Ruby Monstas

Session 10

Agenda

- Recap: Methods
- Project: IRC Bot

Recap

Methods

You’ve used them before!

```
“Hello Rubymonstas”.length

“Test string”.include?(“Something”)

5.odd?

puts “Tadaaa!”
```

Method or Function

A piece of code which has a **name,** takes **input** and returns **output**.

Methods make your code reusable.

Related to the mathematical idea of a function.

Structure

- Method name
  - The name a method is known as in the rest of your code
- Input values (aka arguments, parameters)
  - The information the method needs given to do its job
- Body
  - The code that does something useful with the input
- Return value
  - The value that is return to the code that called the method

Structure

```
def my_first_method(first_argument, second_argument)
    # This is the method body,
    # Here comes some useful code

    return_value
end
```

Input values

- Optional, Comma-separated in parentheses
- Naming: Just like normal variables
- Only valid within the method, not before, not after

```
def calculate_age(year_of_birth)
    # code here
end

def save_user_data!(first_name, last_name, home_town)
    # code here
end
```

Method body

- Everything between the signature and the `end`
- Any Ruby code
- You can also call methods from within methods

```
def calculate_age(year_of_birth)
    age = 2015 - year_of_birth
end
```

Calling methods

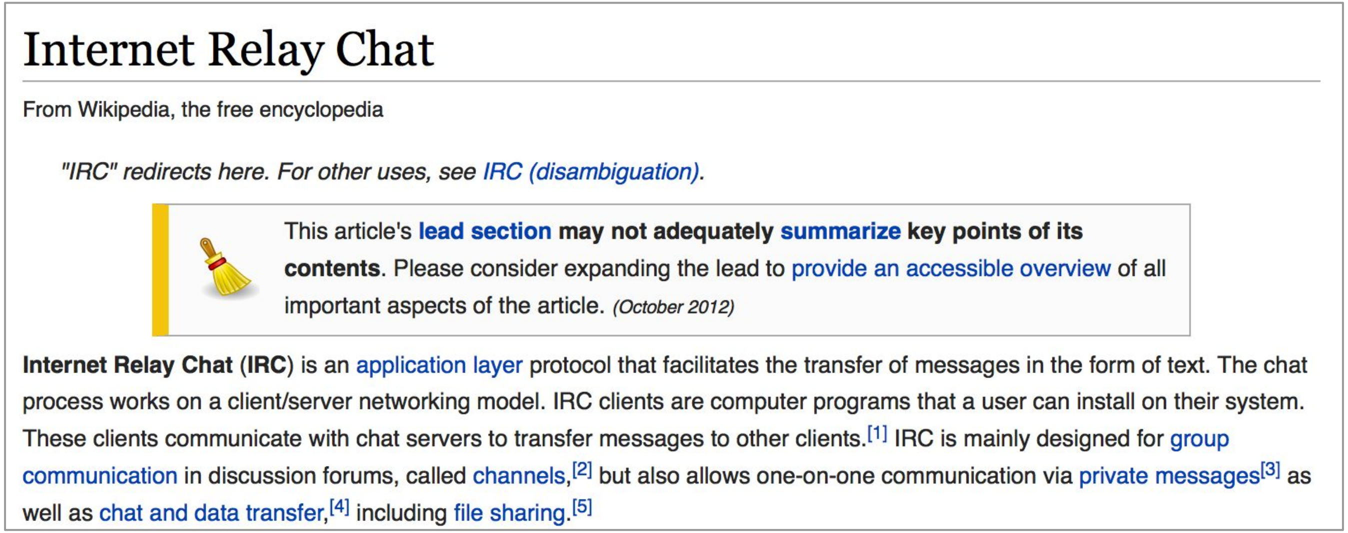
How do I use (“call”) my method?

```
calculated_value = my_first_method(42, “hello”)
my_first_method calculated_value, 53.45
```

Project: IRC Bot

IRC Bot?!

What’s IRC?



