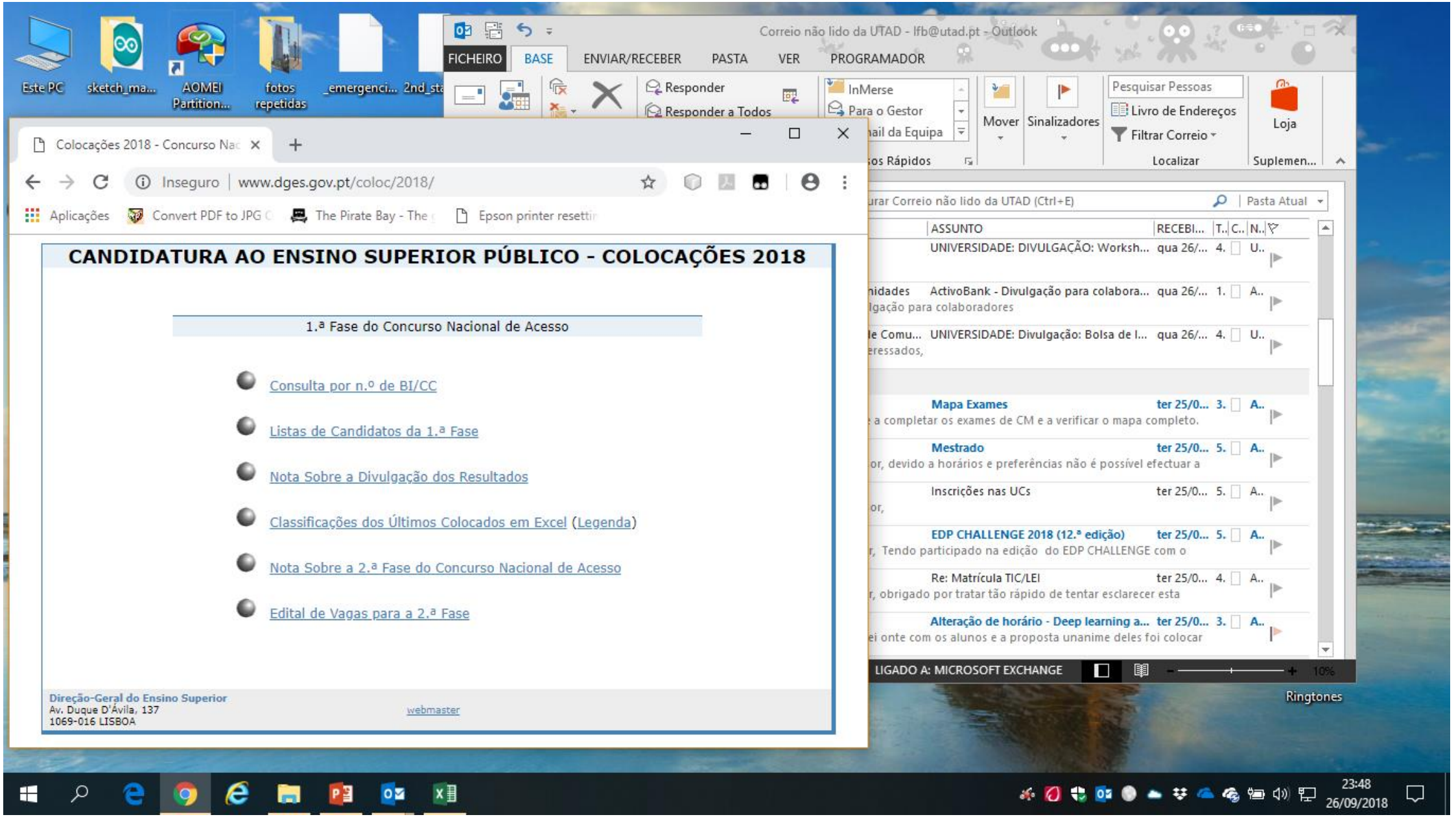
Concept of client-server application

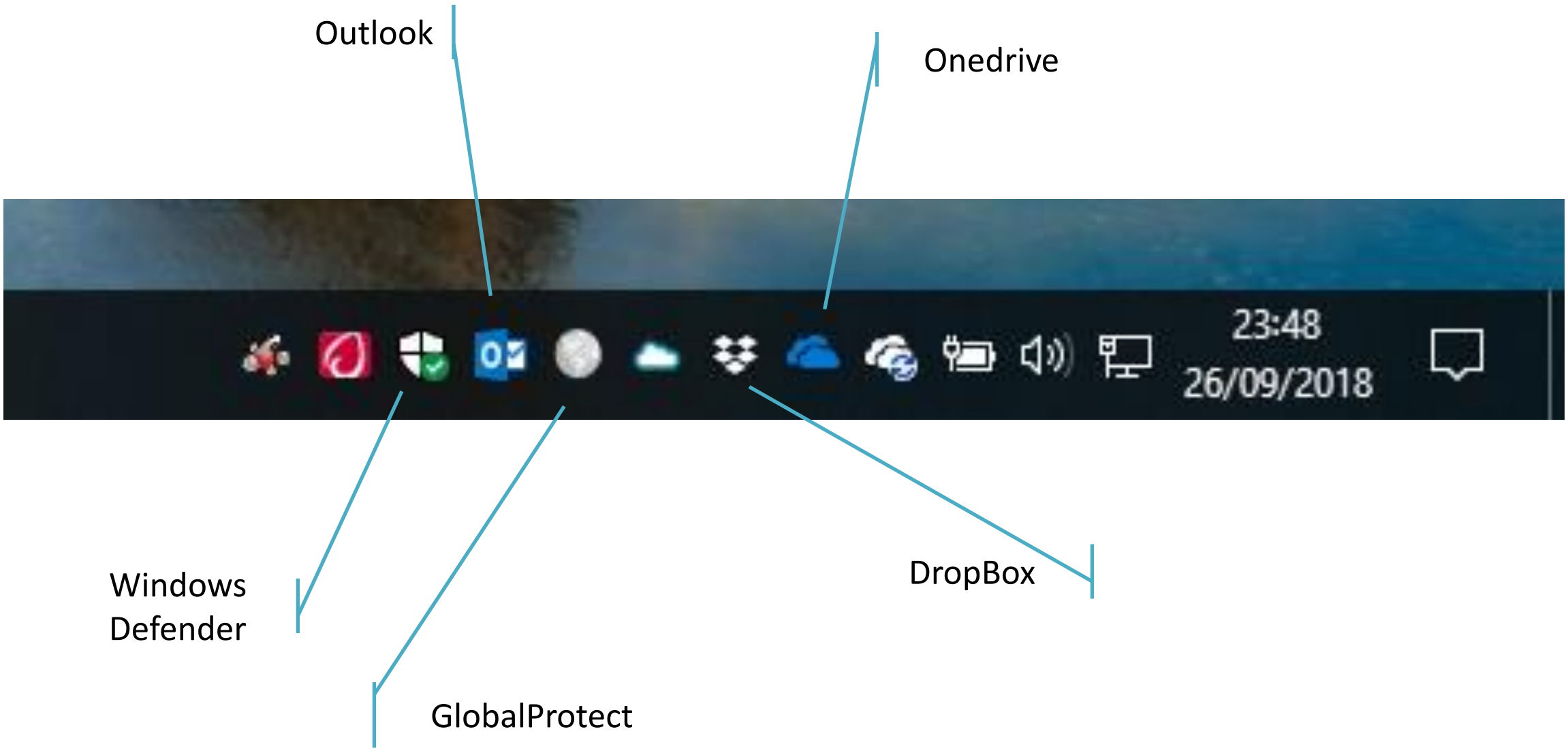
Web Engineering

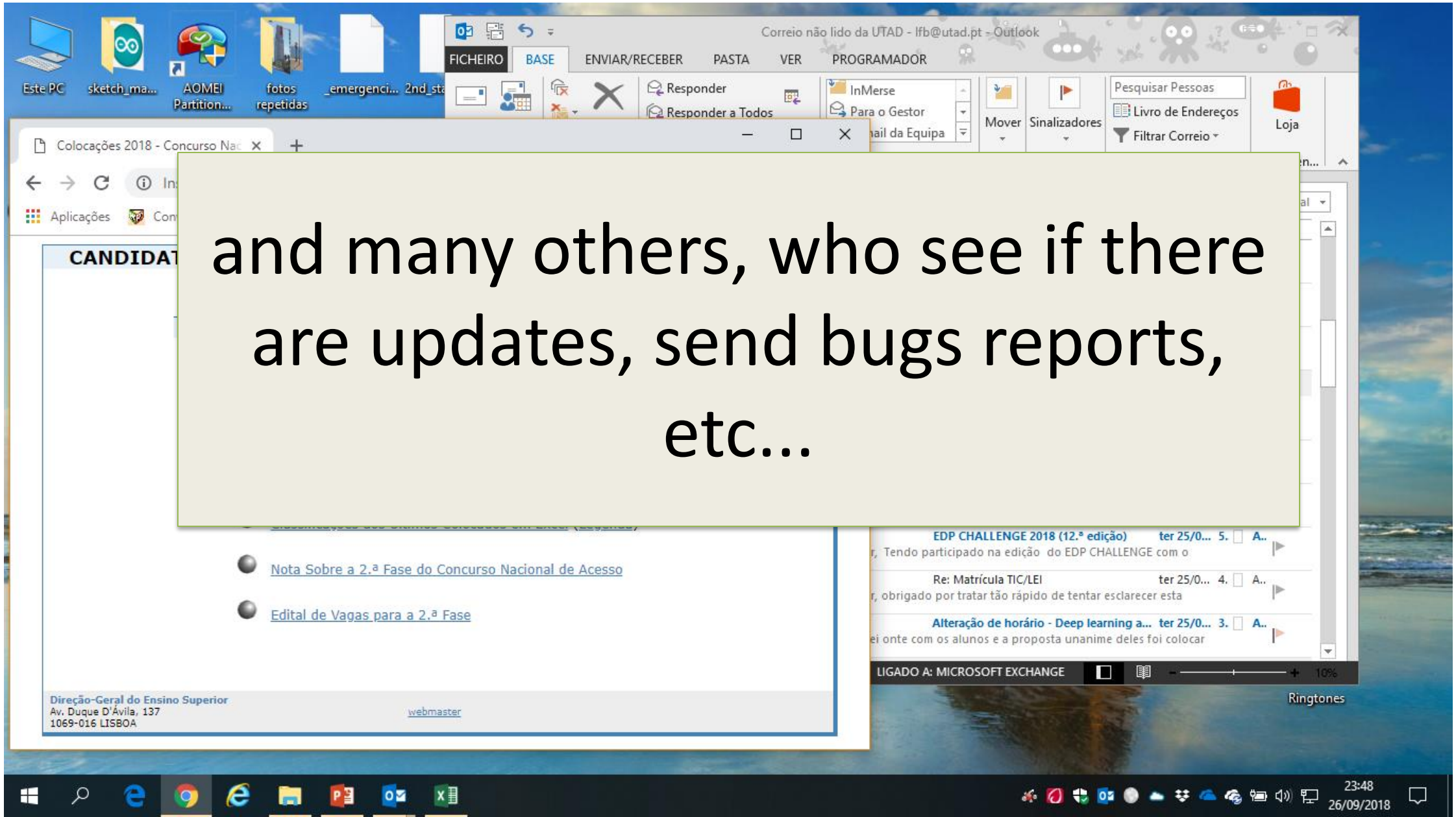
utad UNIVERSIDADE DE TRÁS-OS-MONTES E ALTO DOURO

Client-server Applications?



Client-server Applications?





and many others, who see if there are updates, send bugs reports, etc...

…using the most varied protocols (public, private, different network layers, etc.)

