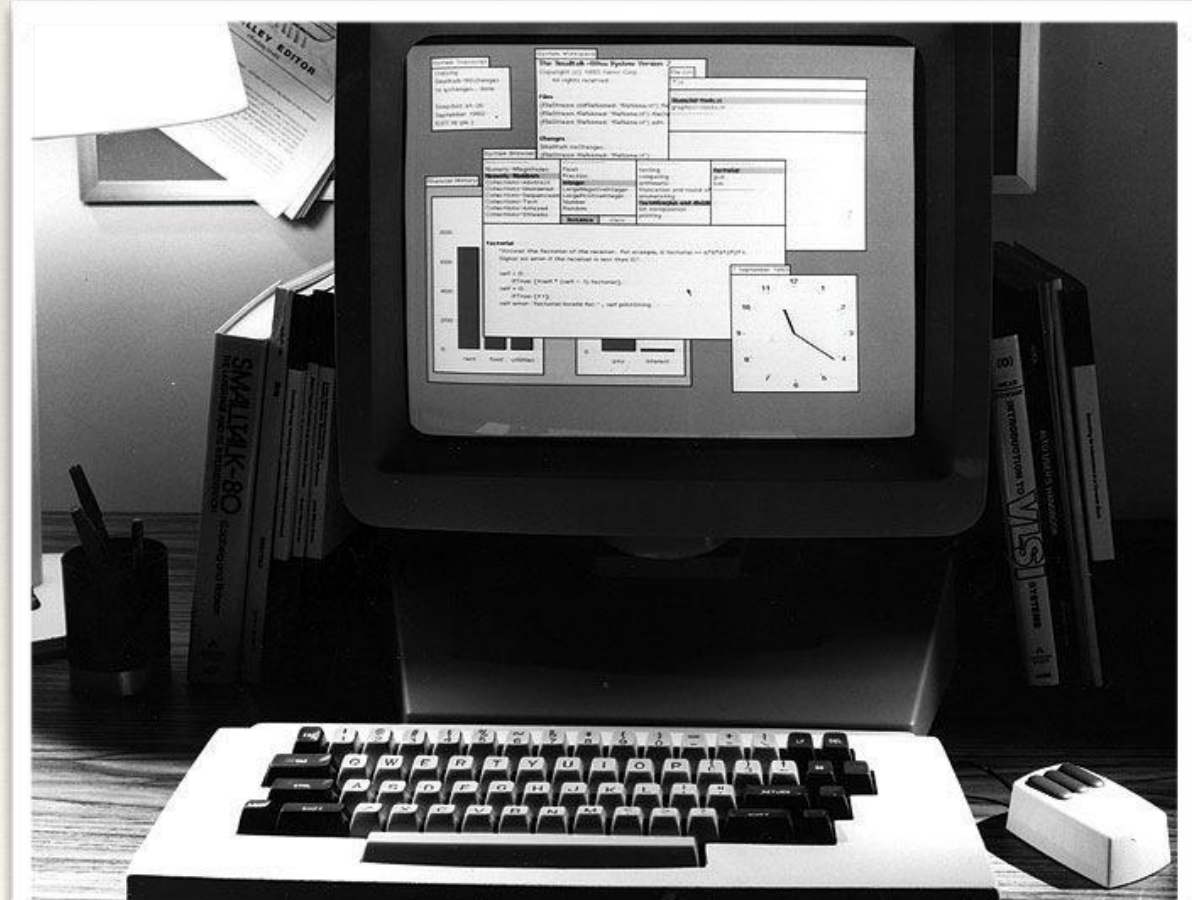
DR. D. C. ENGELBART

*Stanford Research Institute
Menlo Park, California*

a research center for augmenting human intellect

This session is entirely devoted to a presentation by Engelbart on a computer-based, interactive, multicon display system which is being developed at Stanford search Institute under the sponsorship of ARPA, NASA RADC. The system is being used as an experimental oratory for investigating principles by which interactive computer aids can augment intellectual capability. techniques which are being described will, themselves be used to augment the presentation.