Internet Relay Chat

From Wikipedia, the free encyclopedia

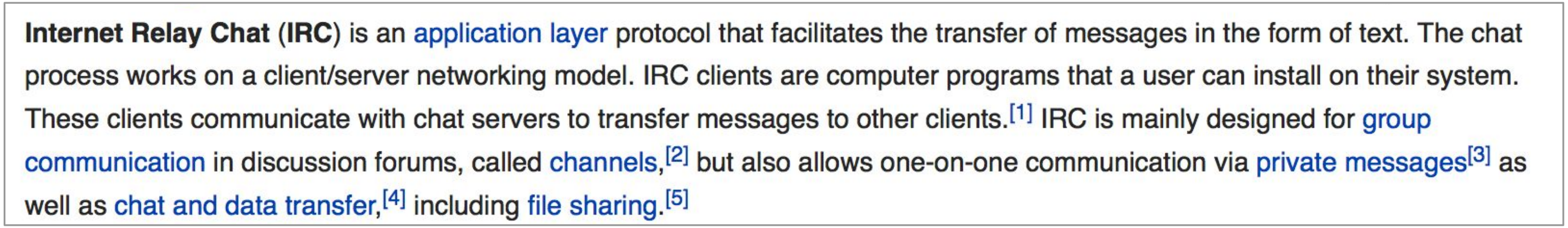
"IRC" redirects here. For other uses, see IRC (disambiguation).

This article's **lead section may not adequately summarize key points of its contents**. Please consider expanding the lead to provide an accessible overview of all important aspects of the article. *(October 2012)*

Internet Relay Chat (**IRC**) is an application layer protocol that facilitates the transfer of messages in the form of text. The chat process works on a client/server networking model. IRC clients are computer programs that a user can install on their system. These clients communicate with chat servers to transfer messages to other clients.[1] IRC is mainly designed for group communication in discussion forums, called channels,[2] but also allows one-on-one communication via private messages[3] as well as chat and data transfer,[4] including file sharing.[5]

Demo

IRC



Internet Relay Chat (**IRC**) is an application layer protocol that facilitates the transfer of messages in the form of text. The chat process works on a client/server networking model. IRC clients are computer programs that a user can install on their system. These clients communicate with chat servers to transfer messages to other clients.[1] IRC is mainly designed for group communication in discussion forums, called channels,[2] but also allows one-on-one communication via private messages[3] as well as chat and data transfer,[4] including file sharing.[5]

