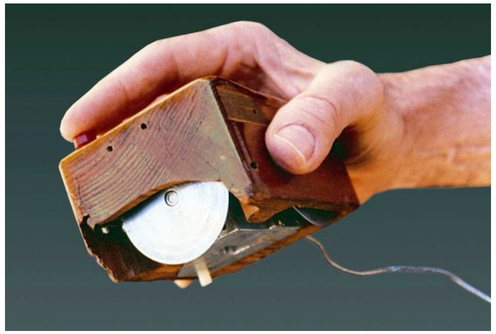
Virtual Reality Haptic Interactions (虛擬實境與觸覺回饋互動)

Lecturer: Ray
Week 2 (9/18)

CLI to GUI



[Engelbart Mouse, 1964]

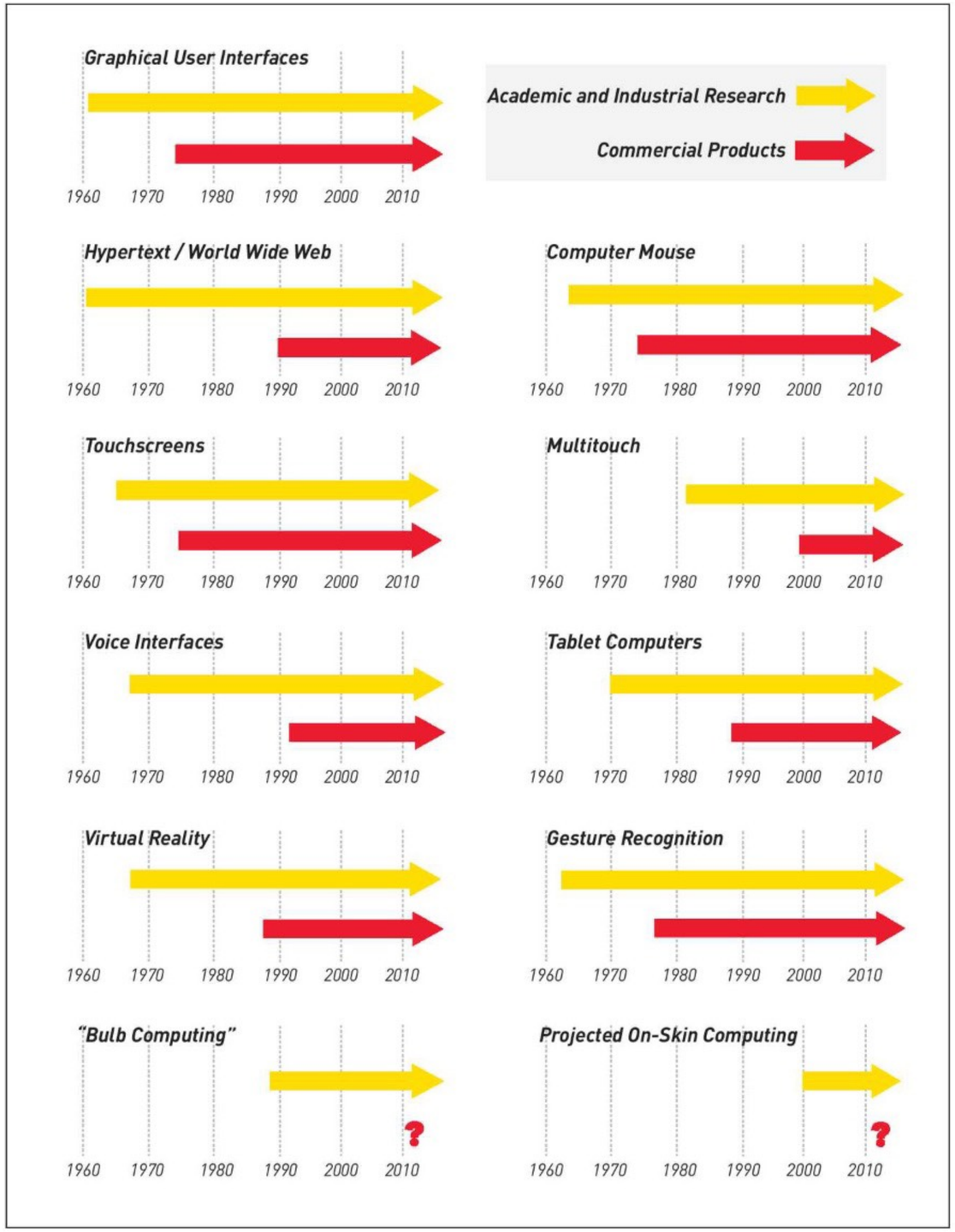
Command Line Interface (CLI)

Graphic User Interface (GUI)
[Macintosh Mouse, 1984]

The HCI Innovator's Dilemma

Chris Harrison, 2018. ACM Interactions.

https://dl.acm.org/citation.cfm?id=3274564



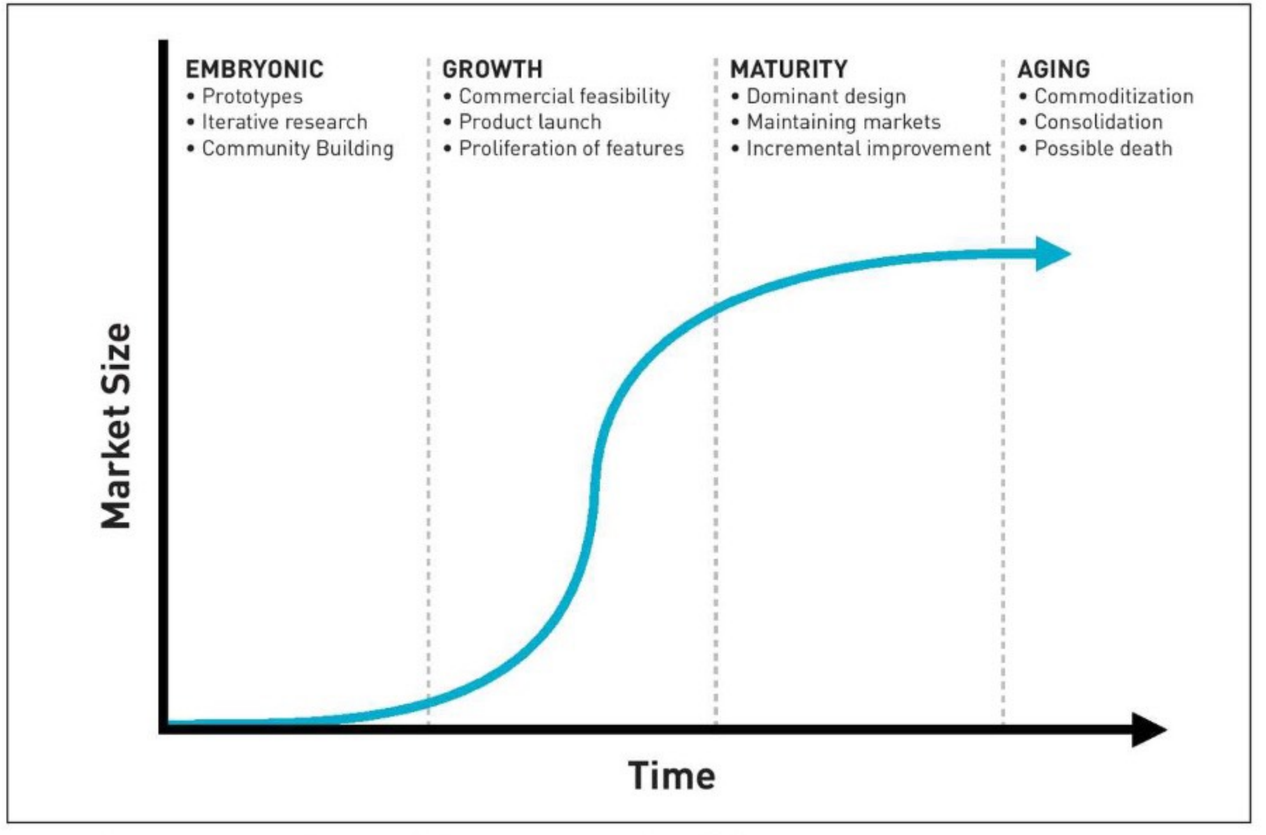


Figure 1. Technology Lifecycle S-Curve, adapted from [1].

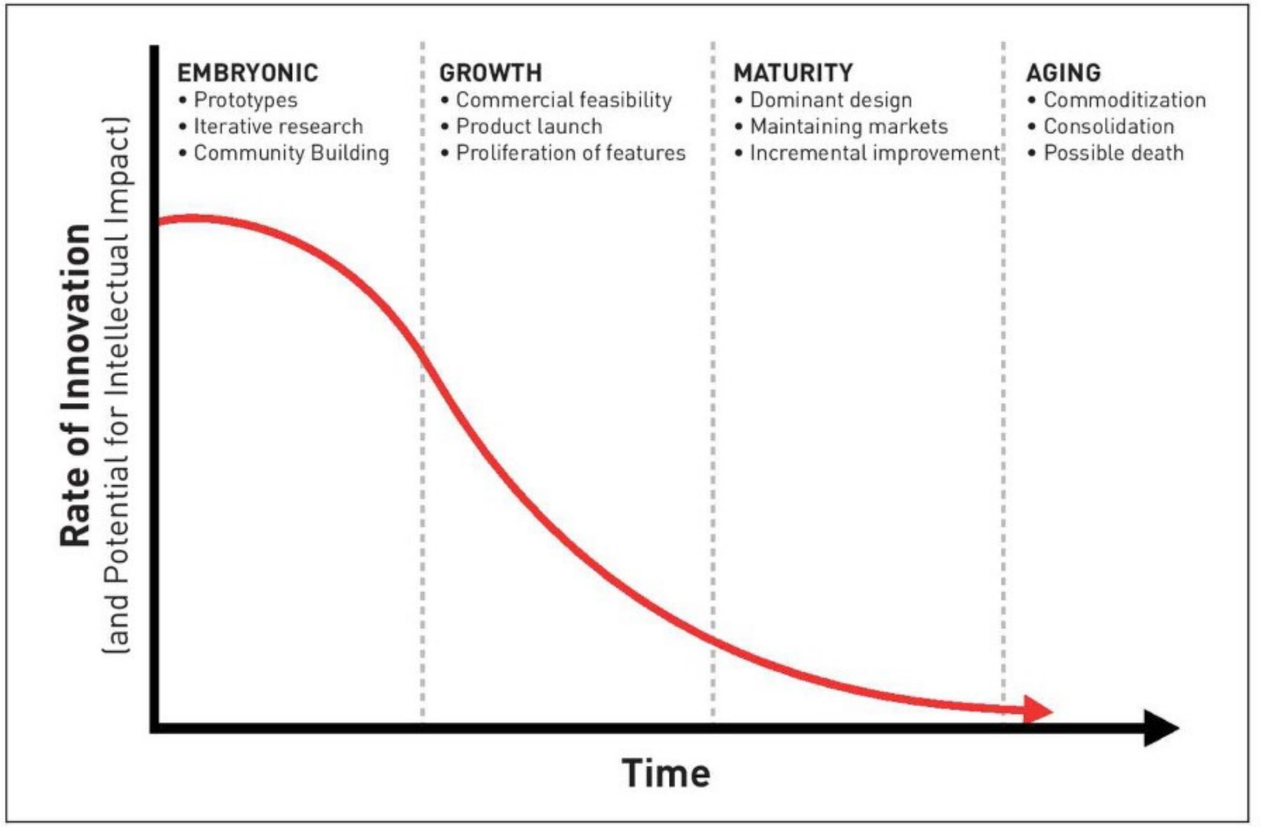


Figure 2. Innovation rate over time, adapted from [2], overlaid approximately onto our earlier technology lifecycle epochs.

