



E210 Engineering Cyber-Physical Systems

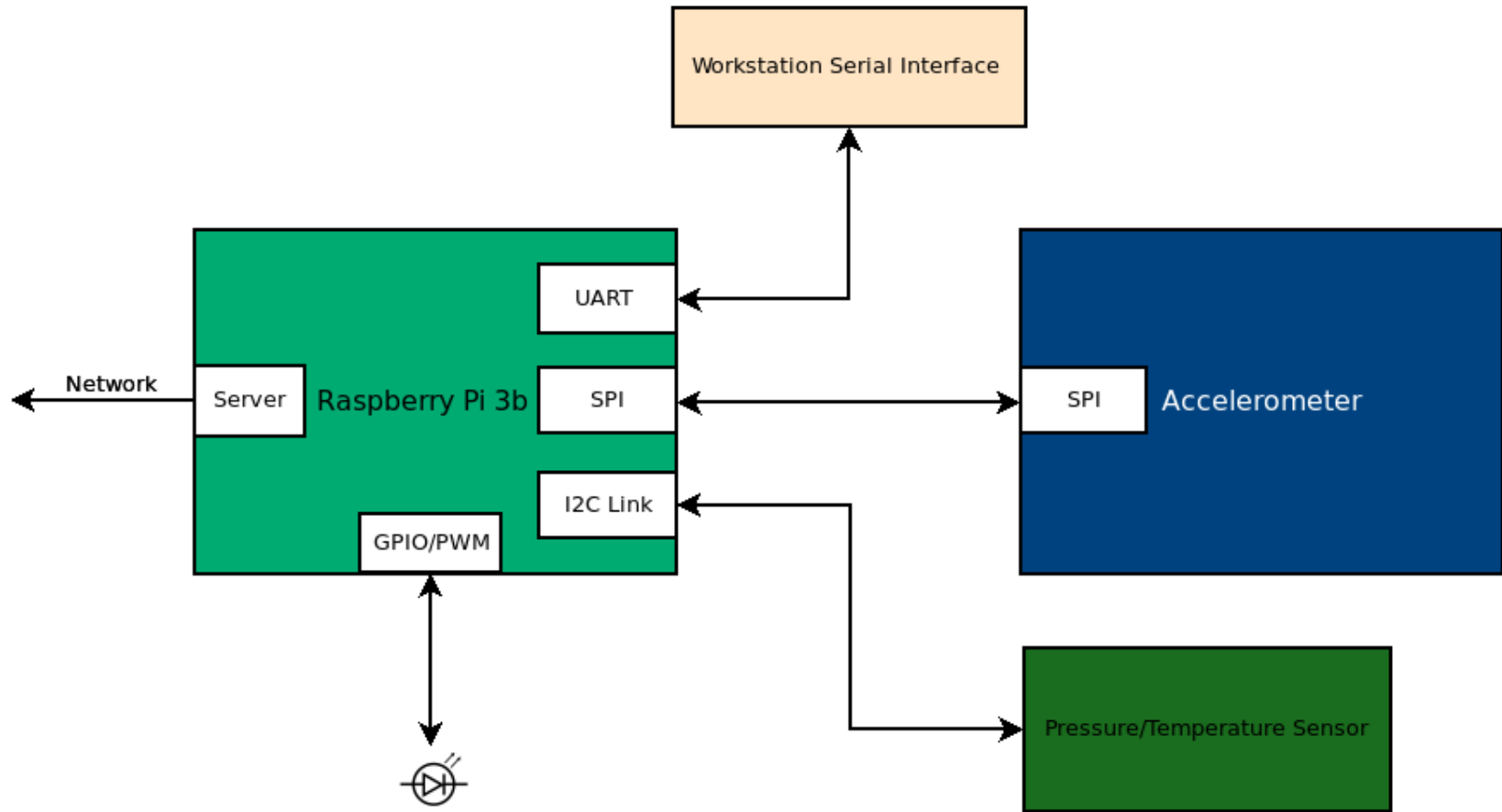
Flask

Bryce Himebaugh

Weekly Focus	Reading	Monday	Wed	Lab
CPS Intro/UART		1/10: CPS Introduction	1/12: Pi Intro/UART Bus	Project 0 Raspberry PI Setup
I2C Bus		1/17: MLK Day	1/19: I2C Bus Overview	Project 1 I2C Pressure/Temperature Sensor
I2C and SPI Bus		1/24: Pressure Sensor	1/26: SPI Bus Overview	Project 2 SPI Accelerometer
SPI/Networking		1/31: Accelerometer	2/2: MQTT	Project 3 MQTT Sensor Data Server
Networking		2/7: GPIO/LED	2/9: Flask	Project 4 Sensor LED Output
Web Server		2/14: No Class	2/16: CPS Wrapup, Exam Review	P5 Demultiplexer
Evaluation		2/21: Exam 1	2/23: CE Intro/ Logic	P6 ALU