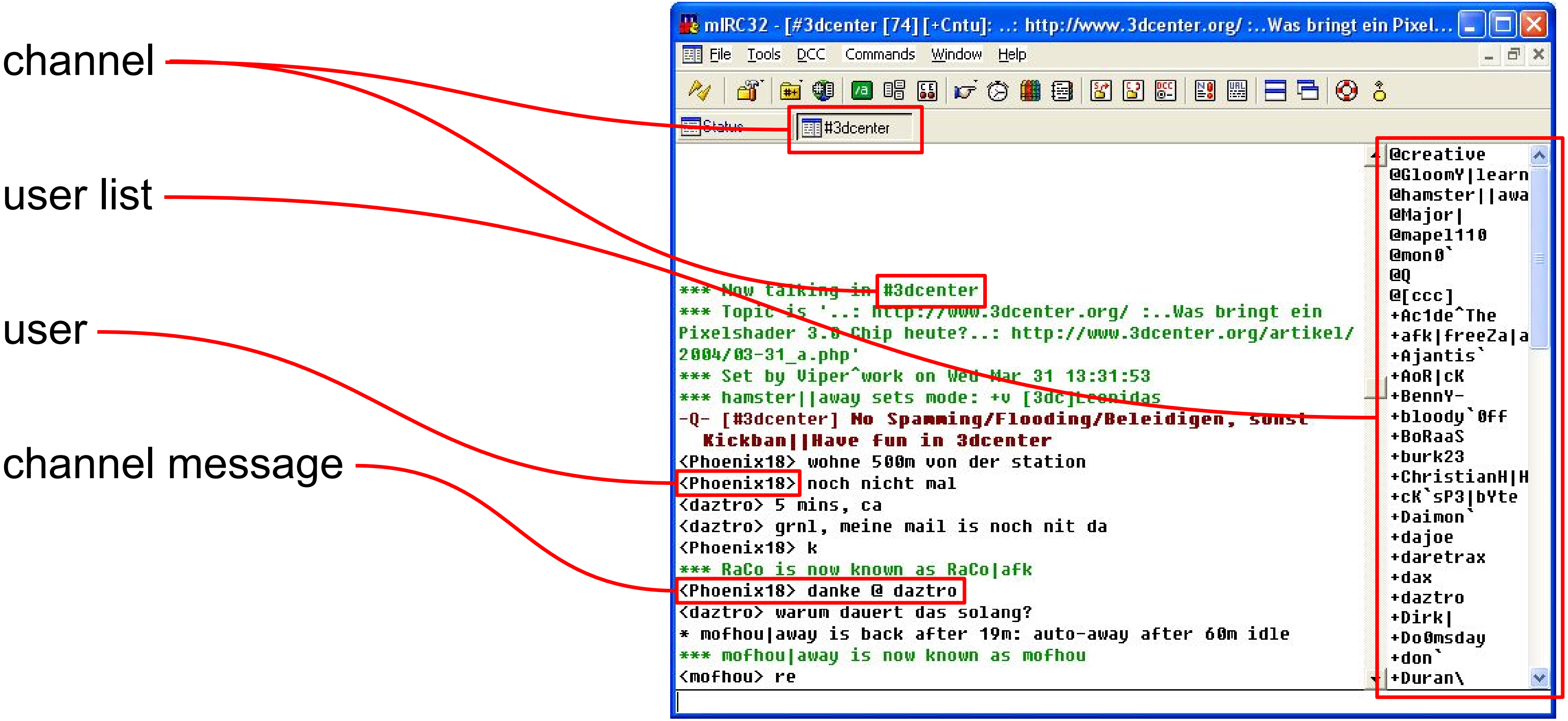
protocol

messages

client / server networking

channels

IRC


channel
user list
user
channel message
mIRC32 - [#3dcenter [74] [+Cntu]: ..: http://www.3dcenter.org/ :..Was bringt ein Pixel...
File Tools DCC Commands Window Help
Status
#3dcenter

```
*** Now talking in #3dcenter
*** Topic is '..: http://www.3dcenter.org/ :..Was bringt ein
Pixelshader 3.0 Chip heute?..: http://www.3dcenter.org/artikel/
2004/03-31_a.php'
*** Set by Viper^work on Wed Mar 31 13:31:53
*** hamster||away sets mode: +v [3dc]Leonidas
-Q- [#3dcenter] No Spamming/Flooding/Beleidigen, sonst
  Kickban||Have fun in 3dcenter
<Phoenix18> wohne 500m von der station
<Phoenix18> noch nicht mal
<daztro> 5 mins, ca
<daztro> grnl, meine mail is noch nit da
<Phoenix18> k
*** RaCo is now known as RaCo|afk
<Phoenix18> danke @ daztro
<daztro> warum dauert das solang?
* mofhou|away is back after 19m: auto-away after 60m idle
*** mofhou|away is now known as mofhou
<mofhou> re
```

```
@creative
@GloomY|learn
@hamster||awa
@Major|
@mapel110
@mon0`
@Q
@[ccc]
+Ac1de^The
+afk|freeZa|a
+Ajantis`
+AoR|cK
+BennY-
+bloody`0ff
+BoRaaS
+burk23
+ChristianH|H
+cK`sP3|bYte
+Daimon`
+dajoe
+daretrax
+dax
+daztro
+Dirk|
+Do0msday
+don`
+Duran\
```

What’s on the client or server?