Browser: HTTP, port TCP 80
Second Life: *Second Life Open Grid Protocol* (SLOPGP), ports TCP 443 e 12043, ports UDP 12035, 12036 e 13000-13050
Outlook: *Office Exchange Protocols* (OEP), functionality dependent ports (for example: NNTP 119 to access *newsgroups* as Outlook shared folders).
Messenger: MSNP10, port TCP 1863.
Skype: *Skype Protocol*, ports TCP 81 e 443 or all TCP/UDP above 1024.
BOINC Manager: HTTP, TCP 81 e 443, access to *localhost* on 31416.

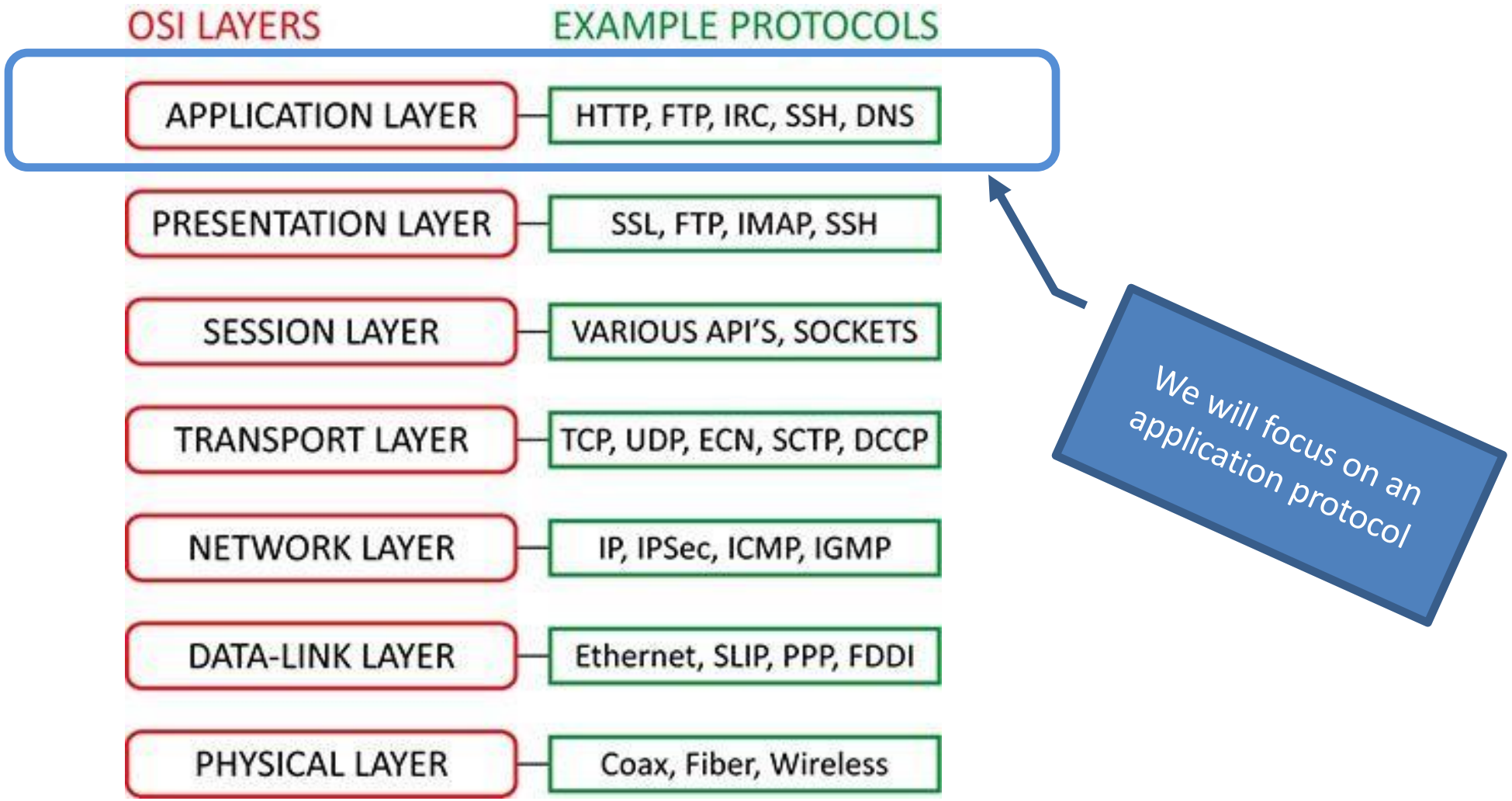
Protocol ports (common examples)

20/TCP	FTP (File Transfer protocol) - data port
21/TCP	FTP (File Transfer protocol) - control (command) port
22/TCP,UDP	SSH (Secure Shell) - Used for secure logins, file transfers and port redirects
23/TCP,UDP	Telnet protocol - Unencrypted text communication
25/TCP,UDP	SMTP (Simple Mail Transfer Protocol) - used for e-mail routing between servers (Currently is being used port 587)
53/TCP,UDP	DNS (Domain Name System)
80/TCP	HTTP (Hypertext Transfer Protocol) - used to transfer WWW pages
81/TCP	Skype protocol
110/TCP	POP3 (Post Office Protocol version 3): E-mail Protocol, version 3 - used for receiving e-mail
143/TCP,UDP	IMAP4 (Internet Message Access Protocol 4) - used to receive e-mail

Protocol ports (common examples)

156/TCP,UDP	SQL Service
443/TCP	HTTPS - HTTP Protocol over TLS/SSL (secure transmission)(secure transport layer)
989/TCP,UDP	FTP Protocol (data) over TLS/SSL
990/TCP,UDP	FTP Protocol (control) over TLS/SSL
991/TCP,UDP	NAS (Netnews Admin System)
992/TCP,UDP	Telnet protocol over TLS/SSL
993/TCP	IMAP4 over SSL (secure transmission)
995/TCP	POP3 over SSL (secure transmission)
...	

Open Systems Interconnection model (OSI model)



What Is a Web Application?

A Web application (Web app) is an **application program that is stored on a remote server and delivered over the Internet through a browser interface**.

Discussion topic...

