Python

INTERMEDIATE

Rapid prototyping using Python libraries and integration with local and remote services

POLITECNICO
DI TORINO

e-Lite

The Starting Point

- You (should) have the basic knowledge for creating Python programs from scratch
- In realizing an application, you can *stand on the shoulders of giants*
  - i.e., reuse things (libraries, packages, etc.) made available by other developers
- In looking for libraries to integrate with your application, it could be useful to
  - Keep It Simple Stupid (KISS)

What can I integrate?

- Libraries and packages to provide more functionality, algorithms, etc. to your own program
  - e.g., complex math operations, statistical packages, OS interfaces, …
- Libraries and packages to integrate external (cloud, web) services
  - e.g., weather, social networks, …

Integrate Python Packages

- To use other Python packages in your applications you should
  - install them (typically)
  - import them
- Python modules can be installed with the `pip` command

```
pip install <package_name>
```

```
Learn more about pip at https://pip.pypa.io
```

Learning by Example

- We would like to realize a Telegram bot (AmIBot) that
  - greets us
  - acts as a textual parrot, by sending back as text whatever we write (*textual echo)*
  - acts as a parrot, by sending back as voice whatever we write (*vocal echo)*

What is a Telegram bot?

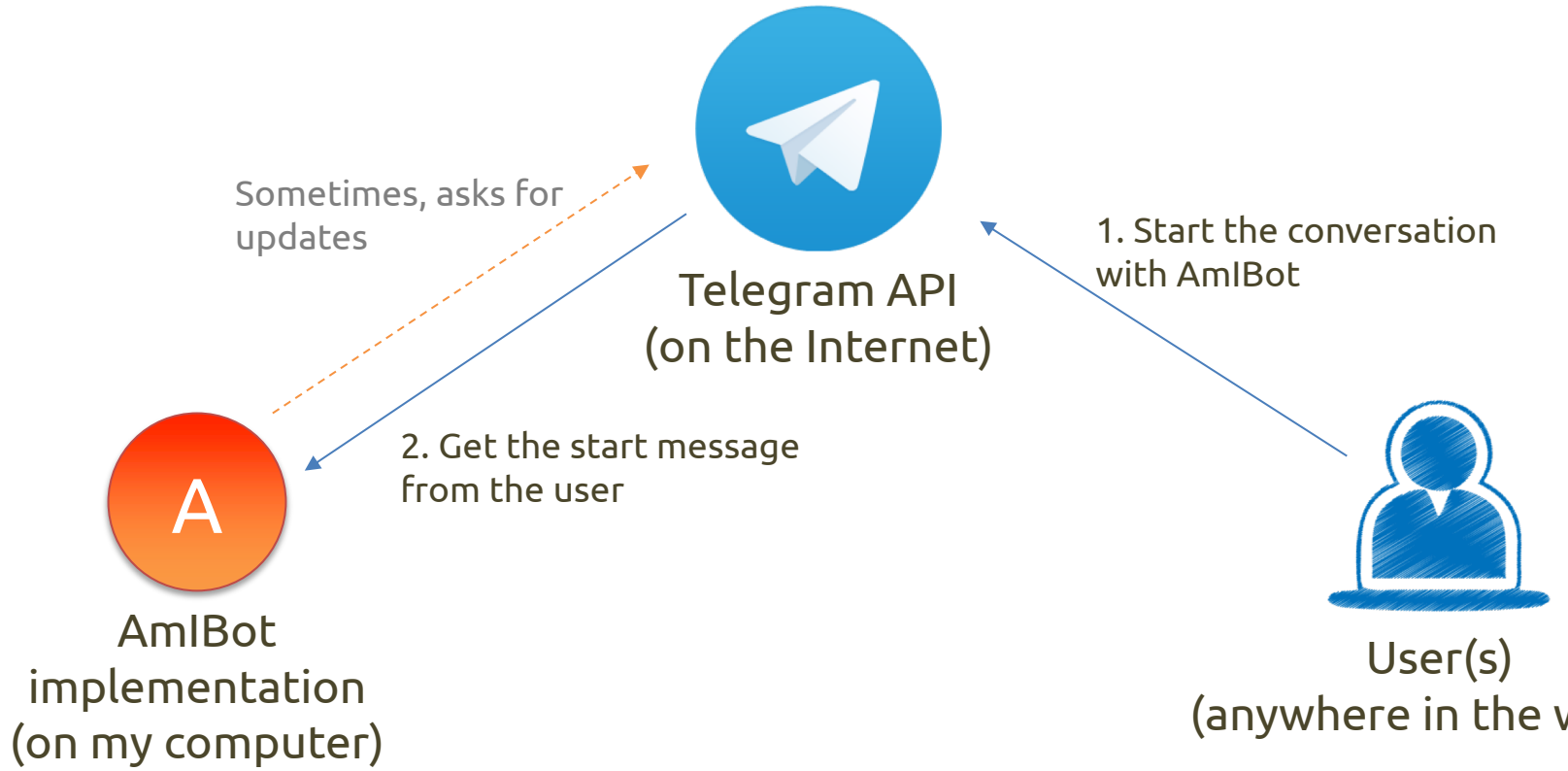
- A third-party application that run inside Telegram
- Users can interact with bots by sending them
  - messages
  - commands
  - inline requests
- Developers can control their own bots using the Telegram bot API or dedicated libraries

```
Discover more at https://core.telegram.org/bots
```