Client

Server

Sends messages to server

Receives messages from server

Receives messages from clients

Sends messages to clients

Holds a list of channels

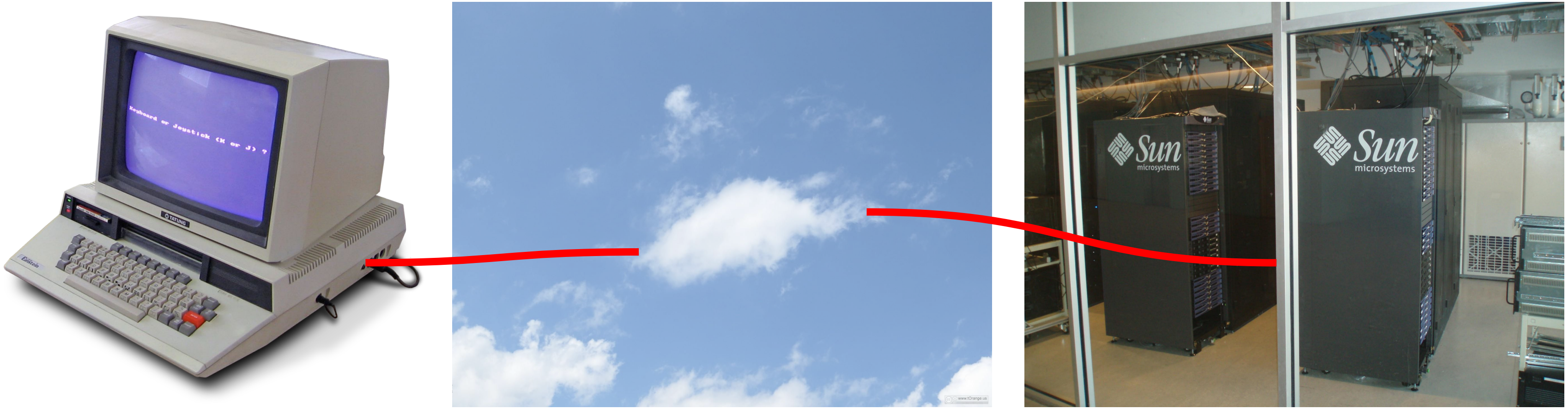
Holds a list of users

Client / Server Networking

Client

Internet (the “Cloud”)

Server



Client / Server Networking

Hey, what’s up?
Let’s talk!

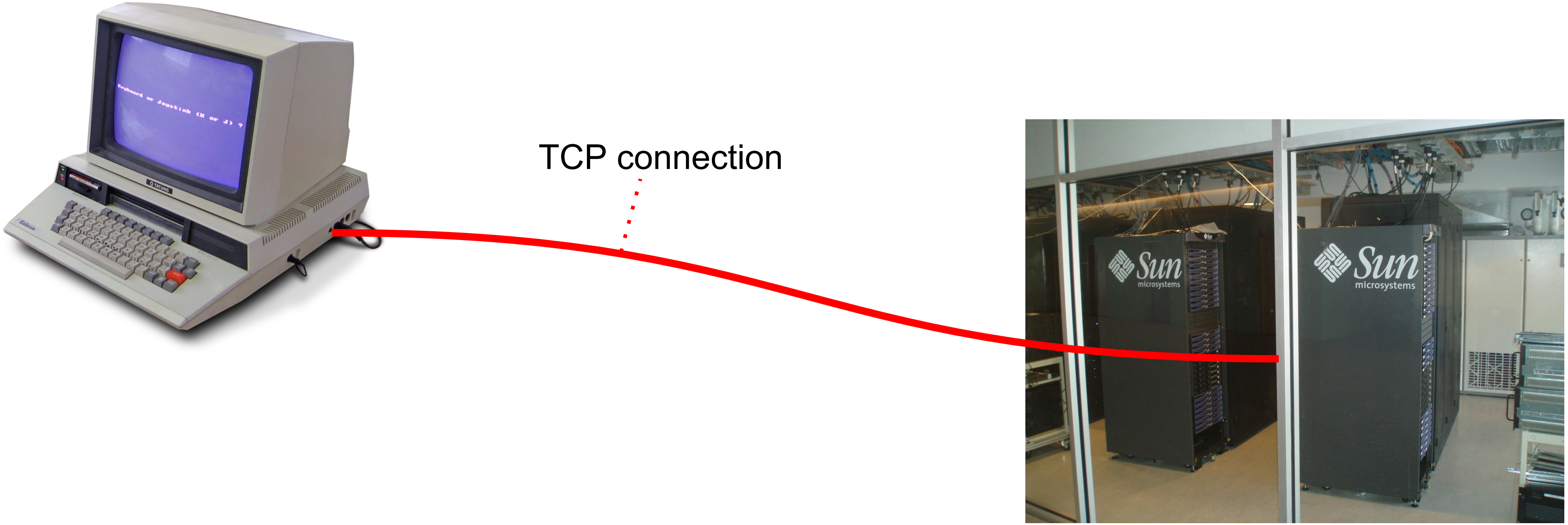
Client / Server Networking

Sure thing, let’s do it!