Web vs Standalone

Discussion topic…

Web vs Standalone

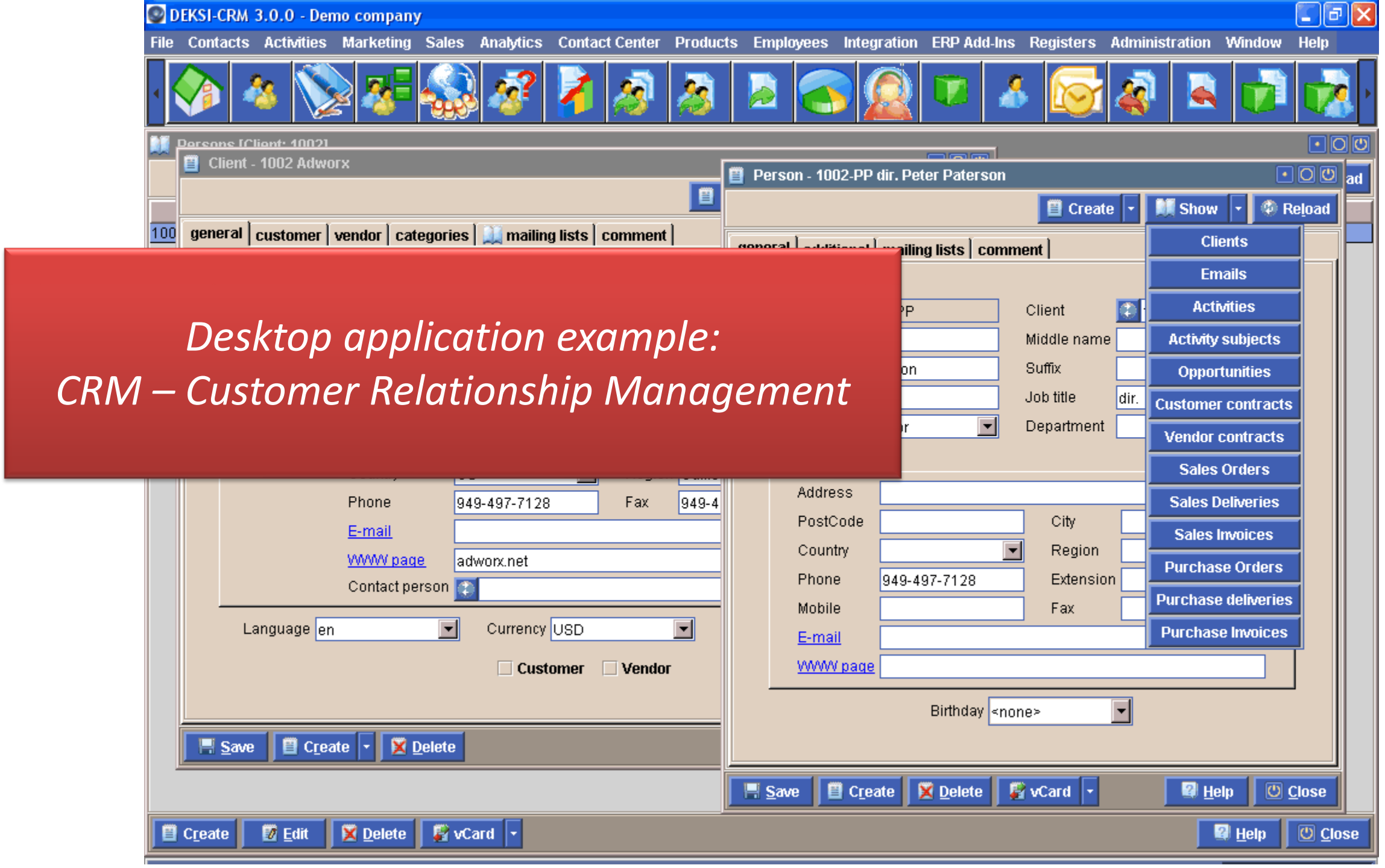
The Pros of Web Apps
- Cross-platform
- Automatic updates
- Resource friendly
- No installation

The Pros of Desktop Apps
- Improved performance
- Stronger security control
- No Internet

The Cons of Web Apps
- Security
- Internet required

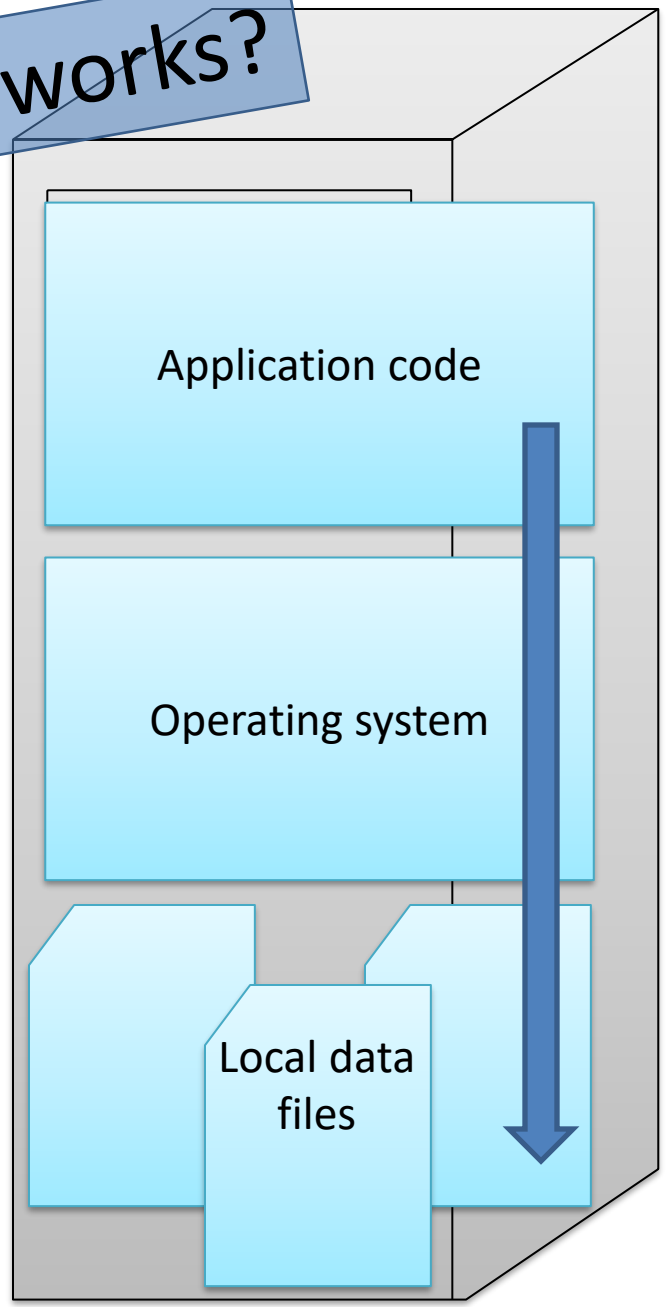
The Cons of Desktop Apps
- Limited to device
- Hard drive resources
- Manual installation and updates