<https://engr210.github.io/>





Flask Overview

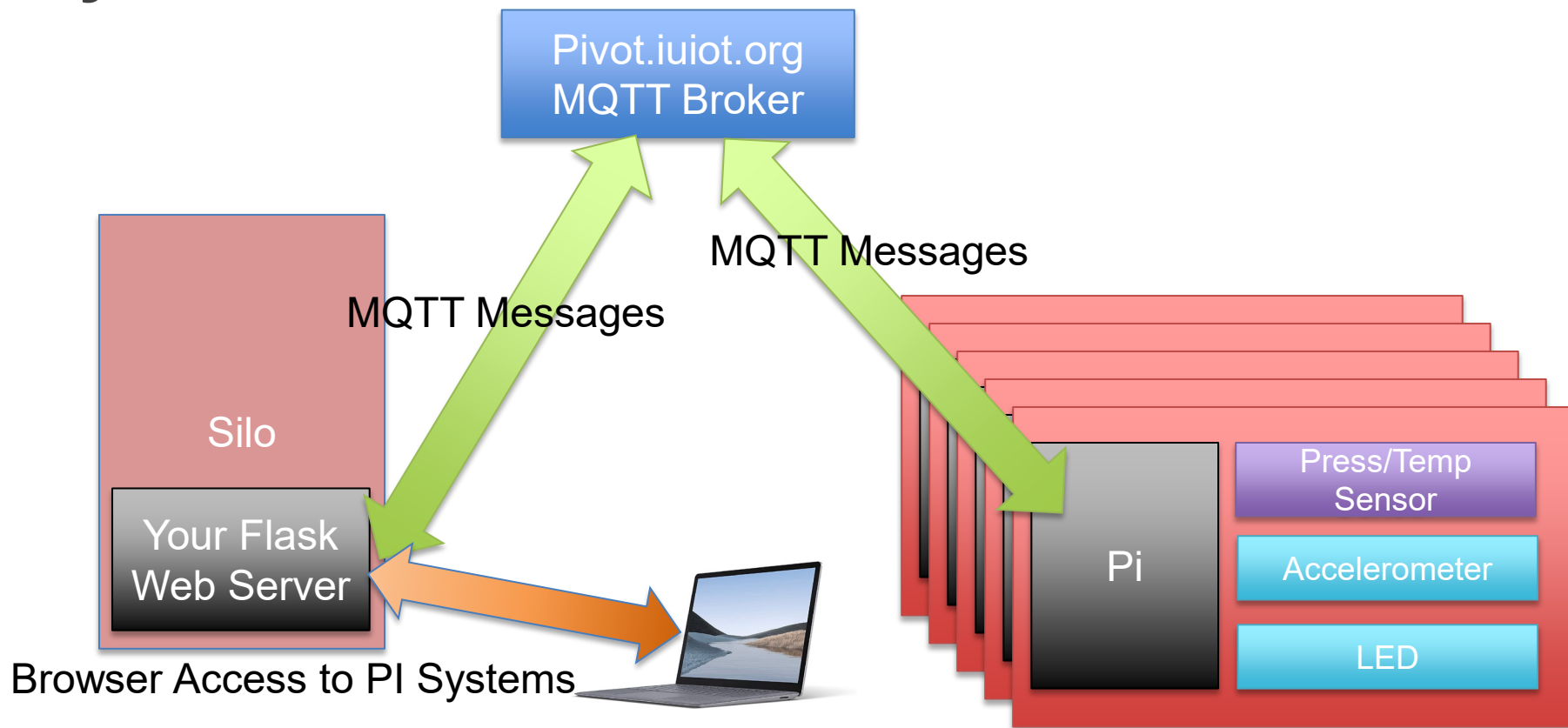
Flask



1. Python Micro Web Framework
 - Minimalist compared to Django Framework
2. First Release in 2010
3. Hammer vs toolbox
4. Used in projects as the API ...