The session will use an on-line, closed circuit television hook-up to the SRI computing system in Menlo Park. Following the presentation remote terminals to the system in operation, may be viewed during the remainder of conference in a special room set aside for that purpose.



gopher

USENET

etc...

ftp

finger

etc...

archie

veronica

wais

etc...

World Wide Web

The V
unive

Every
summ

What

Help

Softw

Tech

Bibli

People

History

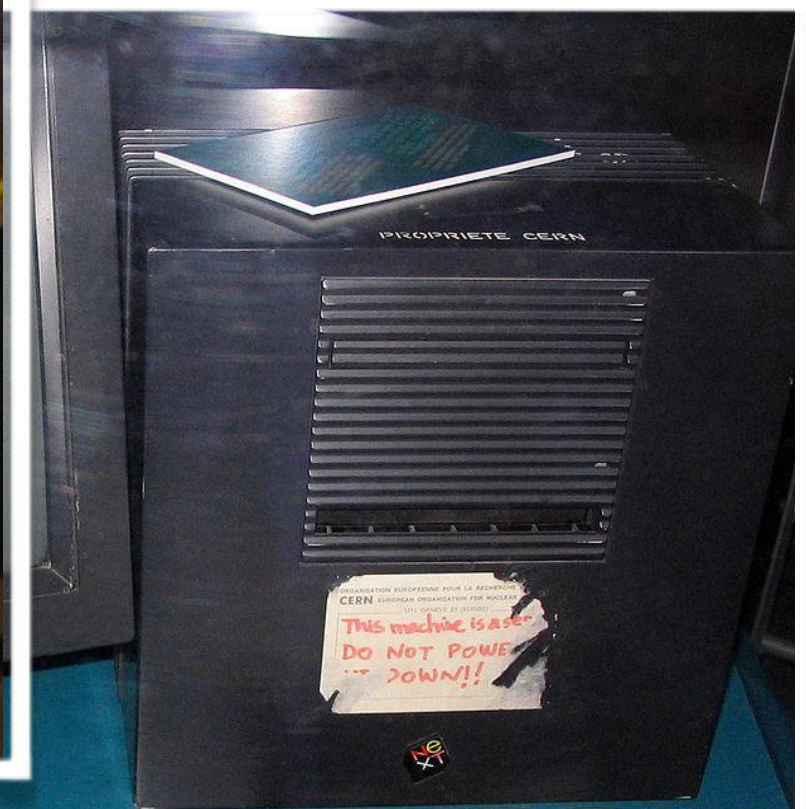
How can I help ?

Getting code

[http://TheProject.html](#)

and initiative aiming to give

document, including an [executive](#)
[Frequently Asked Questions](#).



Paper documentation on W3 and

A list of some people involved in

A summary of the history of the

If you would like to support the web..



How, Who, What

`http://clients.gizmocracy.com/CS260/index.html`

`text/html`


```
Jaegermac-2:~ stebee$ telnet web108.webfaction.com 80
Trying 199.58.84.82...
Connected to web108.webfaction.com.
Escape character is '^]'.
GET http://clients.gizmocracy.com/CS260/minimum.html
<html>
<head>
<title>Extremely Minimal Web Page</title>
</head>
<body>
This web page won't validate according to the HTML5 rules, but
browsers are forgiving and render it just fine anyway.</body>
</html>
Connection closed by foreign host.
Jaegermac-2:~ stebee$
```



```
GET /CS260/minimum.html HTTP/1.1
```

```
Host: clients.gizmocracy.com
```

```
HTTP/1.1 200 OK
```

```
Server: nginx
```

```
Date: Tue, 18 Nov 2014 22:59:09 GMT
```

```
Content-Type: text/html
```

```
Content-Length: 204
```

```
Connection: keep-alive
```

```
Vary: Accept-Encoding
```

```
Last-Modified: Tue, 18 Nov 2014 20:38:12 GMT
```

```
ETag: "7c1007a-cc-5082812e0f500"
```

```
Accept-Ranges: bytes
```

```
<html>
```

```
<head>
```

```
<title>Extremely Minimal Web Page</title>
```

```
</head>
```

```
<body>
```

```
This web page won't validate according to the HTML5 rules, but  
browsers are forgiving and render it just fine anyway.</body>
```

Response Codes

- 1xx: Informational
- 2xx: Success
 - 200: OK
- 3xx: Redirection
 - 301 & 302: Moved (permanently or temporarily)
- 4xx: Bad Request
 - 404: Not found
 - 401 & 403: Unauthorized or forbidden
- 5xx: Server Error

HTTP Verbs

- GET
- HEAD: “If I were to GET this, what would the headers be?”
- POST: Form submission
- PUT, DELETE: Sometimes used.
- TRACE, OPTIONS: Nah.