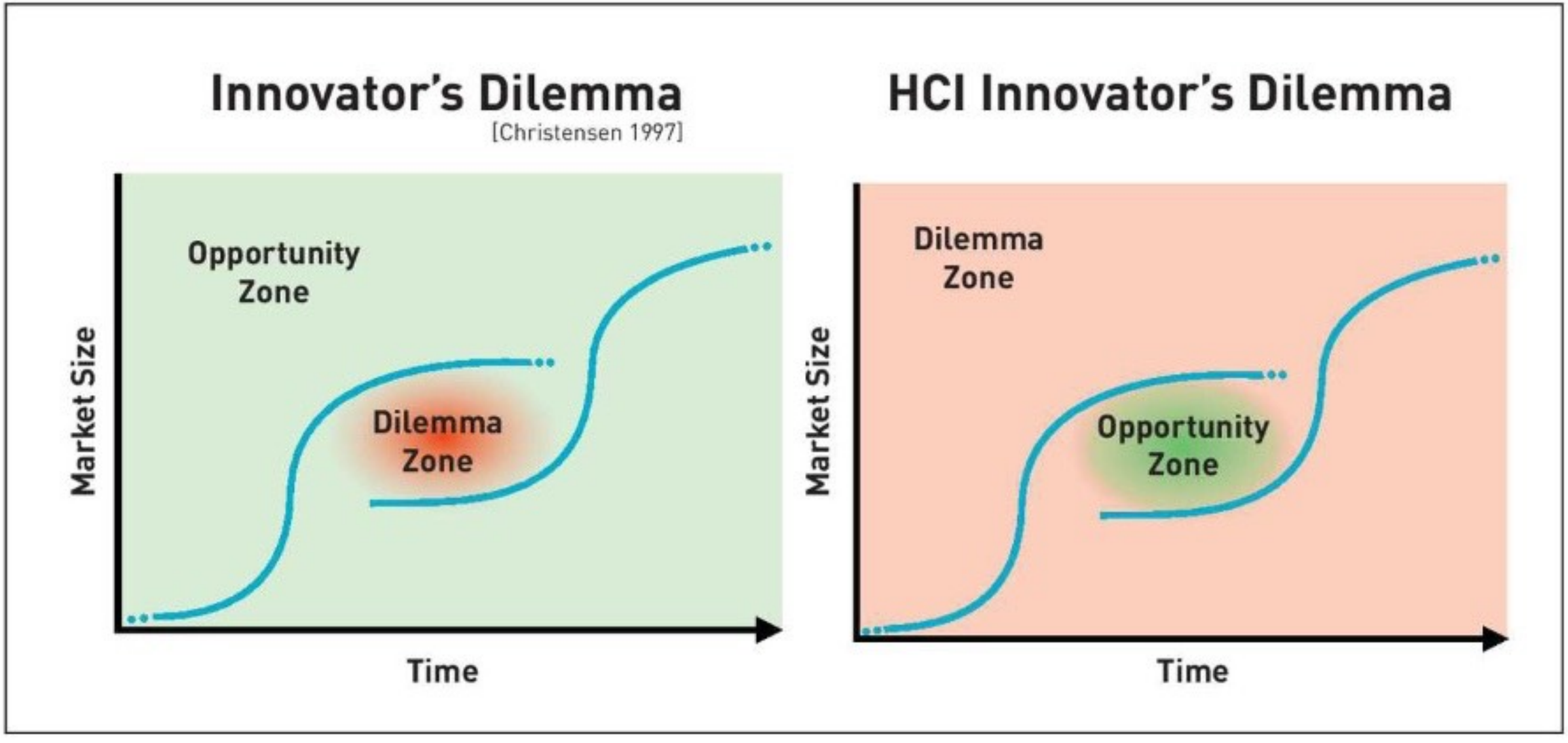


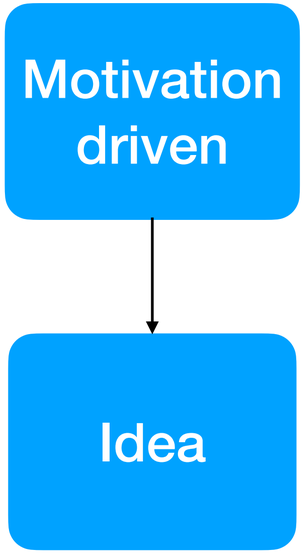
Figure 6. Christensen's Innovator's Dilemma [10] versus the HCI Innovator's Dilemma.

How is an HCI project performed?

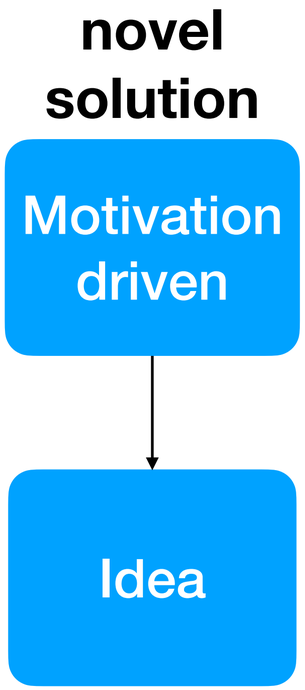
HCI Research Flow

Idea

HCI Research Flow

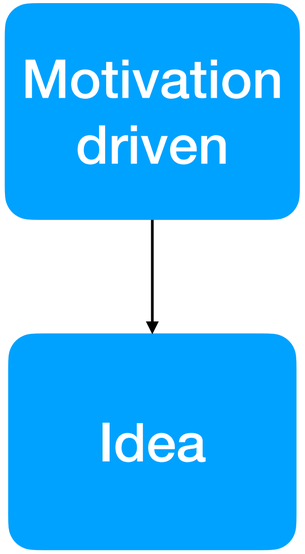


HCI Research Flow



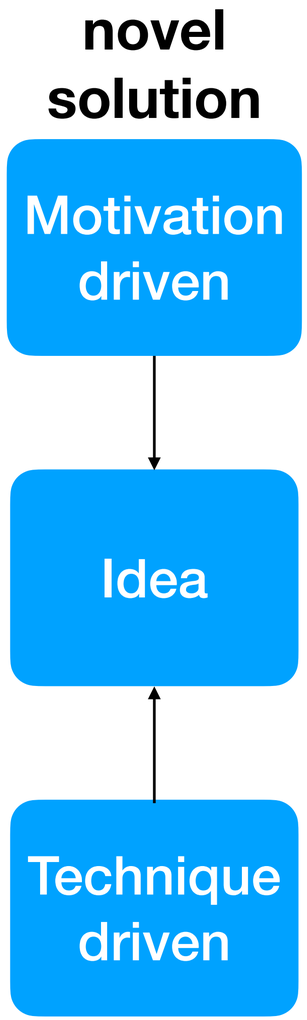
HCI Research Flow

novel solution

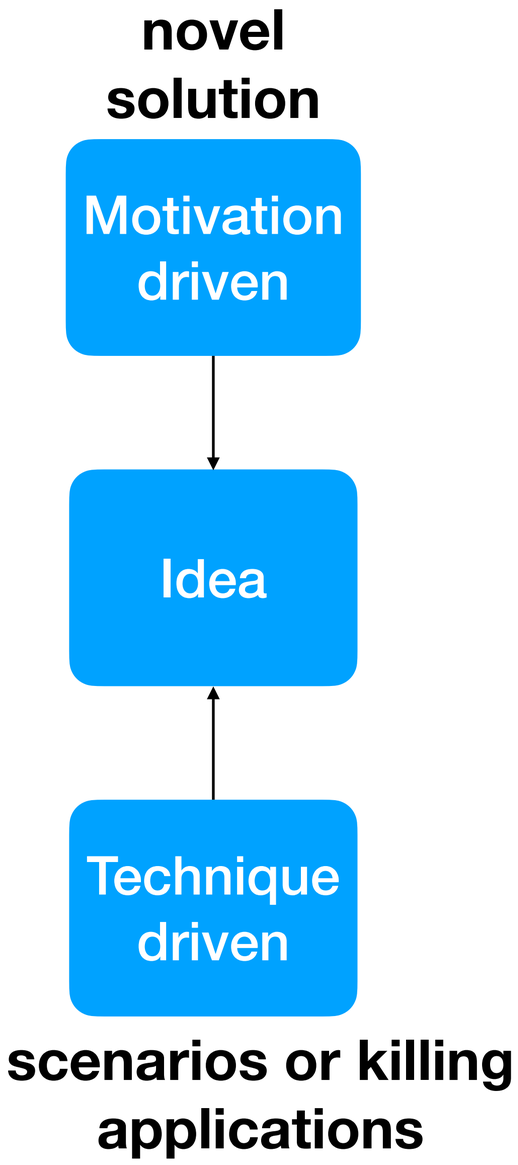


Common challenges:
similar solutions for a well-known problem
small performance improvement