How to create a Telegram bot?

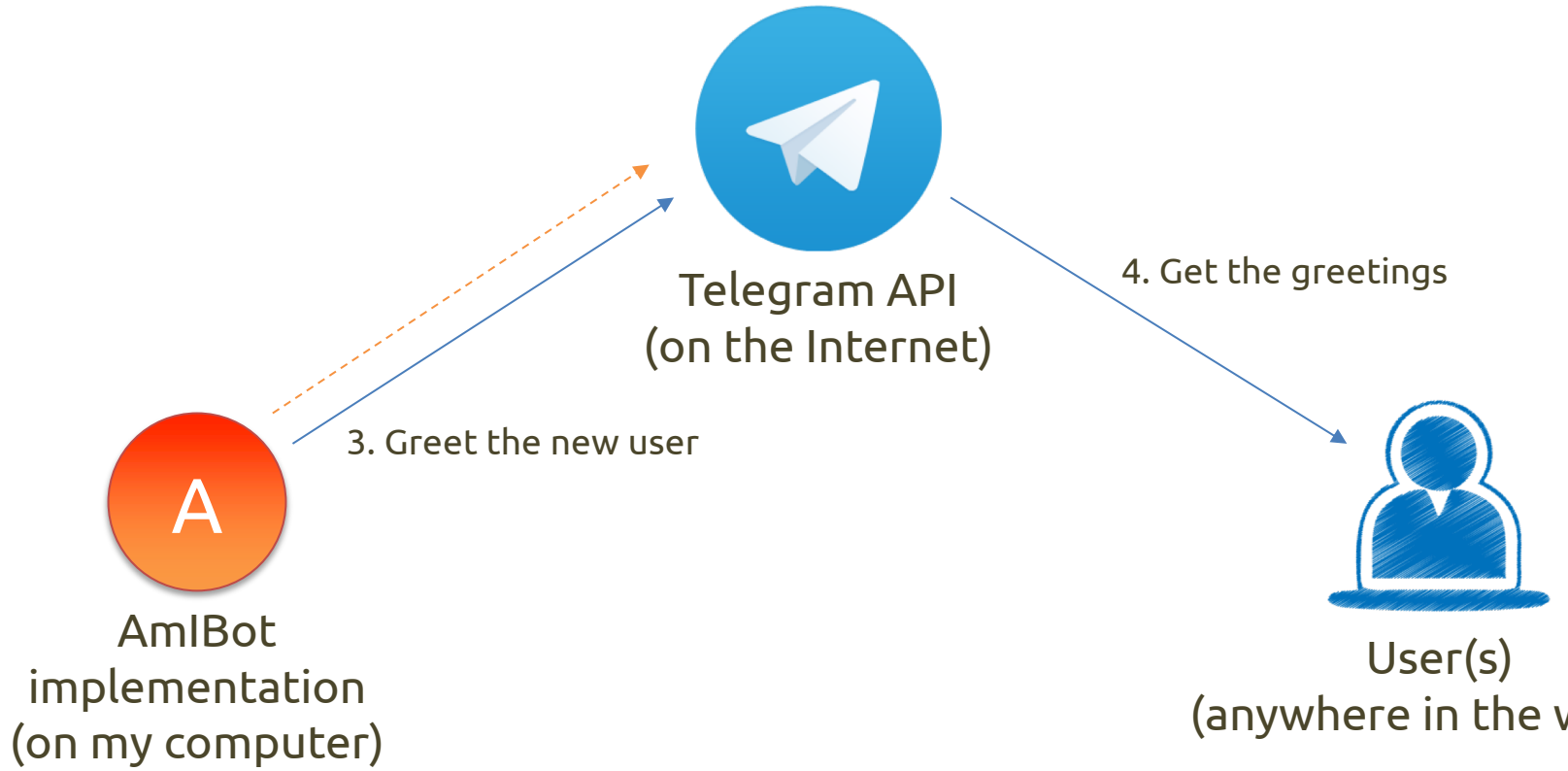
- First, ask BotFather
  - set up a bot account with @BotFather
  - it will ask you for
    - a name (I chose `AmIBot`)
    - a username that must end in 'bot' (I chose `AmI2018_bot`)
  - Then, BotFather will give you
    - a token to use the Telegram API
    - a Telegram link to let users converse with your bot