Discussion topic…



Desktop application example:
CRM – Customer Relationship Management

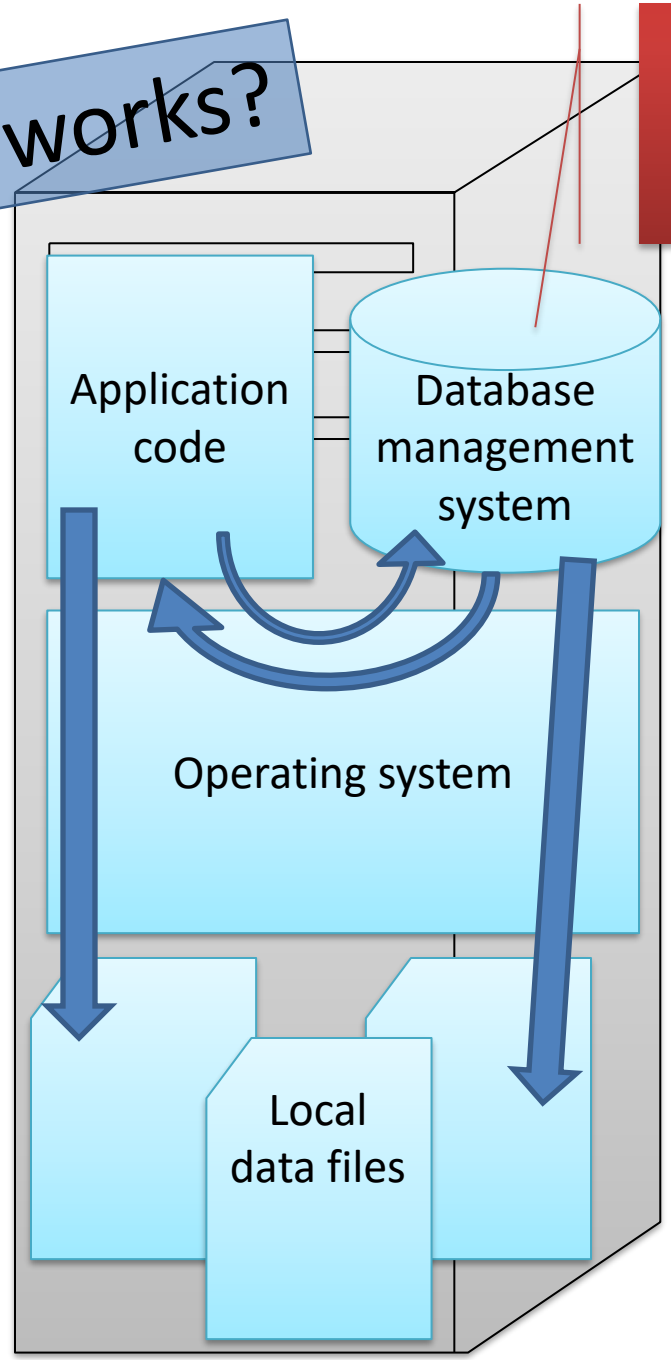
How it works?



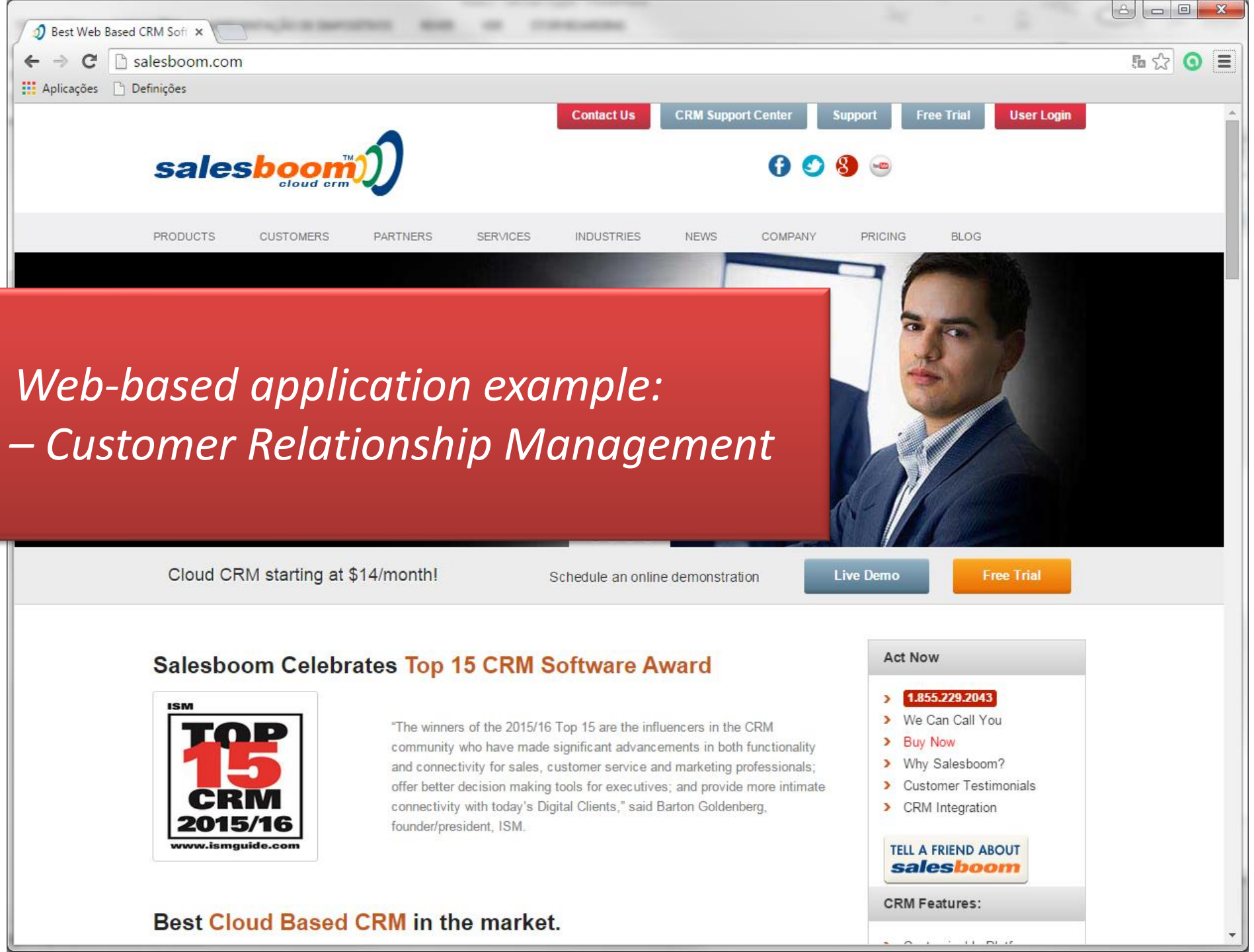
Desktop application example:
CRM – Customer Relationship Management