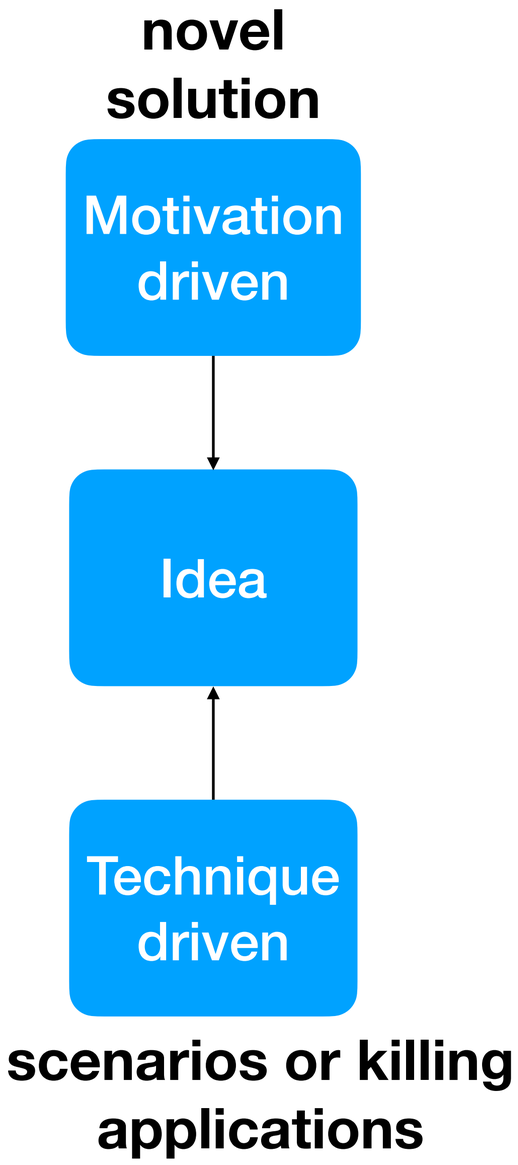
HCI Research Flow



HCI Research Flow



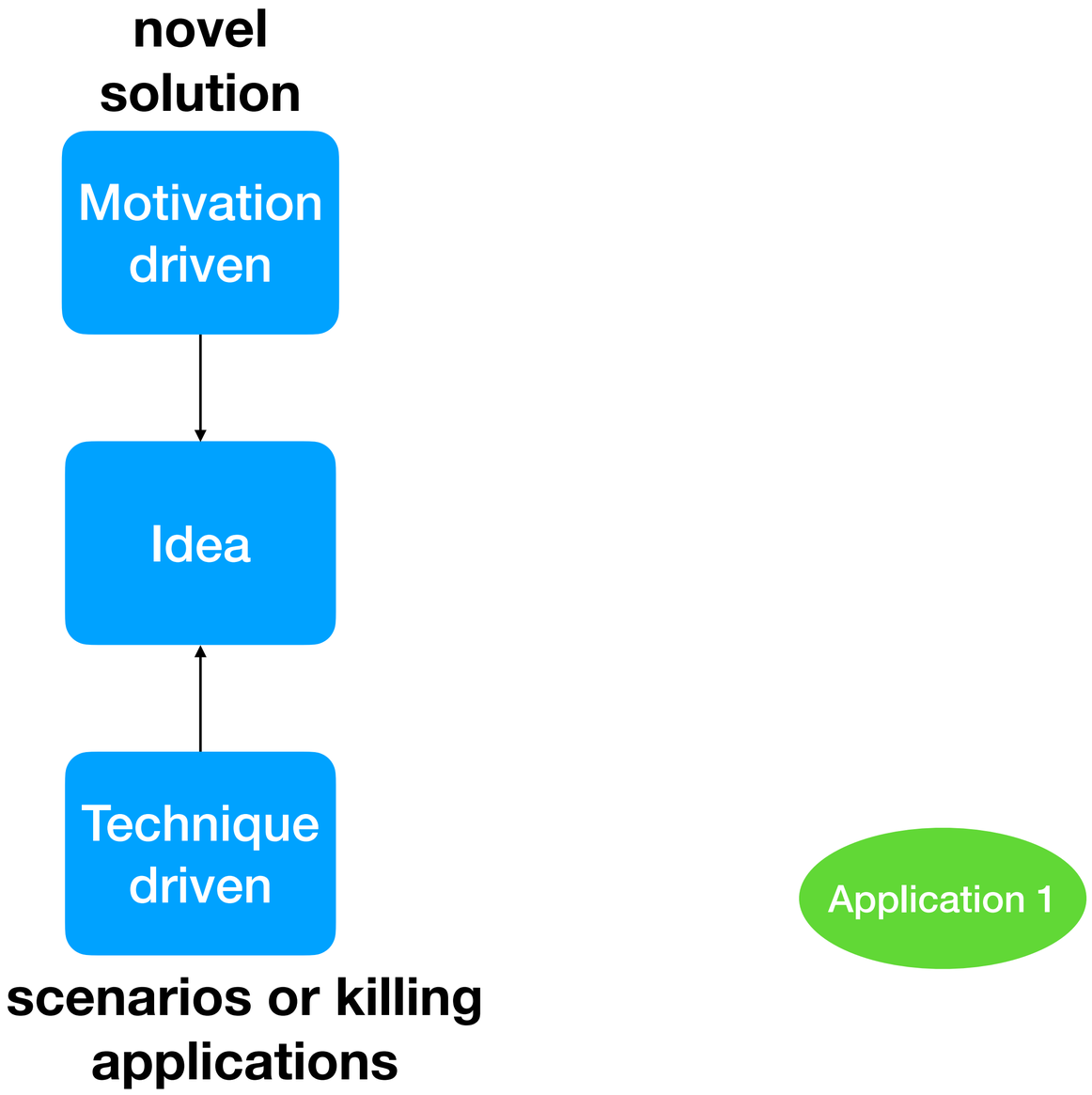
HCI Research Flow



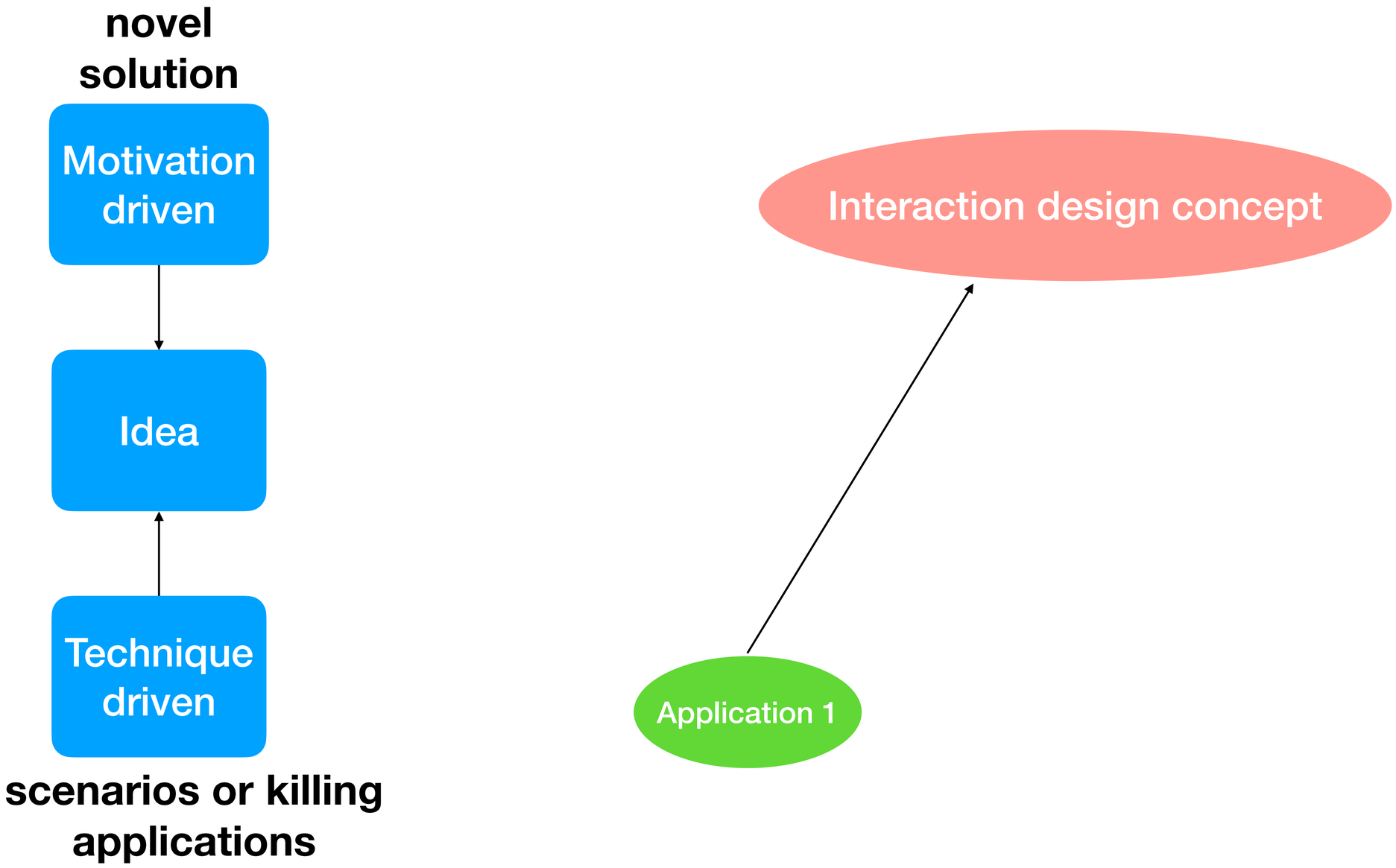
Common challenges:

- **Cool technique with poor motivation**
  why using this?
  alternative solution
- **Imitation revision**
  similar interactions