System Architecture



Minimal Example

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```




```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```



```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```



```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```



```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
```

```
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```



Python Function Decorators

Functions

Python

>>>

```
>>> def add_one(number):  
...     return number + 1  
  
>>> add_one(2)  
3
```

<https://realpython.com/primer-on-python-decorators/#functions>



Functions can be Passed as Arguments

Python

```
def say_hello(name):  
    return f"Hello {name}"  
  
def be_awesome(name):  
    return f"Yo {name}, together we are the awesomest!"  
  
def greet_bob(greeter_func):  
    return greeter_func("Bob")
```

Python

>>>

```
>>> greet_bob(say_hello)  
'Hello Bob'  
  
>>> greet_bob(be_awesome)  
'Yo Bob, together we are the awesomest!'
```

<https://realpython.com/primer-on-python-decorators/#functions>



Functions within Functions (Inner Functions)

Python

```
def parent():
    print("Printing from the parent() function")

    def first_child():
        print("Printing from the first_child() function")

    def second_child():
        print("Printing from the second_child() function")

    second_child()
    first_child()
```

Python Traceback

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'first_child' is not defined
```

first_child and
second_child only exist
inside of parent

Python

```
>>> parent()
Printing from the parent() function
Printing from the second_child() function
Printing from the first_child() function
```

<https://realpython.com/primer-on-python-decorators/#functions>



Returning Functions

Python

```
def parent(num):  
    def first_child():  
        return "Hi, I am Emma"  
  
    def second_child():  
        return "Call me Liam"  
  
    if num == 1:  
        return first_child  
    else:  
        return second_child
```

Python

```
>>> first()  
'Hi, I am Emma'  
  
>>> second()  
'Call me Liam'
```

Can call these functions like
any other function

Python

```
>>> first = parent(1)  
>>> second = parent(2)  
  
>>> first  
<function parent.<locals>.first_child at 0x7f599f1e2e18>  
  
>>> second  
<function parent.<locals>.second_child at 0x7f599dad5268>
```

Function Reference Returned

<https://realpython.com/primer-on-python-decorators/#functions>



Decorators

Python

```
def my_decorator(func):  
    def wrapper():  
        print("Something is happening before the function is called.")  
        func()  
        print("Something is happening after the function is called.")  
    return wrapper  
  
def say_whee():  
    print("Whee!")  
  
say_whee = my_decorator(say_whee)
```

Python

Decorators wrap the original function

```
>>> say_whee()  
Something is happening before the function is called.  
Whee!  
Something is happening after the function is called.
```

<https://realpython.com/primer-on-python-decorators/#functions>



Dynamic Behavior

Python

```
from datetime import datetime

def not_during_the_night(func):
    def wrapper():
        if 7 <= datetime.now().hour < 22:
            func()
        else:
            pass # Hush, the neighbors are asleep
    return wrapper

def say_whee():
    print("Whee!")

say_whee = not_during_the_night(say_whee)
```

When called after bedtime ...

Python

```
>>> say_whee()
>>>
```

<https://realpython.com/primer-on-python-decorators/#functions>



@ symbol

Python

```
def my_decorator(func):  
    def wrapper():  
        print("Something is happening before the function is called.")  
        func()  
        print("Something is happening after the function is called.")  
    return wrapper
```