How will our bot work?



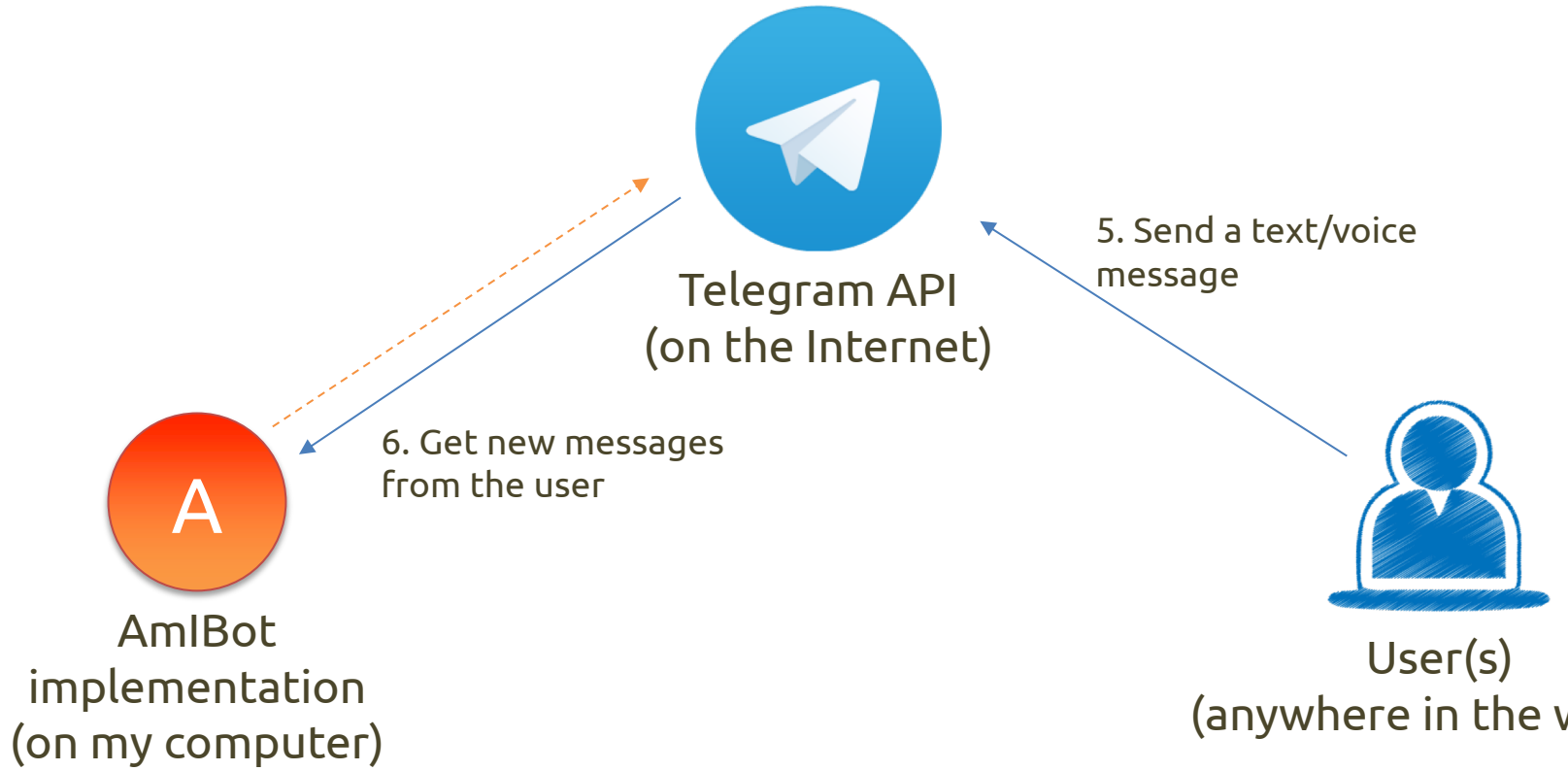
(from 10,000 feet…)