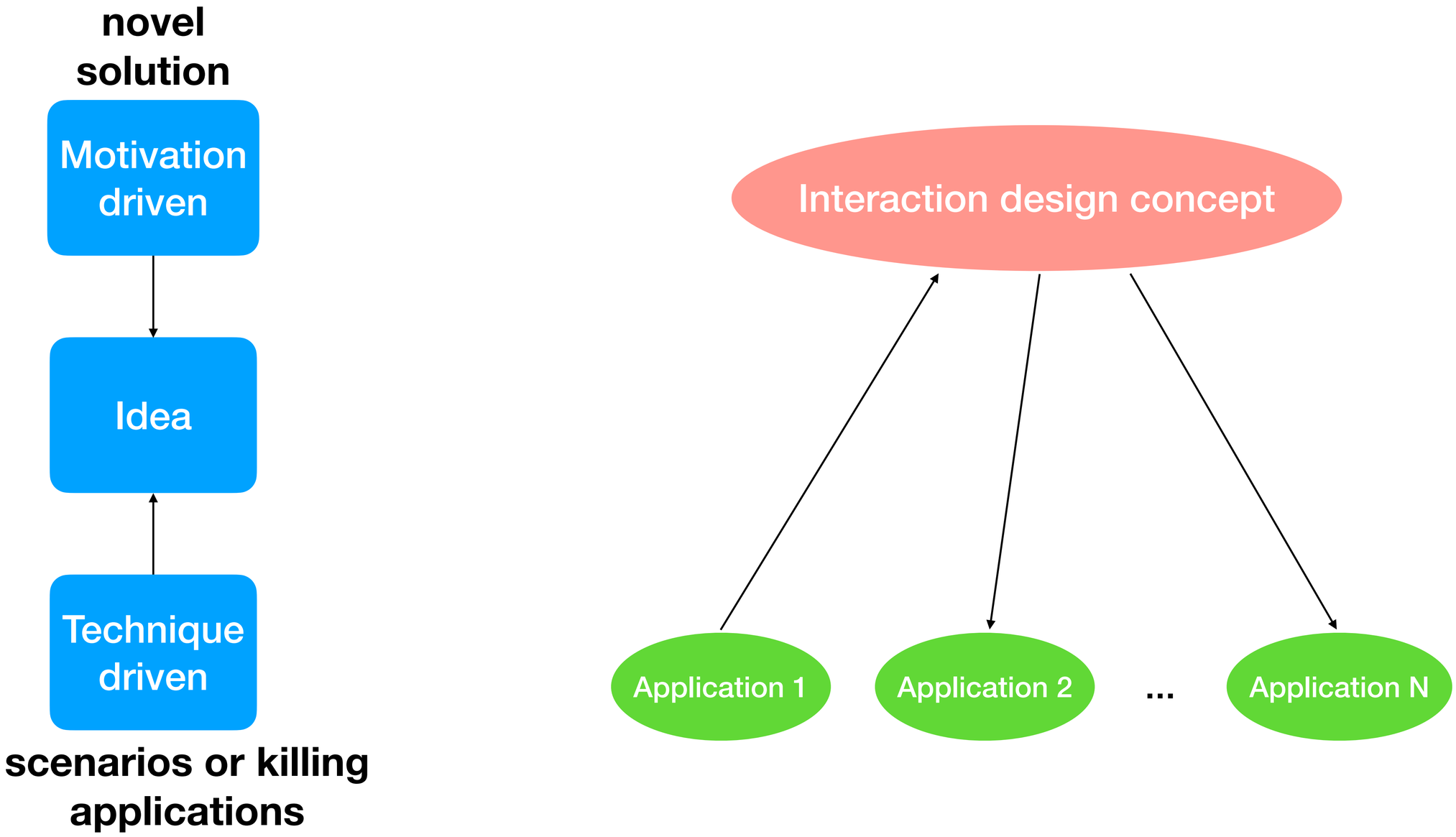
HCI Research Flow



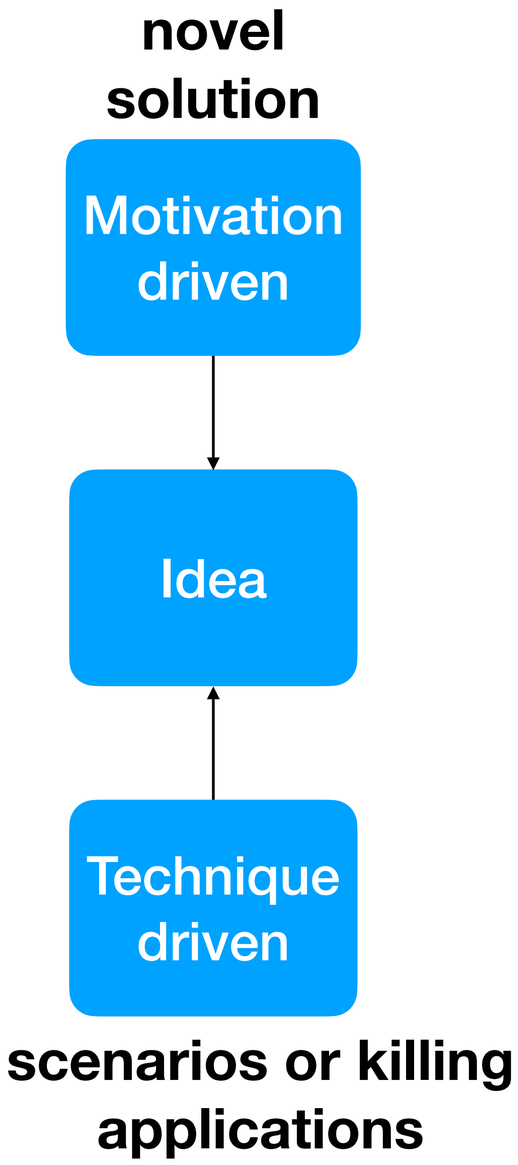
HCI Research Flow

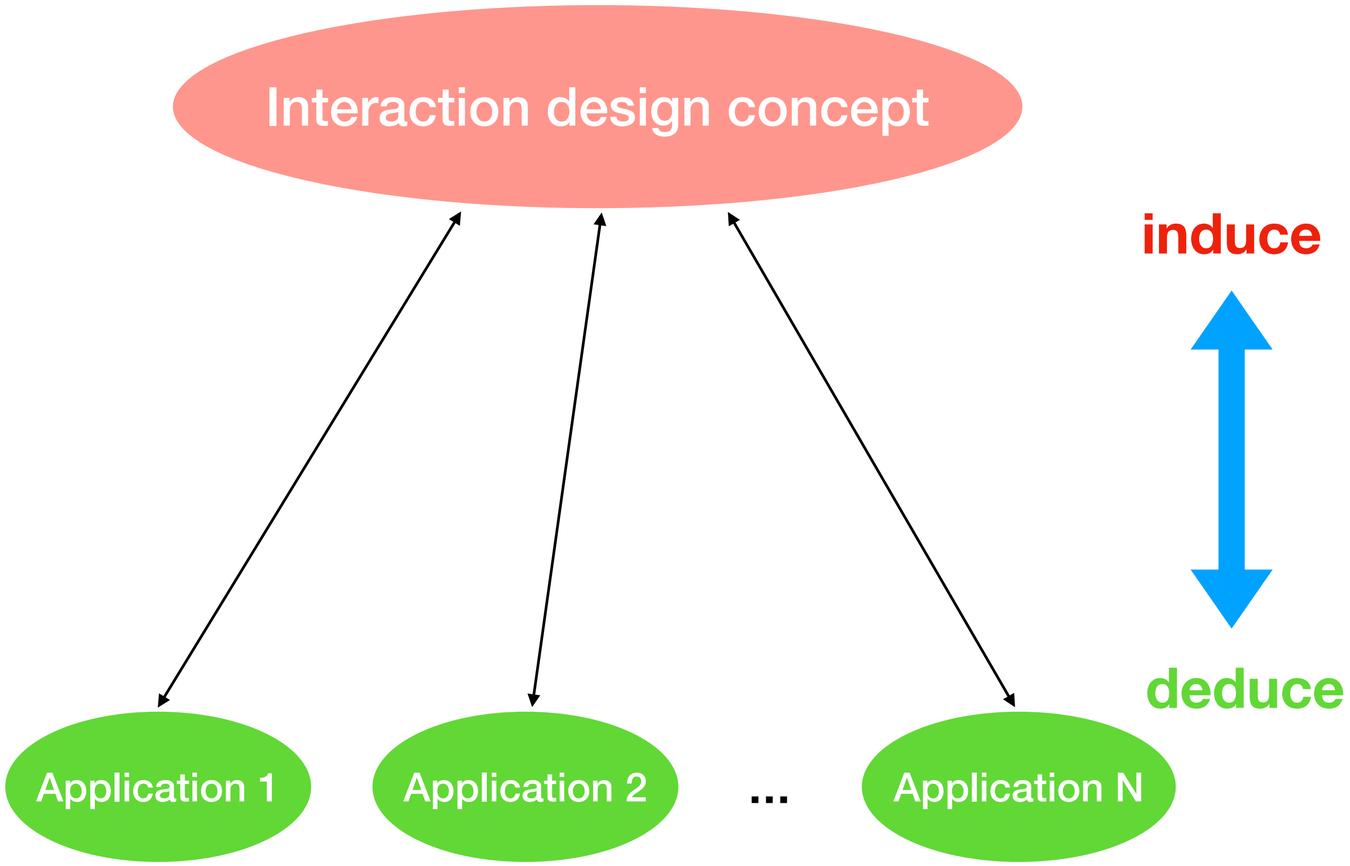


HCI Research Flow

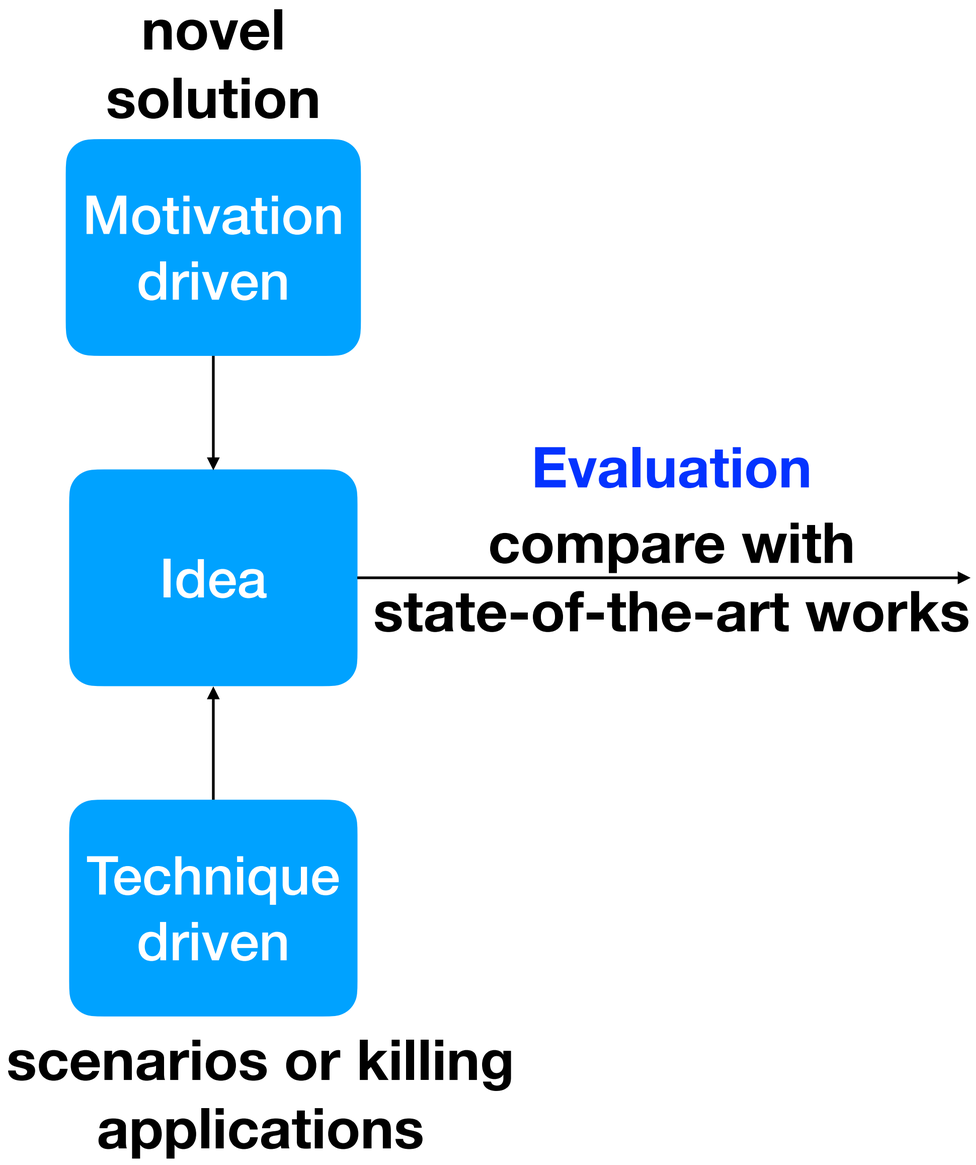


HCI Research Flow