```
@my_decorator  
def say_whee():  
    print("Whee!")
```

Same as `say_whee = my_decorator(say_whee)`

<https://realpython.com/primer-on-python-decorators/#functions>



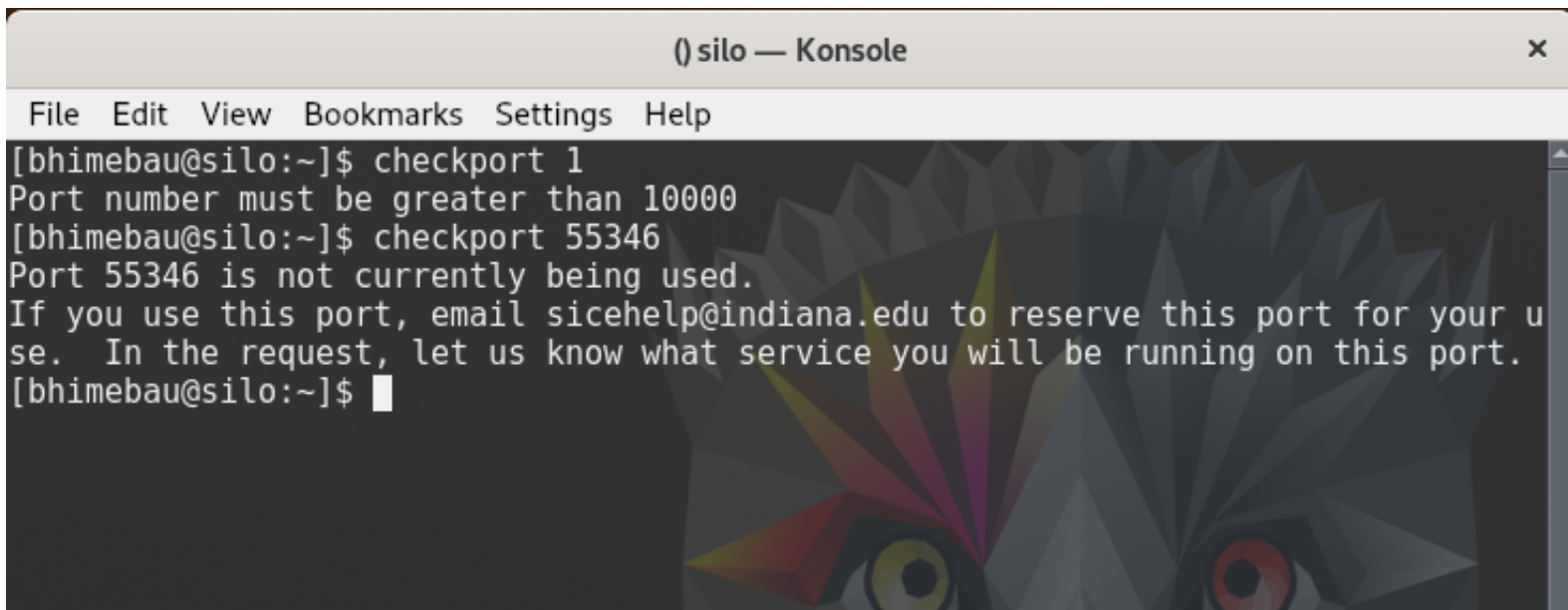
Back to the Example ...

```
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
    return 'Hello world'
if __name__ == '__main__':
    app.run(debug=True, port=55346, host='0.0.0.0')
```



Running the Code on Silo

Find an Open Port



```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~]$ checkport 1
Port number must be greater than 10000
[bhimebau@silo:~]$ checkport 55346
Port 55346 is not currently being used.
If you use this port, email sicehelp@indiana.edu to reserve this port for your use.
In the request, let us know what service you will be running on this port.
[bhimebau@silo:~]$
```



Screen Command

1. Versatile Linux Utility
2. Allows User Processes to Continue to Run after Terminal Closes
 - Detach and Reattach
3. Key Bindings: https://www.gnu.org/software/screen/manual/html_node/Default-Key-Bindings.html



Running Screen

```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$ screen
```

New Screen Process

```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$
```

Start Server

```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$ python3 helloWorld.py
* Serving Flask app "helloWorld" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
```

Detach with Server
Still Running

```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$ screen
[detached from 15382.pts-9.silo]
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$
```

C-a d

C-a C-d

(detach)

Detach screen from this terminal. See [Detach](#).



~ : bash — Konsole

File Edit View Bookmarks Settings Help