Jaegermac-2:~ stebee\$ telnet web108.webfaction.com 80

Trying 199.58.84.82...

Connected to web108.webfaction.com.

Escape character is '^]'.
POST /CS260/echo_post.cgi HTTP/1.1

Host: clients.gizmocracy.com

Content-Type: application/x-www-form-urlencoded

Content-Length: 7

foo=bar

foo=bar

HTTP/1.1 200 OK

Server: nginx

Date: Tue, 18 Nov 2014 23:55:05 GMT

Content-Type: text/plain

Transfer-Encoding: chunked

Connection: keep-alive

Vary: Accept-Encoding

8

foo=bar

2

0

URL Encoding

- *Must* escape the following:
 - Space and below
 - Delimiters: < > # % "
 - Reserved: ; / ? : @ & = + \$,
 - Everything beyond 7-bit ASCII
- *Should* escape { } | \ ^ [] `
- Spaces become +, all others become %**xx** (e.g. @ = %**40**)
- Start with RFC1738, then RFC3986

Headers

- Content
- Cookies
- Caching
- Authentication & Authorization

Content Headers

- Content-Type: Body posts don't have to be `application/x-www-form-urlencoded`
 - Web services often use `application/json`
- Accept: On a request, specify the desired response type
 - Some web services use the file extension of the URL
 - Can specify multiple formats, ranked by desirability
- Accept-Encoding: Used to support compressed responses

Cookie Headers

- Cookies are a mechanism to maintain state across HTTP requests
 - `Set-Cookie: name=value; Expires=Wed, 18 Nov 2014 06:54:10 GMT`
 - `Cookie: name=value; name2=value2`
- Can set Path and Domain attributes to restrict where cookie gets submitted to
- Secure and HttpOnly attributes control submission over HTTPS
- Don't use cookies

Caching

- Lots of systems in place to provide time-based client-side content caching; not good for dynamic content
- **ETag:** "686897696a7c876b7e"
- **If-None-Match:** "686897696a7c876b7e"
 - 304 response if that's the current version
 - Normal response otherwise

Authentication & Authorization

- Authentication is proving who you are
- Authorization is verifying what you are allowed to do
- “HTTP Basic Authentication”
 - `WWW-Authenticate: Basic realm="gizmocrazy.com"`
 - `Authorization: Basic QWxhZGRpbjpvcmVudHJlcn2FtZQ==`
- Must use HTTPS at all times!
- Other schemes are better

It's Dangerous to Go Alone

- Having just showed you how to write your own HTTP client code: Don't.
 - Lots of complex details to get right
 - Security implications
- Most environments provide you an HTTP library; use that.
- In C++, I suggest libcurl, but there are alternatives.

Inside Browsers

- Code running inside a browser (HTML5, WebGL, Flash or Unity) has some restrictions on the kind of web requests it can make
 - GET and POST only
 - Limited/no support for modifying headers
- Will usually need a crossdomain.xml file

Crossdomain.xml

- Looked for at the root of the server you're targeting
 - Before you request
`http://clients.gizmocracy.com/CS260/index.html`,
the plugin looks for
`http://clients.gizmocracy.com/crossdomain.xml`

```
<?xml version="1.0"?>
<cross-domain-policy>
    <allow-access-from domain="*" />
</cross-domain-policy>
```

HTTP

- Invented to serve documents that contained links to any resource type; later extended to serve any resource type directly
- Thanks to its versatility, has effectively become its own transport layer
- Use a client library, don't write your own socket code
- Every web service has different requirements, but you'll use the same set of tools:
 - Request verbs (GET, POST, etc.)
 - URL-encoded querystrings and form bodies
 - Other POSTed document types
 - Headers