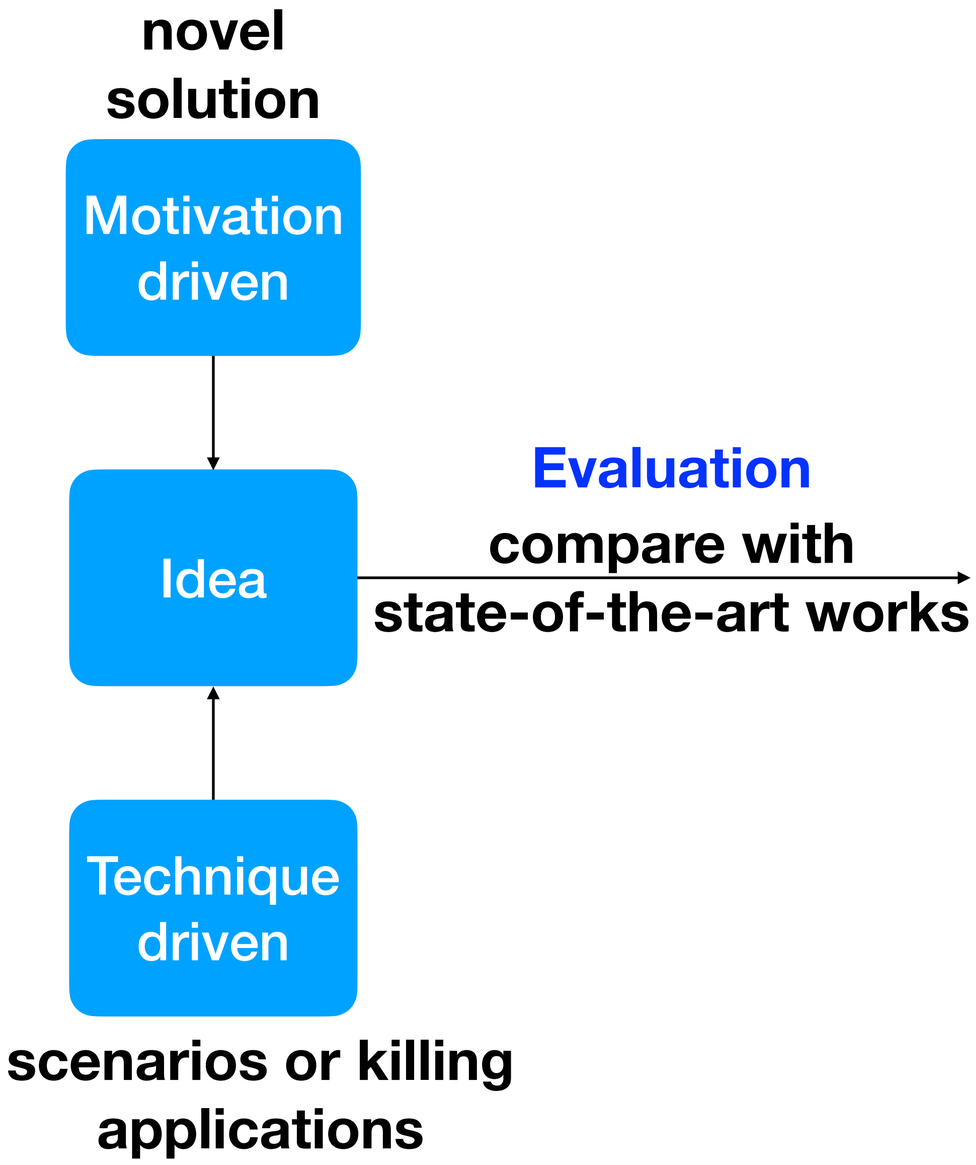


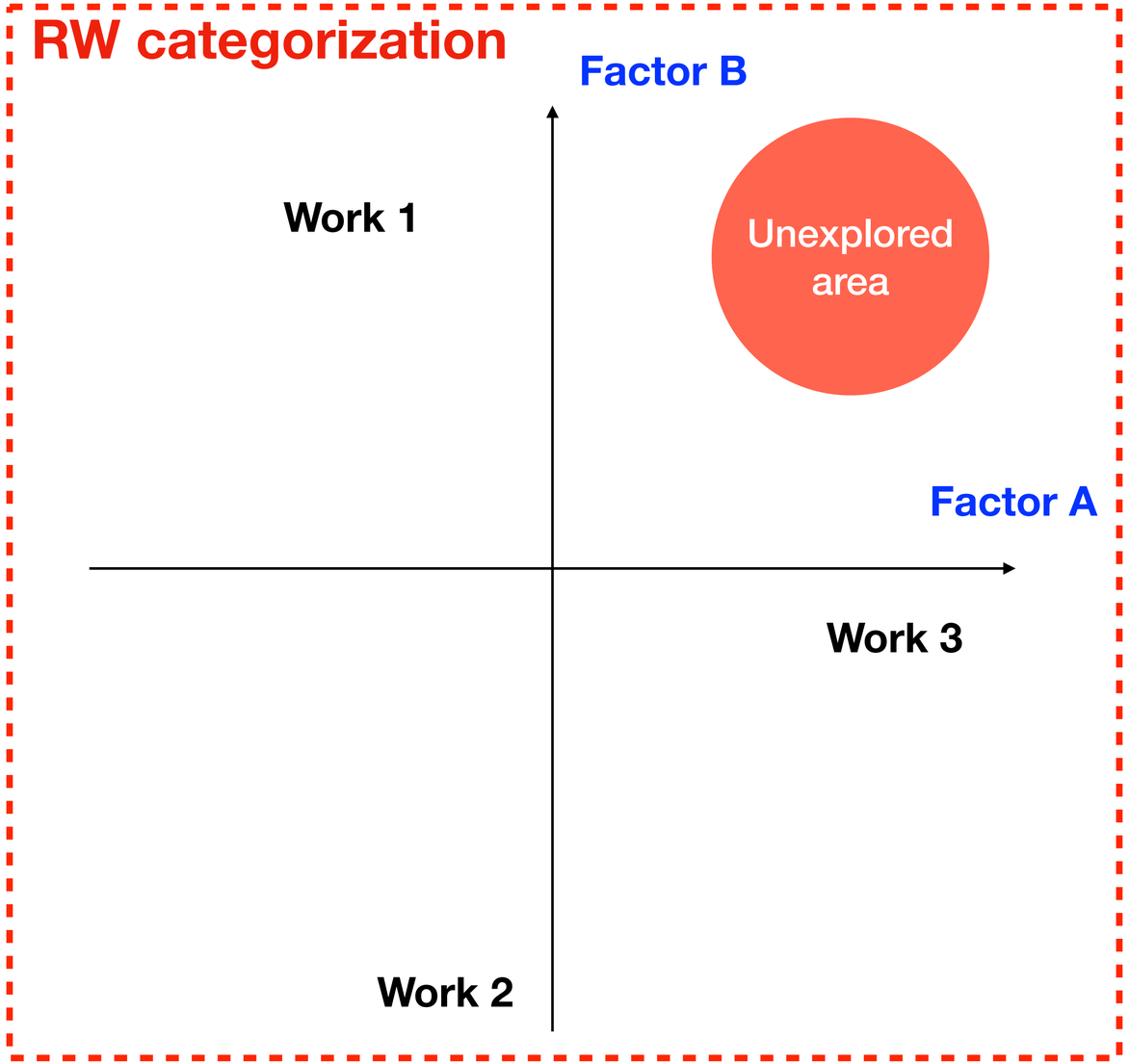


HCI Research Flow

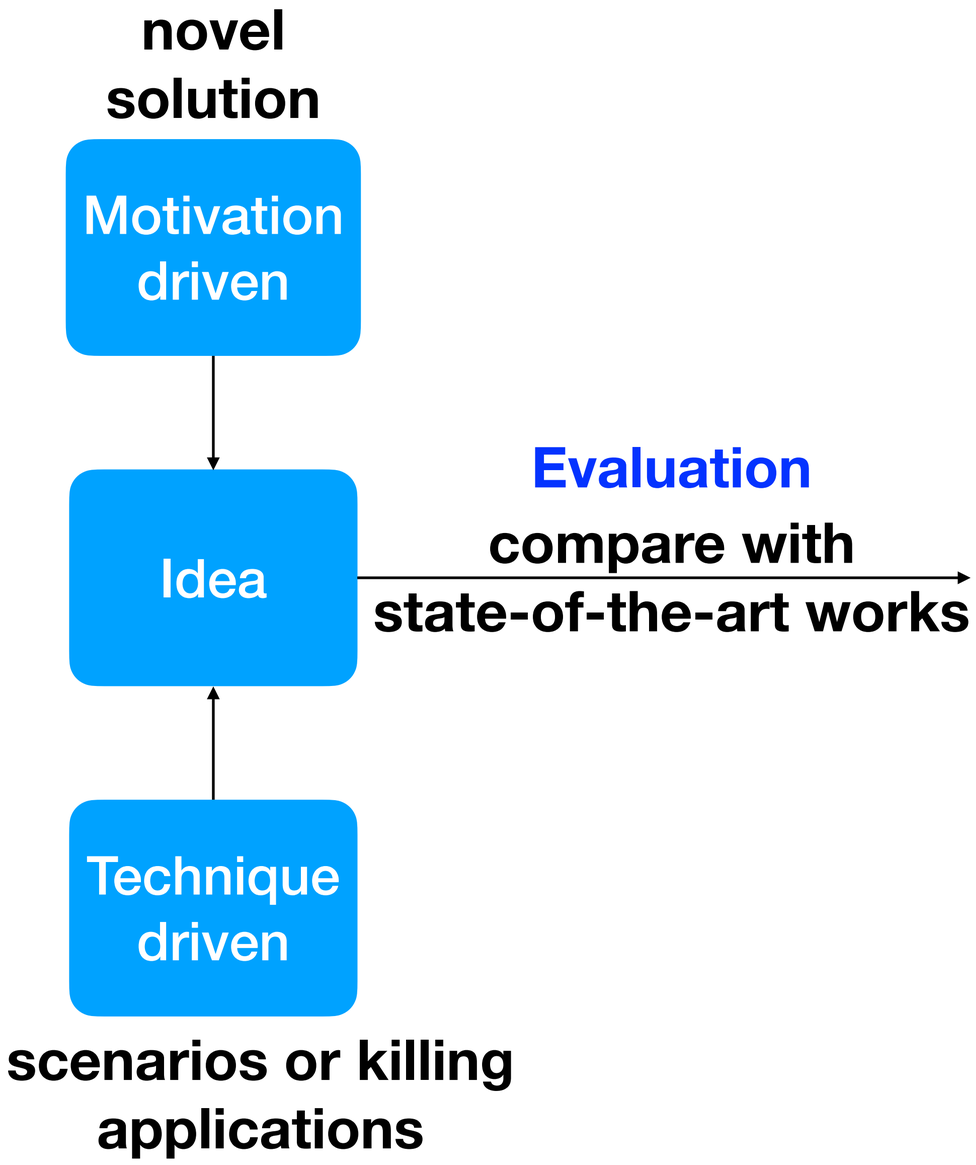


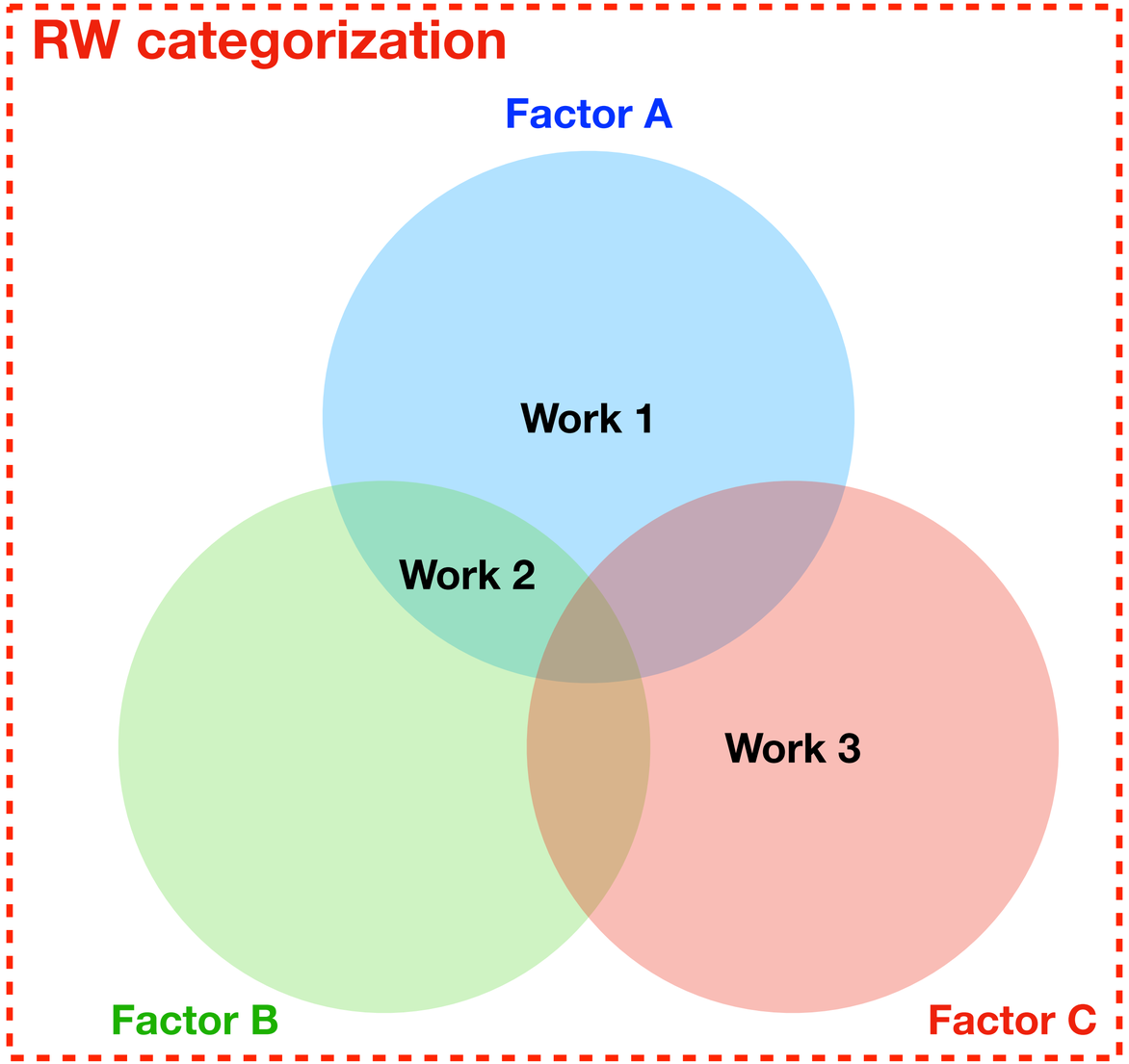
HCI Research Flow





HCI Research Flow