How it works?

Database can be an application-server or be integrated.

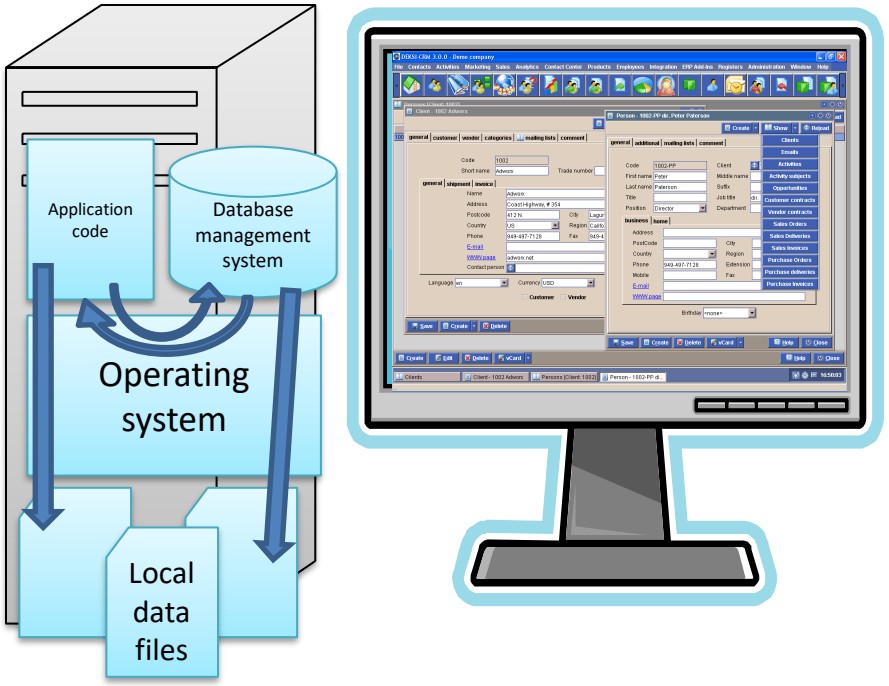


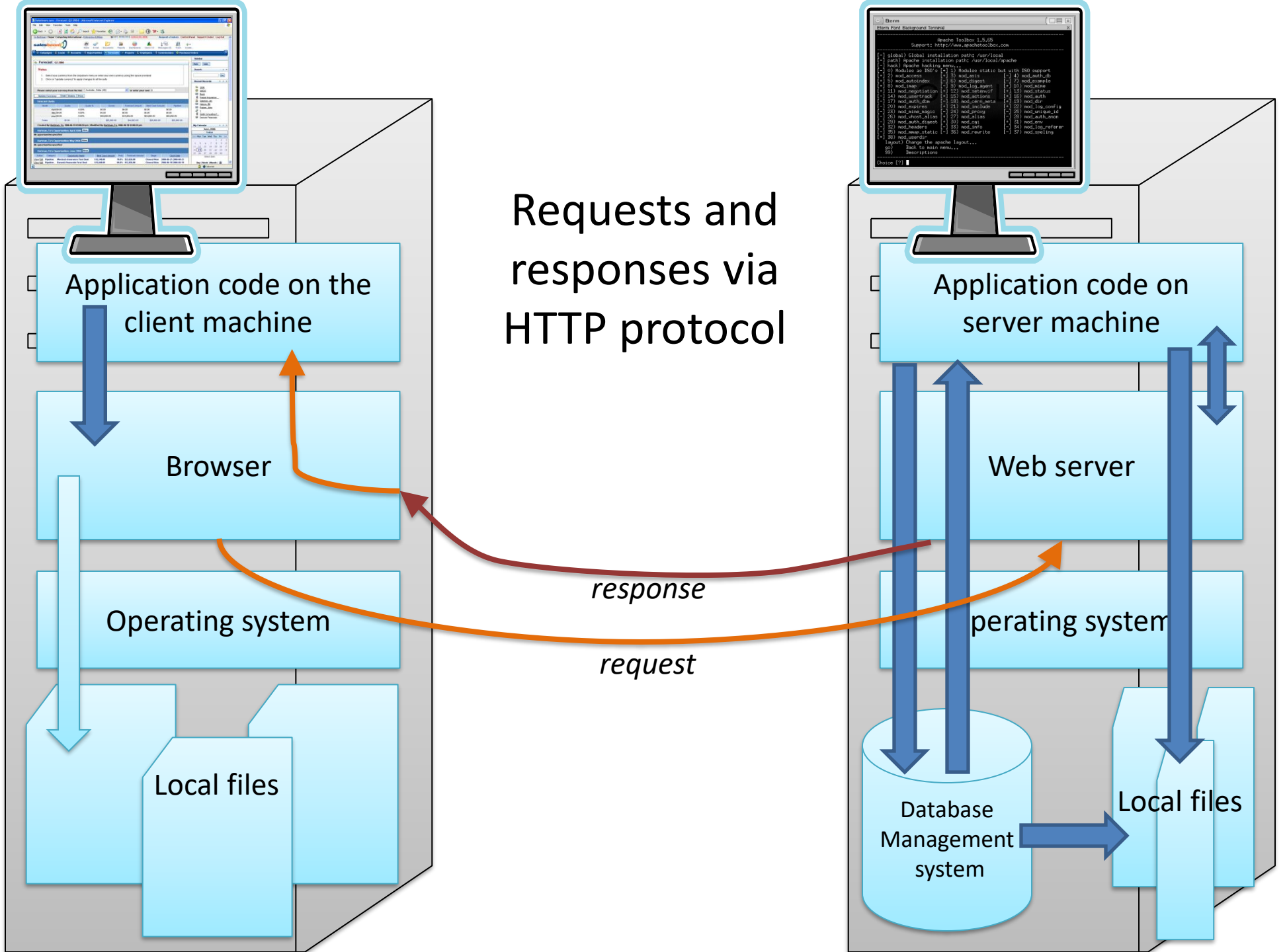
Desktop application example:
CRM – Customer Relationship Management