```
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$ screen
```

```
[detached from 15382.pts-9.silo]
```

```
[bhimebau@silo:~/forge/SICE-E210/flask_hello]$ exit
```

```
logout
```

```
Connection to silo closed.
```

```
bhimebau@orion:~$
```

silosice.indiana.edu:55346



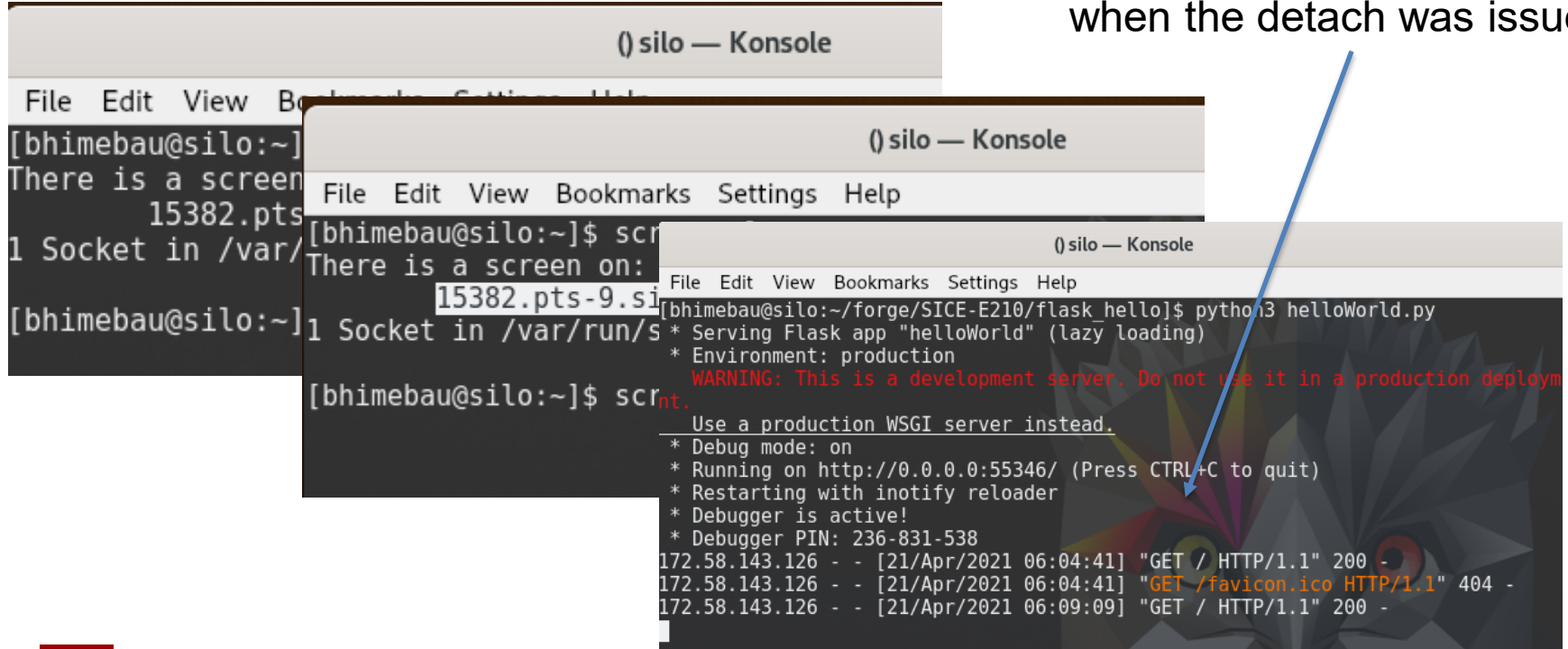
Not secure | http://silosice.indiana.edu:55346

Hello world



Reattaching

Right back where we left off
when the detach was issued



The image shows three overlapping terminal windows titled "() silo — Konsole".

The leftmost window shows a prompt `[bhimebau@silo:~]` and the command `screen -r 15382.pts-9.sil`. The output includes "There is a screen on:" and "1 Socket in /var/run/screen".

The middle window shows the same prompt and command, with the output "There is a screen on:" and "1 Socket in /var/run/screen".

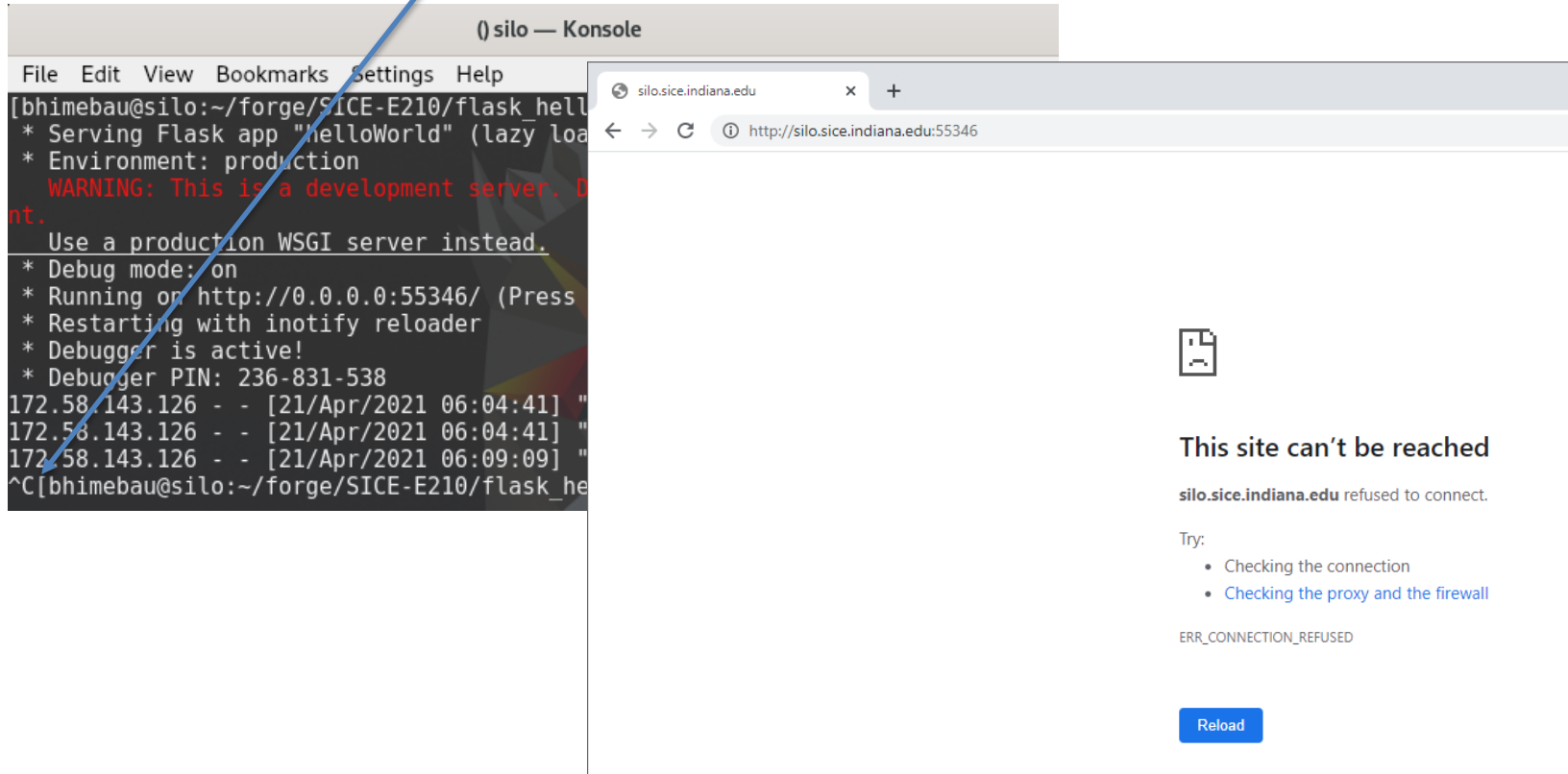
The rightmost window shows the prompt `[bhimebau@silo:~/forge/SICE-E210/flask hello]$` and the command `python3 helloWorld.py`. The output includes:

```
* Serving Flask app "helloWorld" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:55346/ (Press CTRL+C to quit)
* Restarting with inotify reloader
* Debugger is active!
* Debugger PIN: 236-831-538
172.58.143.126 - - [21/Apr/2021 06:04:41] "GET / HTTP/1.1" 200 -
172.58.143.126 - - [21/Apr/2021 06:04:41] "GET /favicon.ico HTTP/1.1" 404 -
172.58.143.126 - - [21/Apr/2021 06:09:09] "GET / HTTP/1.1" 200 -
```

A blue arrow points from the text "Right back where we left off when the detach was issued" to the rightmost terminal window.



CTRL-C to stop



The image shows a terminal window on the left and a web browser on the right. A blue arrow points from the text 'CTRL-C to stop' to the terminal window.

Terminal Window:

```
() silo — Konsole
File Edit View Bookmarks Settings Help
[bhimebau@silo:~/forge/SICE-E210/flask_hell
* Serving Flask app "helloWorld" (lazy loa
* Environment: production
WARNING: This is a development server. D
nt.
Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:55346/ (Press
* Restarting with inotify reloader
* Debugger is active!
* Debugger PIN: 236-831-538
172.58.143.126 - - [21/Apr/2021 06:04:41] "
172.58.143.126 - - [21/Apr/2021 06:04:41] "
172.58.143.126 - - [21/Apr/2021 06:09:09] "
^C[bhimebau@silo:~/forge/SICE-E210/flask_he
```

Web Browser:

The browser shows the address bar with `http://silo.sice.indiana.edu:55346`. The page content displays a sad face icon and the message:

This site can't be reached

`silo.sice.indiana.edu` refused to connect.

Try:

- Checking the connection
- [Checking the proxy and the firewall](#)

ERR_CONNECTION_REFUSED

[Reload](#)



System Architecture