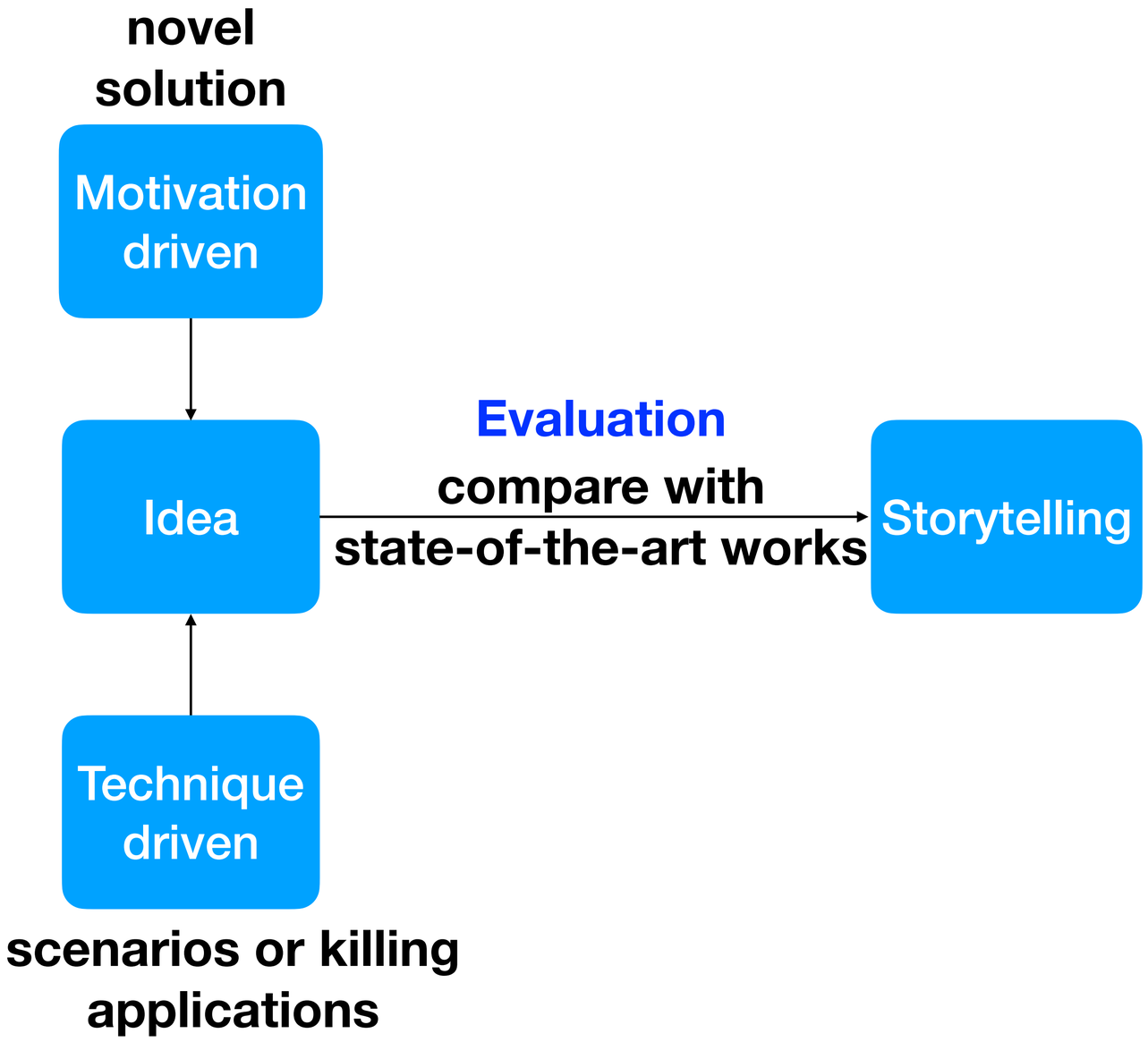




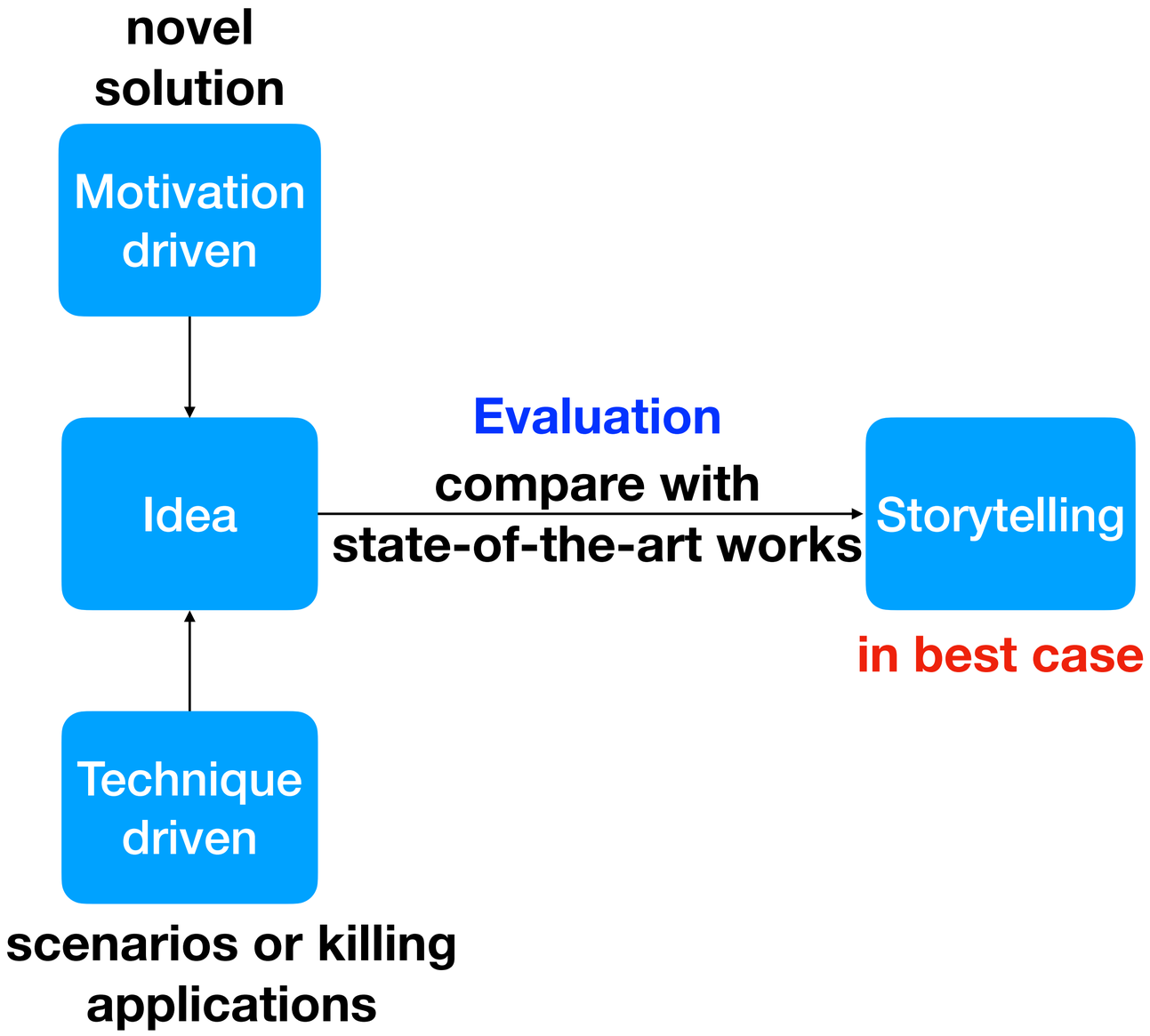
Evaluating a Work

- Novel idea (innovative, insightful)
- Technical contribution (novel techniques, difficult to implement)
- Complete studies (deep investigating, insightful results)