Client / Server Networking

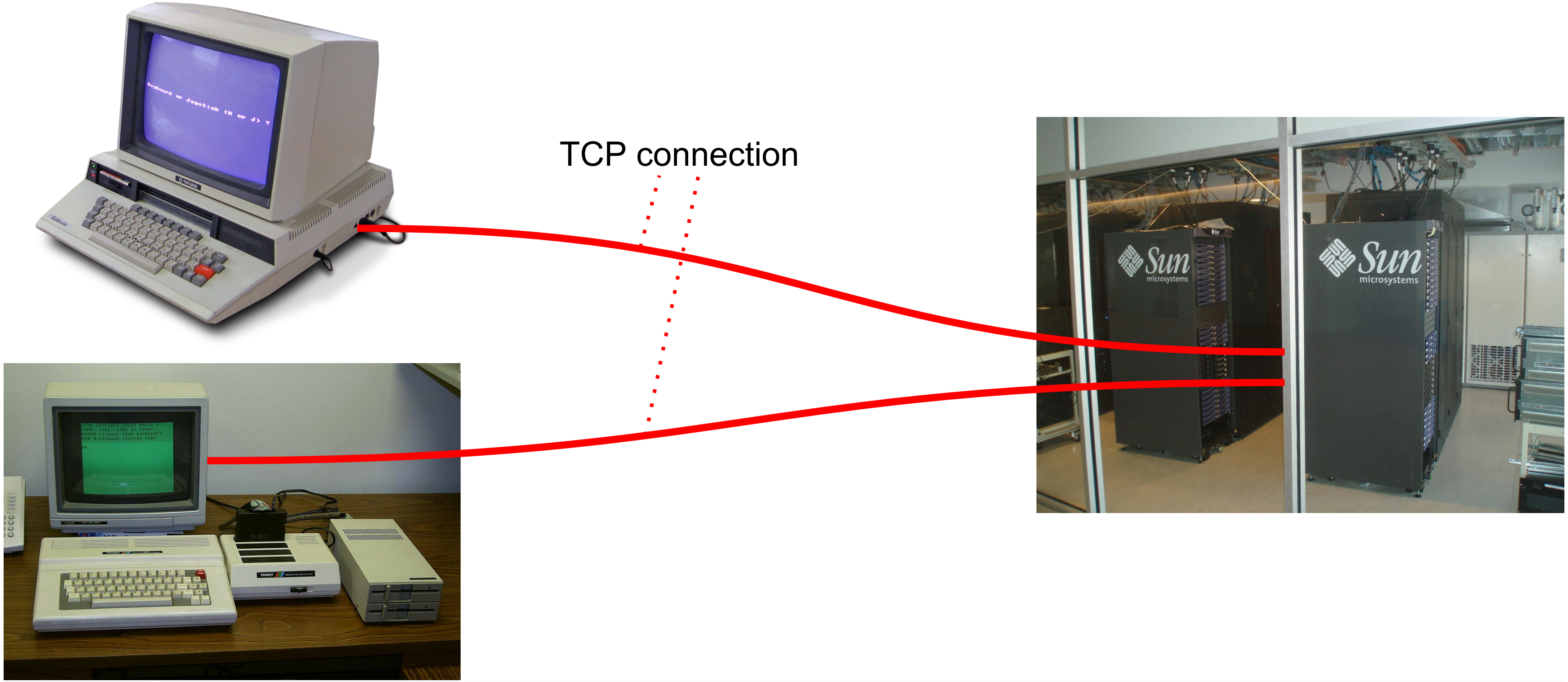
OK cool.

Client / Server Networking



TCP connection

Client / Server Networking



TCP connection

Client / Server Networking

IRC clients

IRC server

IRC Protocol

How do client and server talk to each other?

There’s a lot to clarify.

IRC Protocol

What could a **client** want to say?
“I want to choose a username, it should be `cool_guy123`”

“I want to join channel #coolcrowd”

“I want to write a message to the channel, it should be `‘Hi there!’`”

What could a **server** want to say?
“There’s a new message on the channel `#rubymonstas` by `cool_guy123`, he says `‘Hi there!’`”
“`ruby_programmer85` just joined `#rubymonstas`!”

“`nerdinand` just left `#rubymonstas`!”

IRC Protocol: Client

```
NICK nethad-bot

USER nethad-bot 0 * :nethad-bot

JOIN #rubymonstas

PRIVMSG #rubymonstas :hi all, how are you doing?
```

IRC Protocol: Server

```
:nethad!83.78.226.54 JOIN #rubymonstas

:nethad!83.78.226.54 PART #rubymonstas

:nethad!83.78.226.54 PRIVMSG #rubymonstas :hey, what’s up?
```

Bot

So what’s a (IRC) bot, then?

From Wikipedia:

“An **IRC bot** is a set of scripts or an independent program that connects to Internet Relay Chat as a client, and so appears to other IRC users as another user. An IRC bot differs from a regular client in that instead of providing interactive access to IRC for a human user, it performs automated functions.”

Things a bot could do

- Let the user play a game
- Google something and return results
- Return information about the weather
- Return random quotes from users
- … ? You name it!

Time to build a bot!

Let’s get to it!