How will our bot work?



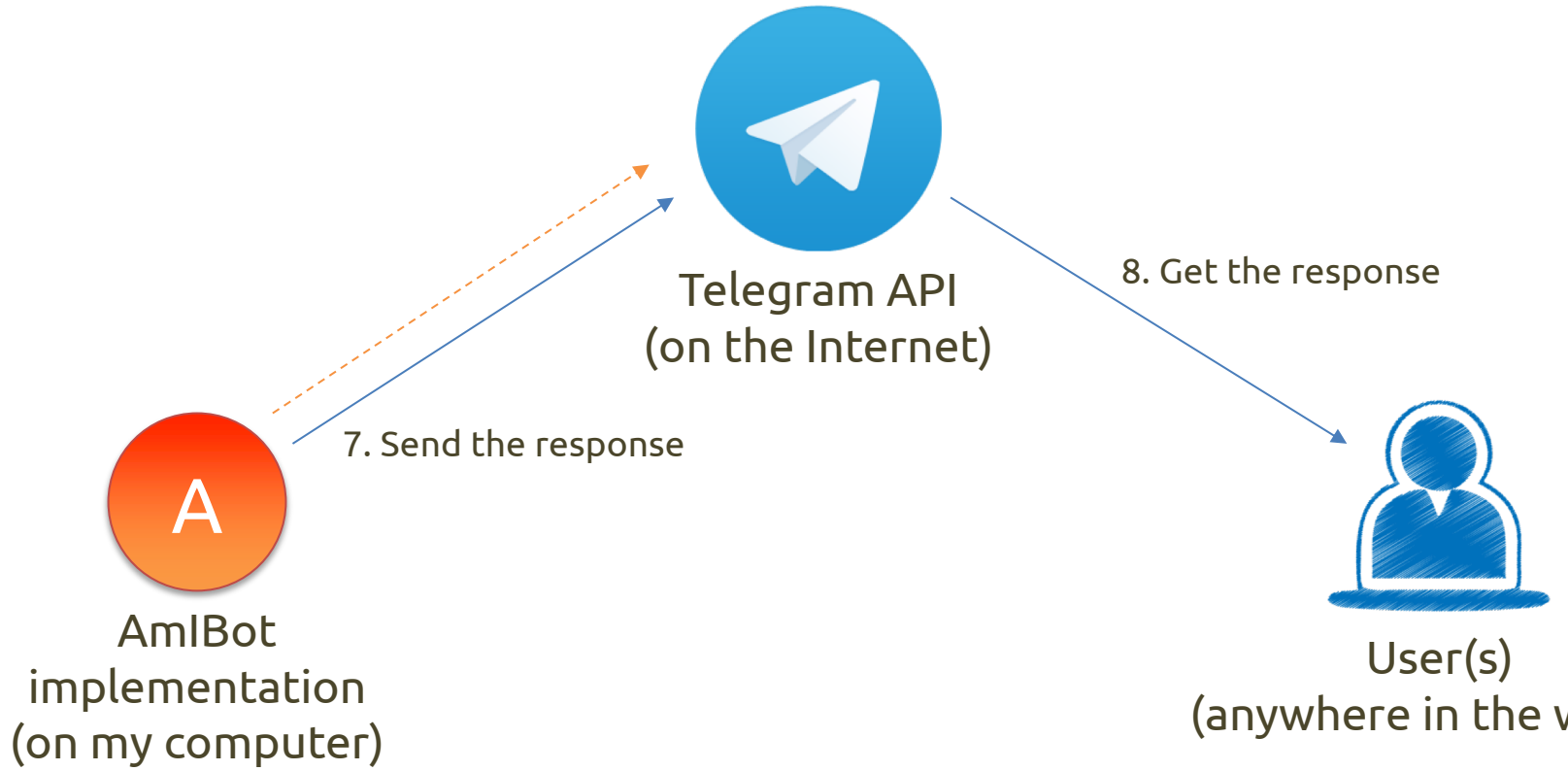
(from 10,000 feet…)