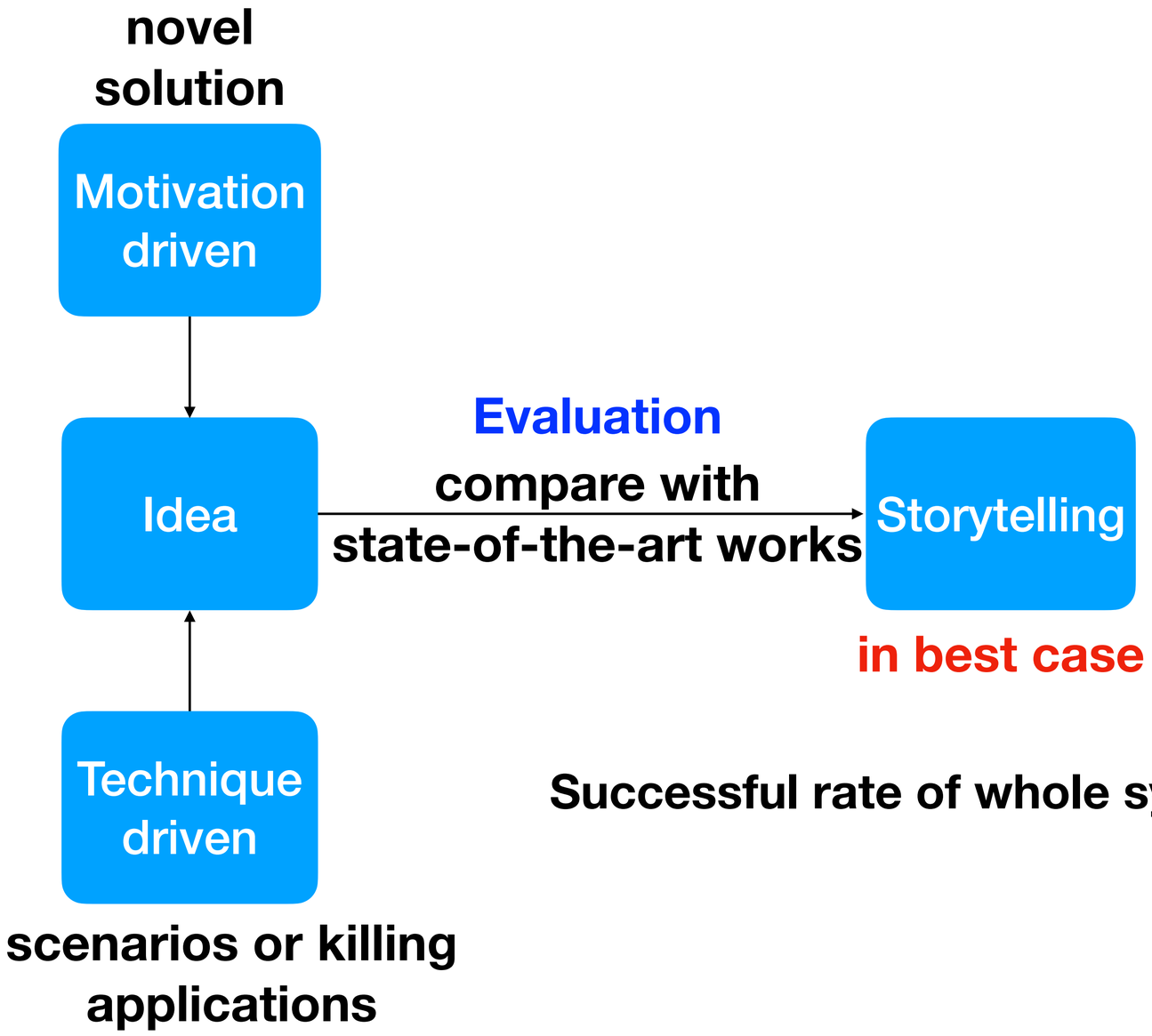
HCI Research Flow



HCI Research Flow

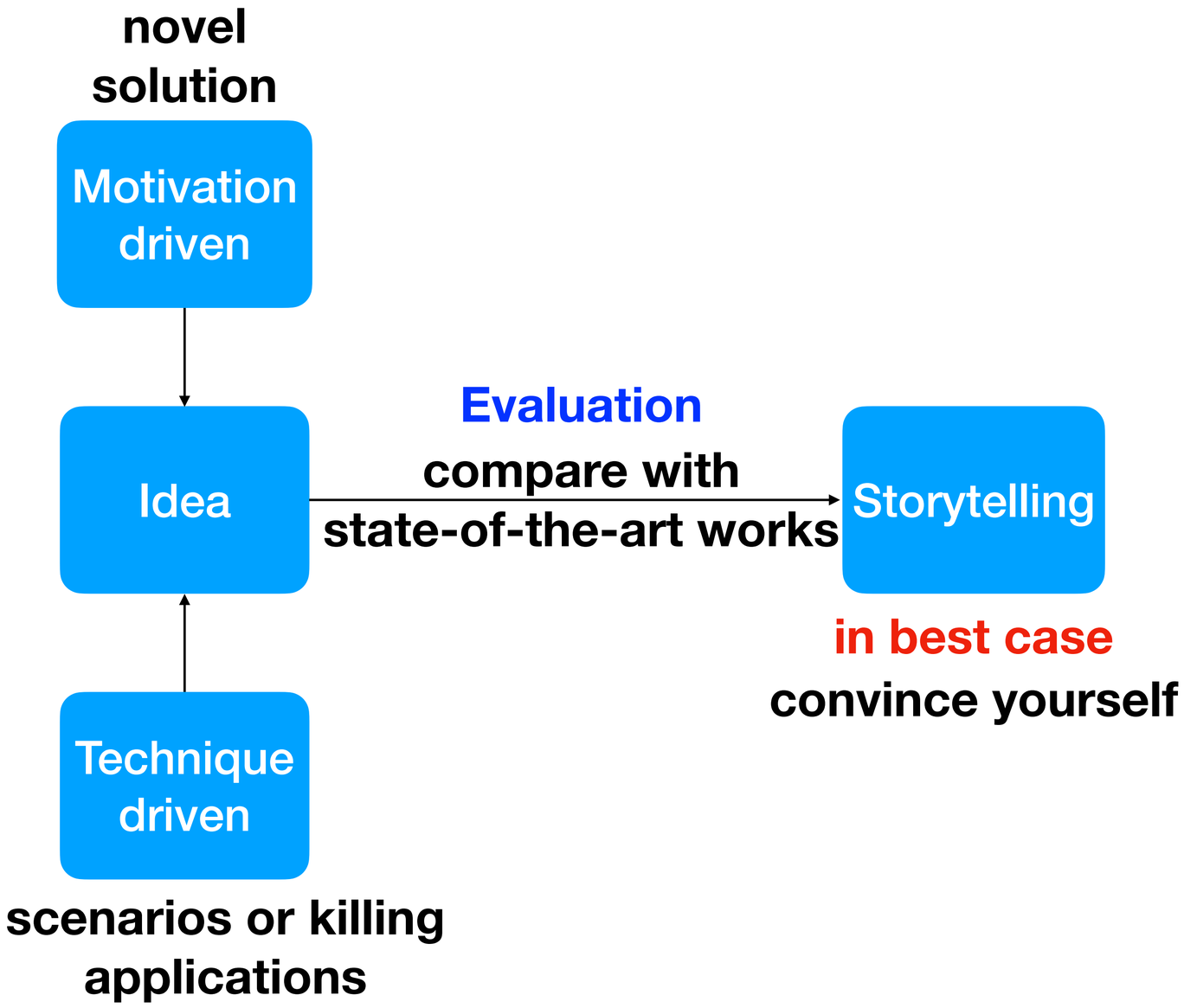


HCI Research Flow

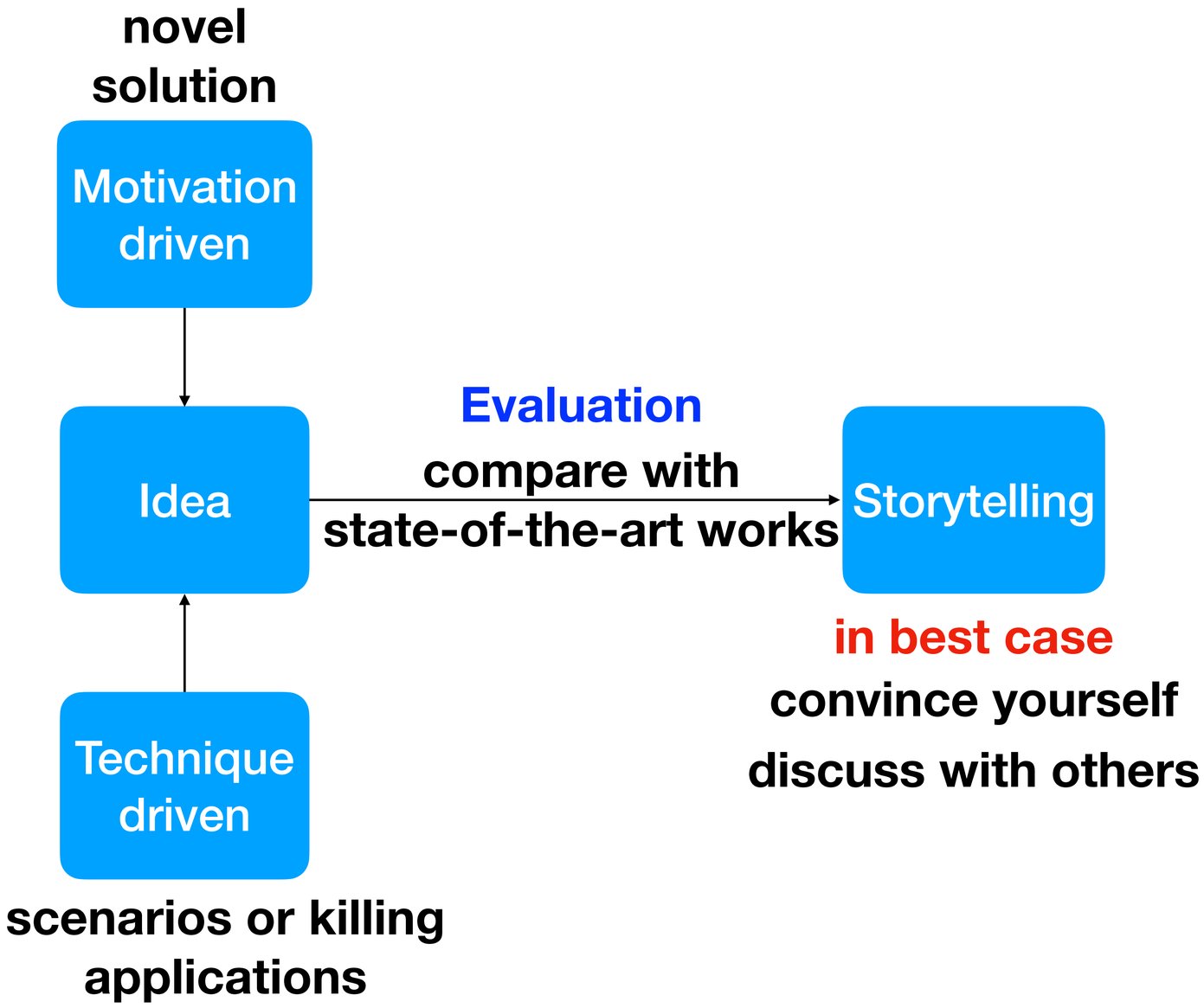


Successful rate of whole system $S = S_1 \times S_2 \times \ldots \times S_n$ (all parts)

HCI Research Flow



HCI Research Flow



HCI Research Flow

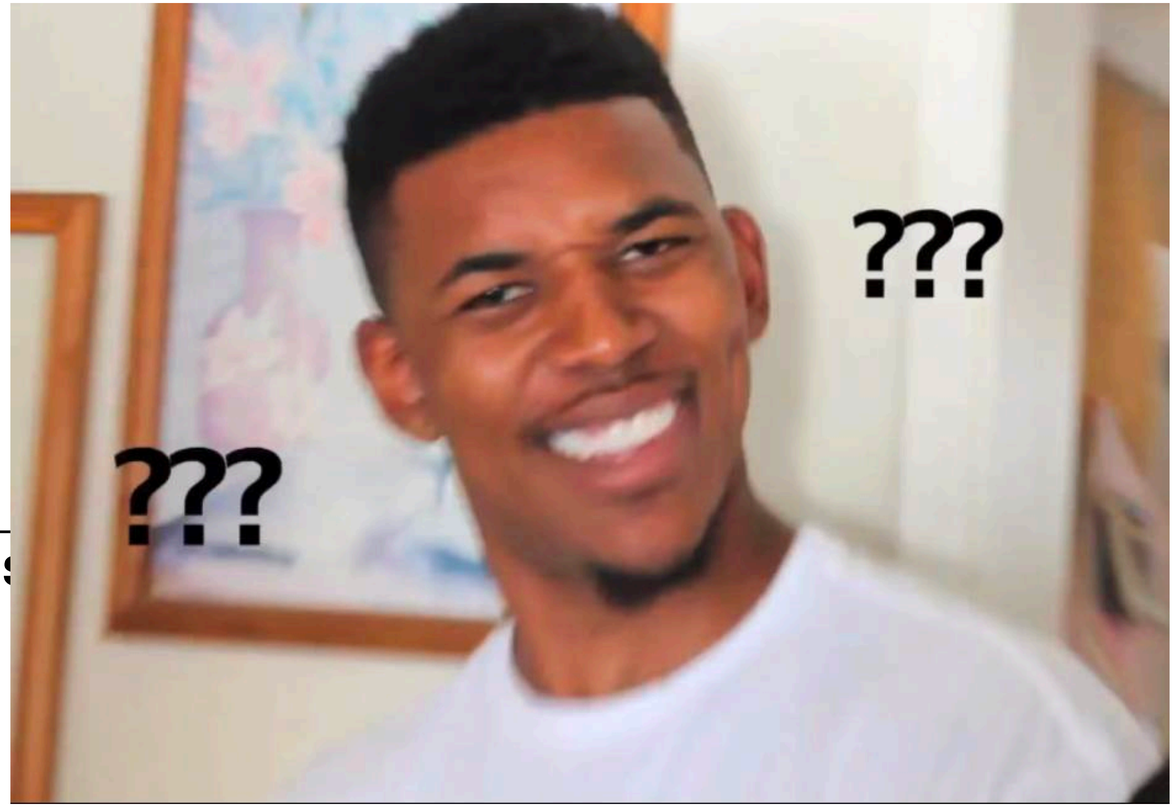
novel solution

Motivation driven

Idea

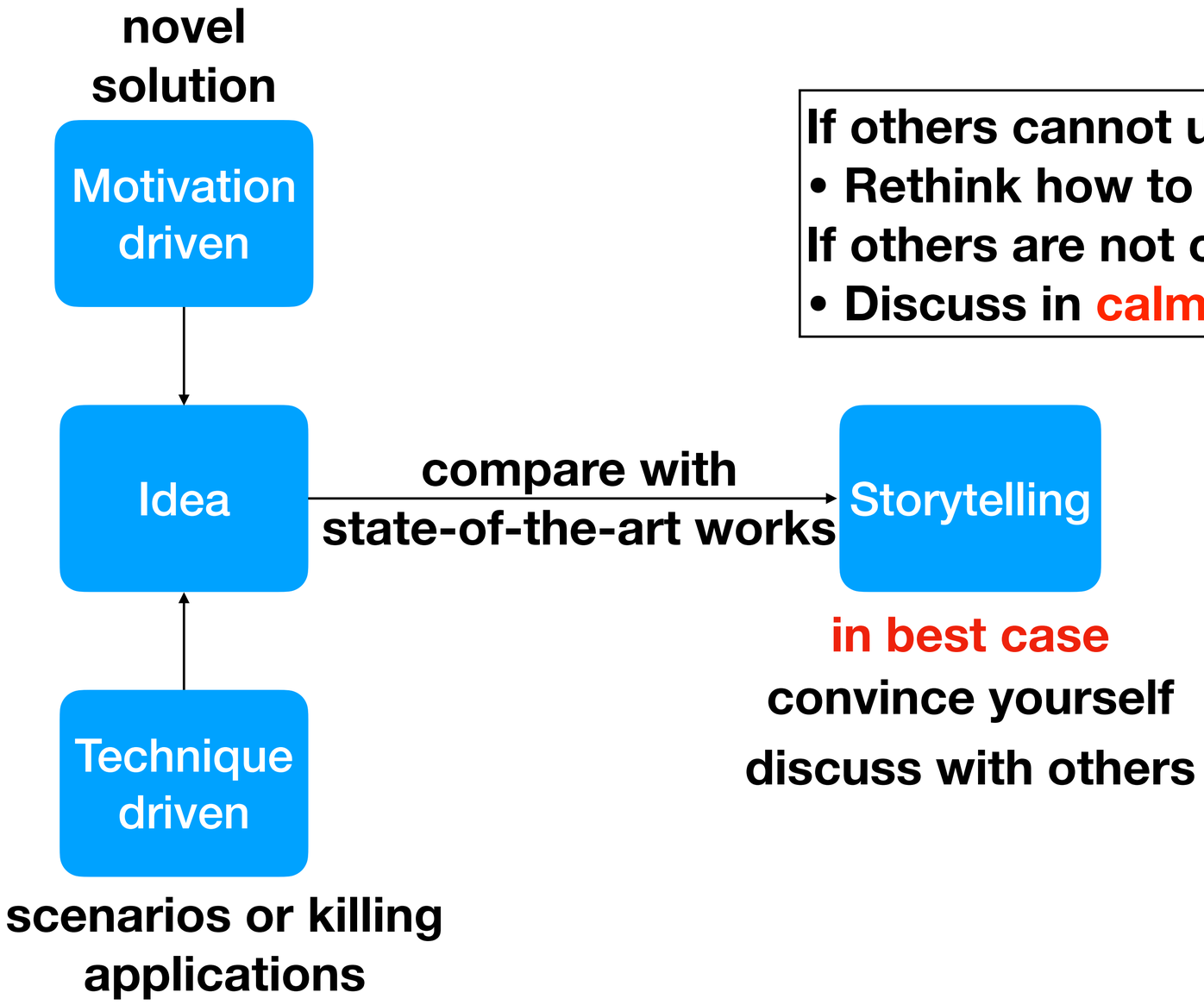
Technique driven

scenarios or killing applications



discuss with others

HCI Research Flow



If others cannot understand the story
- Rethink how to organize and tell the story