Web-based application example: CRM – Customer Relationship Management

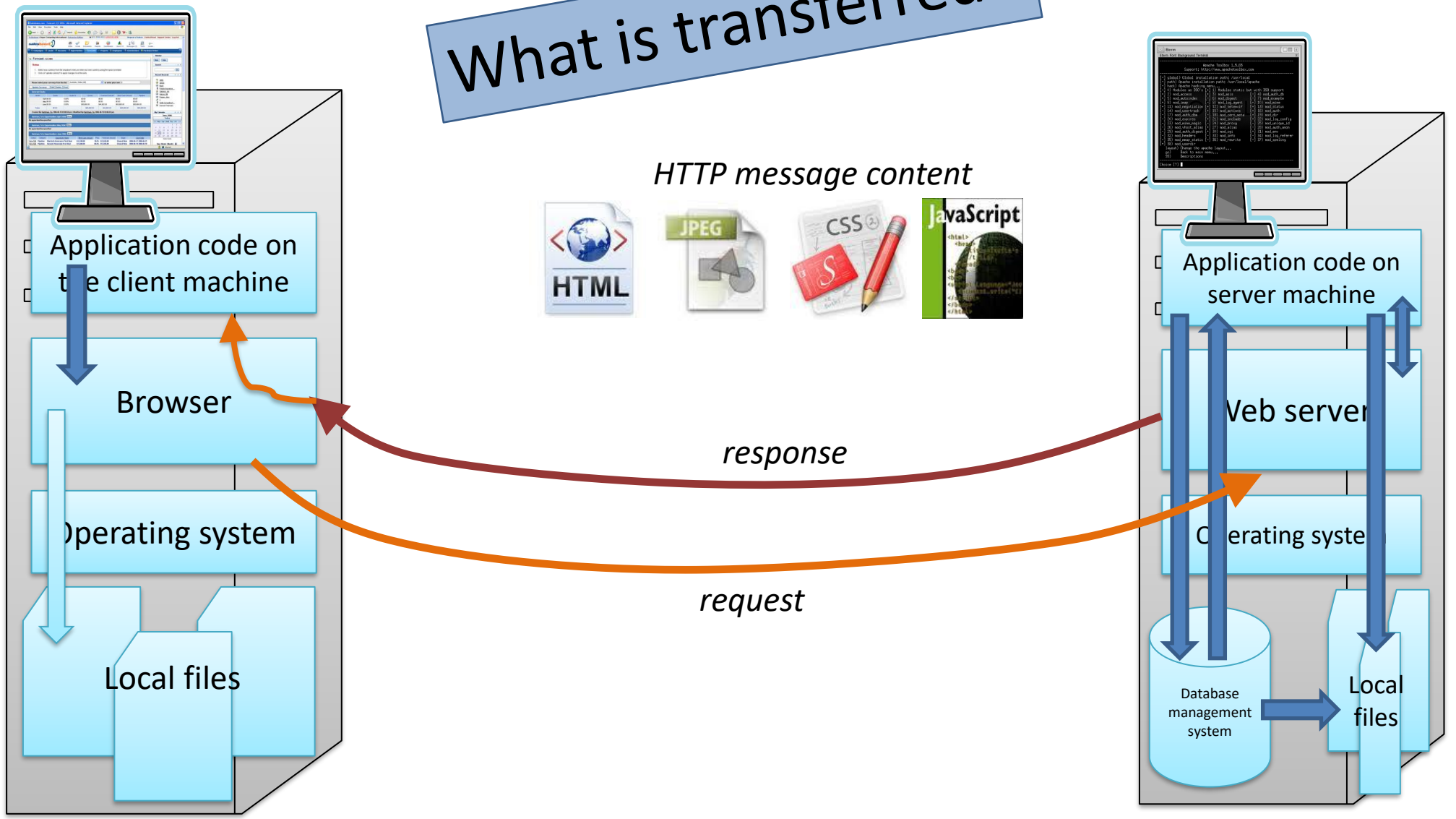
With the Web, what changes in previous models?





Web-based application example: CRM – Customer Relationship Management

What is transferred?