How will our bot work?



(from 10,000 feet…)

How will our bot work?



(from 10,000 feet…)

Which library do we choose?

- No official Telegram library for Python
- Three libraries are "recommended" by Telegram…
  - python-telegram-bot
    - https://github.com/python-telegram-bot/python-telegram-bot
  - Telepot
    - https://github.com/nickoala/telepot
  - twx.botapi
    - https://github.com/datamachine/twx.botapi
- Sometimes, other libraries may be widely used in tutorials and examples
  - e.g., by searching with Google

Which library do we choose?

- All the libraries
  - support Python 3
  - are available on pip
  - have some sort of documentations
  - are maintained
- How can we choose?
  - python-telegram-bot has more commits/stars/forks/watches… than the other two
  - twx.botapi has no available examples and implements a slightly older version of the bot API
- Let's go with **python-telegram-bot**, then!

Command vs non-command messages

- Command messages
  - e.g., \start, \edit, \something
- Non-command messages
  - free text messages
- We want to send greetings with the \start command message
- While the "echo" operation will be performed on every non-command message

The Text-To-Speech dilemma

- How to convert a text into speech?
- Online services exist
  - e.g., Google Text to Speech, VoiceRSS, …
- We will use the gTTS Python package
  - https://github.com/pndurette/gTTS
  - available on pip
  - it takes some text and, by using the Google Text to Speech API, return a mp3 file
- In our bot, we will send the mp3 back to the user
  - as a parrot (*vocal echo*)

Playground

- If you want to experiment more, you can…
- … perform an "inverted" echo
  - e.g., the user write "hello, bot" and the bot should reply with "olleh, tob"
- … ask for the weather
  - https://pypi.python.org/pypi/yahooweather

Questions?

01QZP Ambient Intelligence

Luigi De Russis
luigi.derussis@polito.it

POLITECNICO DI TORINO
e-Lite
SOME RIGHTS RESERVED
BY NC SA

License

- This work is licensed under the Creative Commons “Attribution-NonCommercial-ShareAlike Unported (CC BY-NC-SA 4.0)” License.
- You are free:
  - to **Share** - to copy, distribute and transmit the work
  - to **Remix** - to adapt the work
- Under the following conditions:
  - **Attribution** - You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
  - **Noncommercial** - You may not use this work for commercial purposes.
  - **Share Alike** - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.
- To view a copy of this license, visit https://creativecommons.org/licenses/by-nc-sa/4.0/