What is transferred?
(content examples)

HTTP message content


```
<!doctype html>
<html>
  <head>
    <title>Peanut Butter and Jelly Sa
    </title>
  </head>
  <body>
     <h1> style="color:red; align:cer
     Butter and Jelly Sandwich </h1>
<br>
<h2>Ingredients:</h2>
<ul>
<li>Peanut Butter</li>
<li>Jelly</li>
<li>Bread</li>
</ul>
<br>
<p>Smear the peanut butter on the bre
Then place a slice of bread on top.
  </body>
</html>
```

```
ÿØÿà ┼JFIF ÿþ ;CREATOR: gd-jpeg v1.0 (using IJG JPEG v62), quality = 80ÿÛ C
─┘|─|┘—|─••─◘┼ ¶⁋☼┼┤¶↑↑┤¶┬┬→%→←#┬┬ , #&')*)├-0-(0%0(ÿÛ C •••◘!!!(→┬→
((((((((((((((((((((((((((((((((((((((((((((((((((ÿÀ ◄◘ ¼ ¼└ " ◄ └◄ ÿÄ |
└┘|─•◘ ß ÿÄ µ┼ └└ ┘└|┘┘ } └┘◄|↕!1A─‼Oa•"σ¶2'::◘#R+Á┴RÑÃ$3br
┬┤↑├→%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz
´µ¶·¸¹º ÂÃÄÅÆÇÈÉÊÒÓÔÕÖ×ØÙÚáâãäåæçèéêñòóôõö÷øùúÿ
└┘|─•◘ ß ÿÄ µ◄ ┘┘ └•|┘┘ w └◄┘|!1─
#3Rð┴brÑ┬$4á%ñ┤↑├→&'()*56789:CDEFGHIJSTUVWXYZcde
˜™š¢£¤¥¦§¨©ª²³´µ¶·¸¹ºÂÃÄÅÆÇÈÉÊÒÓÔÕÖ×ØÙÚâãäåæçèé
( □Í┴ïÿ 4Hcð¼Zê-³Ïöß*Dš◄#J€p‰ž„"@1ƒèh¯nñÎô‹←¹-µåÚ
´€¹ë´n>š áp}┴ôG¹¼K¦ëz=Êé'Z—◄□Ëun-±=à|éêzŠó✲Œ→¶✲‰
(¢€(¢€(¢€(¢€(¢€(¢€(¢€(¢€(¢€(¢€(œß¶ QWãÓ%}┬]H:yi
$øe ^A<→~·*<q»©rÛßÞø◄éÞ¼‡[Ò/ô=F[VÚK{Ï*ã¯¸=Ç½ P¢Š
ôWw✲€Rß^›J×5H¬Zx·é×{w[]↕xùû ?Ï¯1â-♦Qðî¢öZµ³A0ä‼Ê¸ô
$¸Ž?«□Nµ«ñ7Á‰à›Ý6ËíMu<öþ✲Œ(mÄaG\qÞ€8Ê+¢ñ┤„ï´}:ËS
{§ÁŸ–hPxy|]â) •Ao-%ÁŠ┼§├`z¶z✲§z—Æ|Ô¾ ë ¨é:U¦‰
„Zç‡u[MPC¦kööï½¿┘Œ¥ñë◘ Ÿb□ïü_áo♫ø÷Ã×┤ÖQ®›«Y!Þ→1'0
$┬pK<îp±Ä¥™¾€UWHÔ´‰◄5[♫»'q•┤◄4e¾™¶ FŠJZ (¢Š (¢
JgP┼tAþ4 ÏR³–8PIö©-!k›`¨A–•Â îN+¯‡Á:"R♦X†'ë"ÖU+Ó¤íÍ
£ûãük+Rð}õ„×2ÄÁ"]Ä—SY¬e ;)¯¾\ñîróQ]ÇÃï 7‰m.õMJ
\Ó¼ |à¯‼<¶^‼ñ-ãj¨…':¸ÀY1éòƒüëË5+)ôÝFæÆñ6\[ÈÑH¾Œ♫
¢ŠÑðíu¯kVšeŠnžáÂ✲EØû "@├ÔW¡üYð|¯,┤Lû-ì÷Mu¿□¨ ·1
Rp□* å¨¯Q—Ã□✝÷5´^,Ô|Êñç4@ÄO·Ê8ükÌ$$pŒY ↕1'@¢´ü/
€¼♦ êRXjž*¾,ê0ß"'» ŒŽBÐ♦□Ñ^¥sðËMÕ´‹›ï└x5g¶←¤µ'6Io□L
sõ ♫&Šé¾"ø]¼#â<3Ìya²C+✝┤B?žr?æ£F′EHÔ³± ÔŸJ J+Õ¼
°3u└¿┴å4 QE¶ QE¶ W§øK["Møc¨Ï♫bKWhâÿ fÍp7}Bƒù×¯W£øo
```

```
body {font-size: medium;
        font-family: Serif;
        background-color: bl
        color: white;}
#page {background-color: whi
        color: black;
        padding: 1em;}
h1 {font-size: xx-large;
   font-family: Sans-Serif;
   color: black;
   text-align: center;
   border-bottom: solid 4px
p {text-indent: 1em;
   text-align: justify;
   lin-height: 150%;}
```

```
function get_nextSibling(n)
{
var x=n.nextSibling;
while (x.nodeType!=1)
  {
  x=x.nextSibling;
  }
return x;
}

var x=document.getElementsByTagName("li")[0];
document.write(x.nodeName);
document.write(" = ");
document.write(x.innerHTML);

var y=get_nextSibling(x);

document.write("<br />nextsibling: ");
document.write(y.nodeName);
document.write(" = ");
document.write(y.innerHTML);
```

Resume

CLIENT-SERVER APPLICATION

WHAT IS A CLIENT-SERVER APPLICATION?
HOW IT WORKS?
EXAMPLES?

STANDALONE AND WEB APPLICATIONS

WHAT ARE THE DIFFERENCES BETWEEN STANDALONE APPLICATIONS AND WEB APPLICATIONS?
ADVANTAGES AND DISADVANTAGES?
HOW BOTH WORK?

Next section

REQUEST-RESPONSE SEQUENCE OF A WEB PAGE

Readings until September 23's class

Cap. 3: “Birth of the World Wide Web: HTTP”, pp. 29 to 44.