If others are not convinced
- Discuss in calm and rational

HCI Research Flow

novel solution

Motivation d

Tec driven

scenarios or killing applications

If others cannot understand the story

- Rethink how to organize and tell the story

HCI Research Flow

novel solution

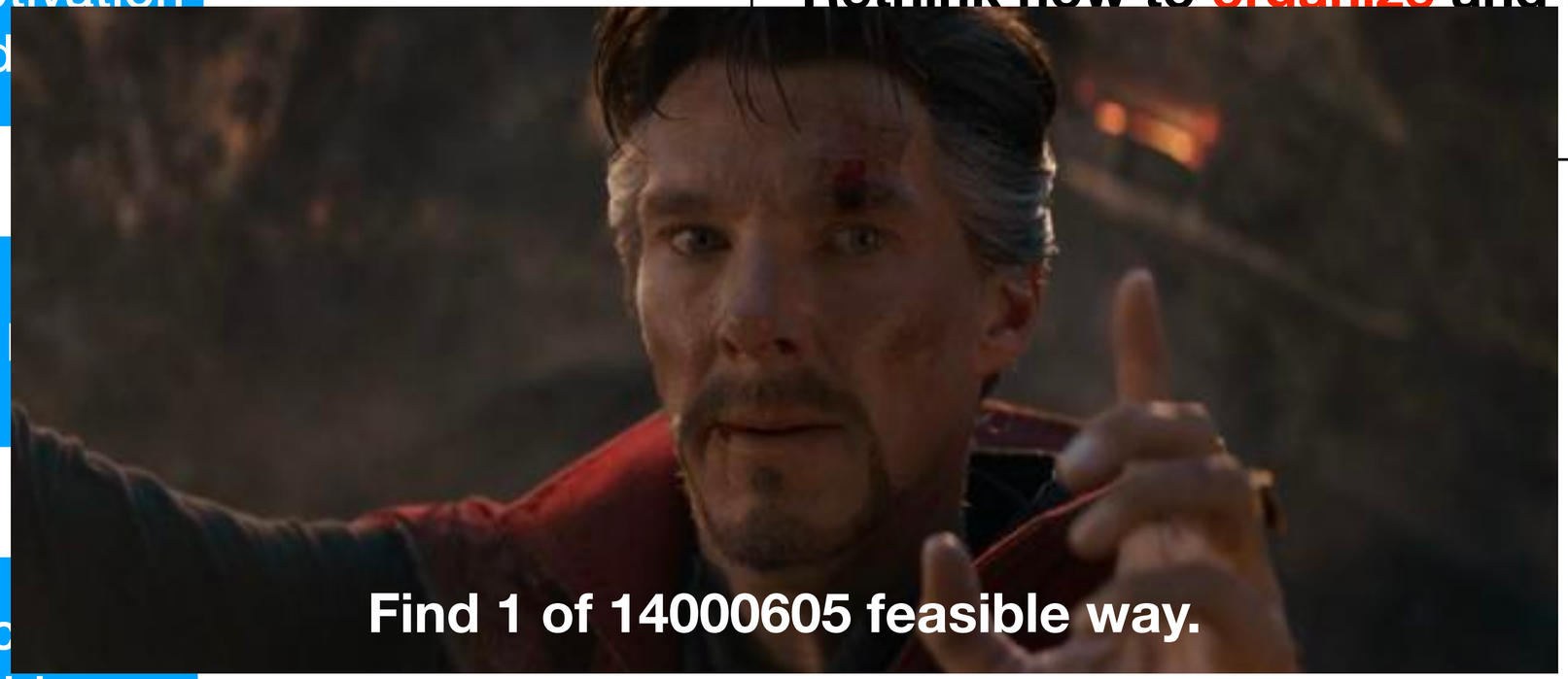
Motivation d

Tec driven

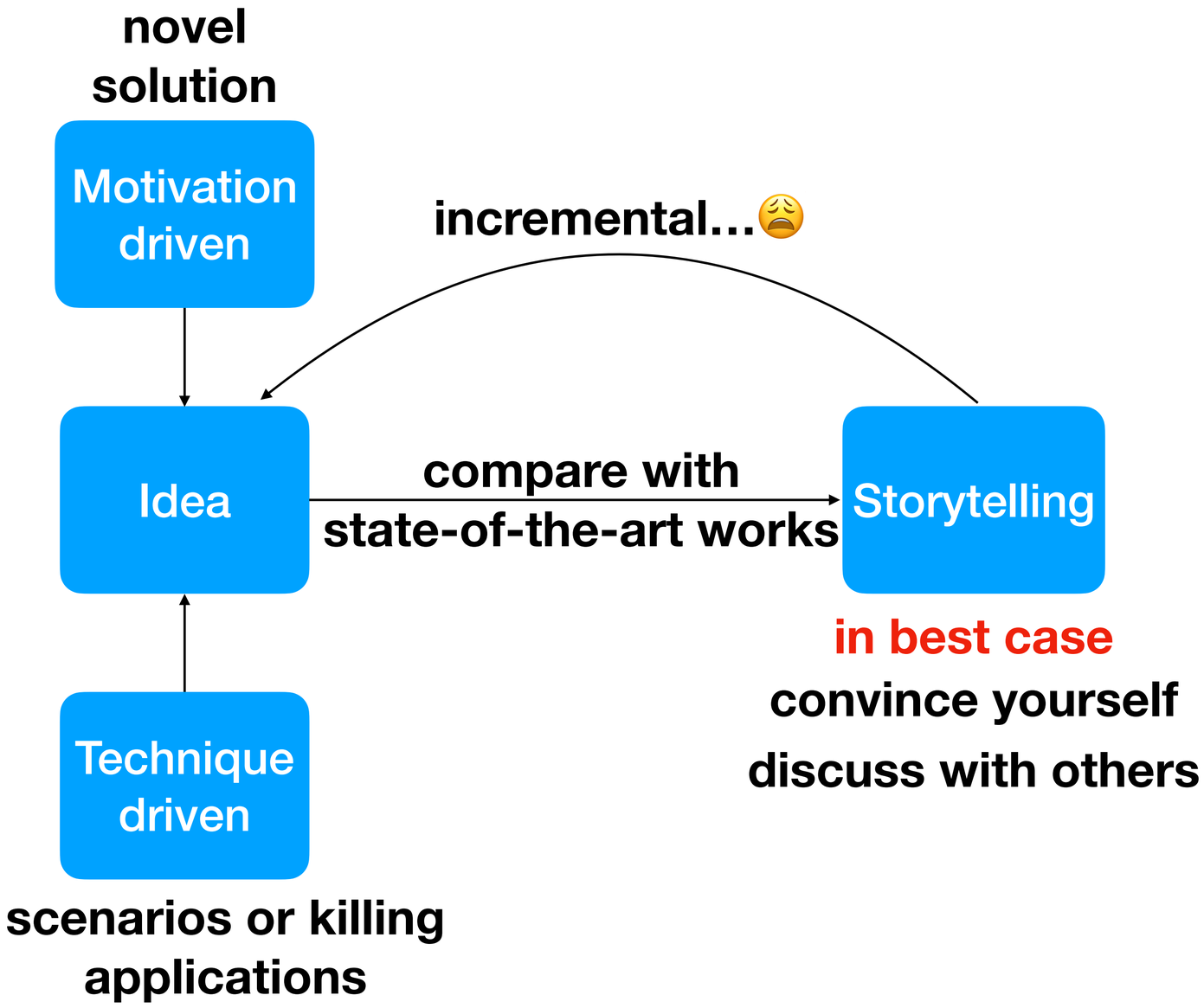
scenarios or killing applications

If others cannot understand the story

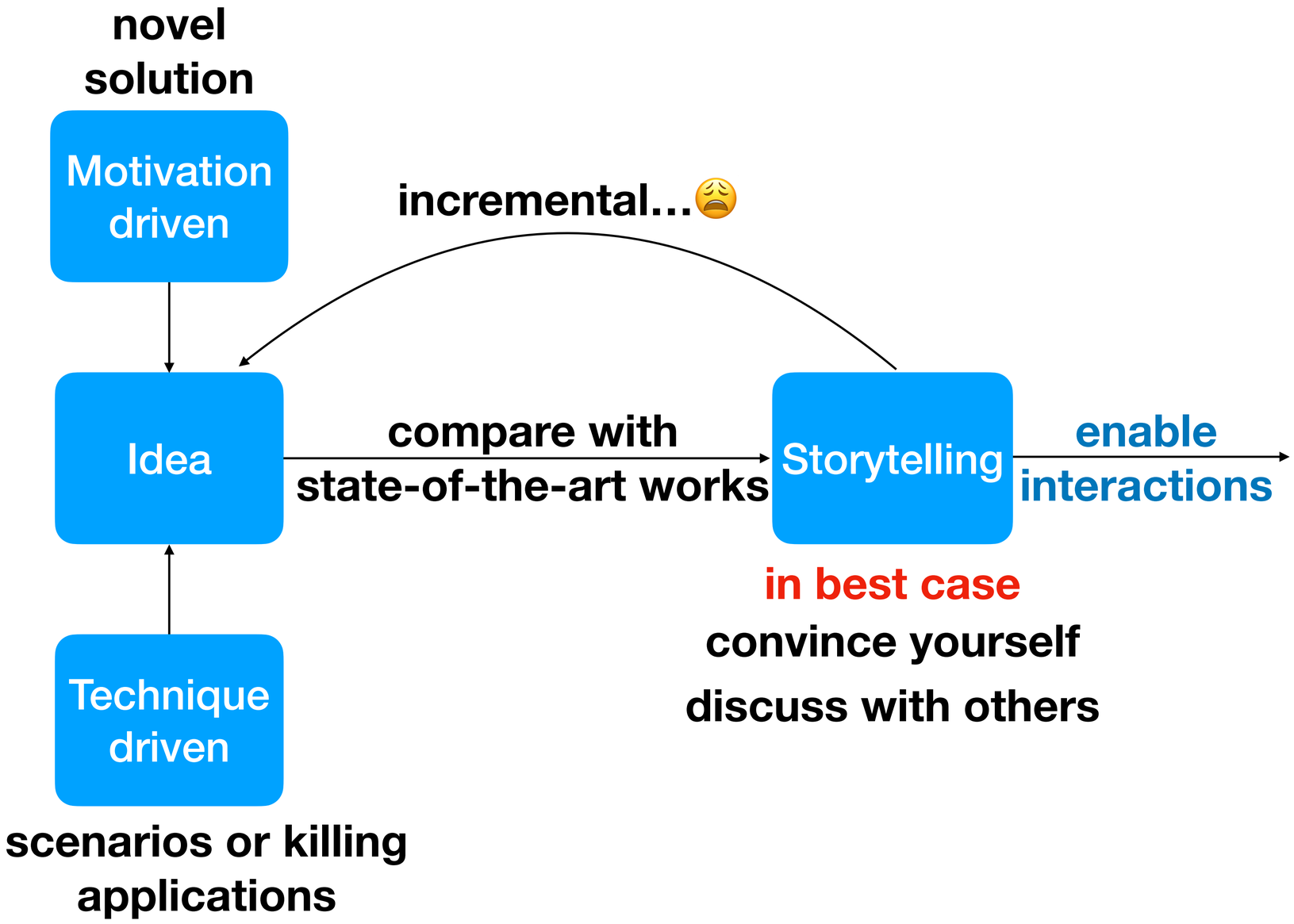
- Rethink how to organize and tell the story



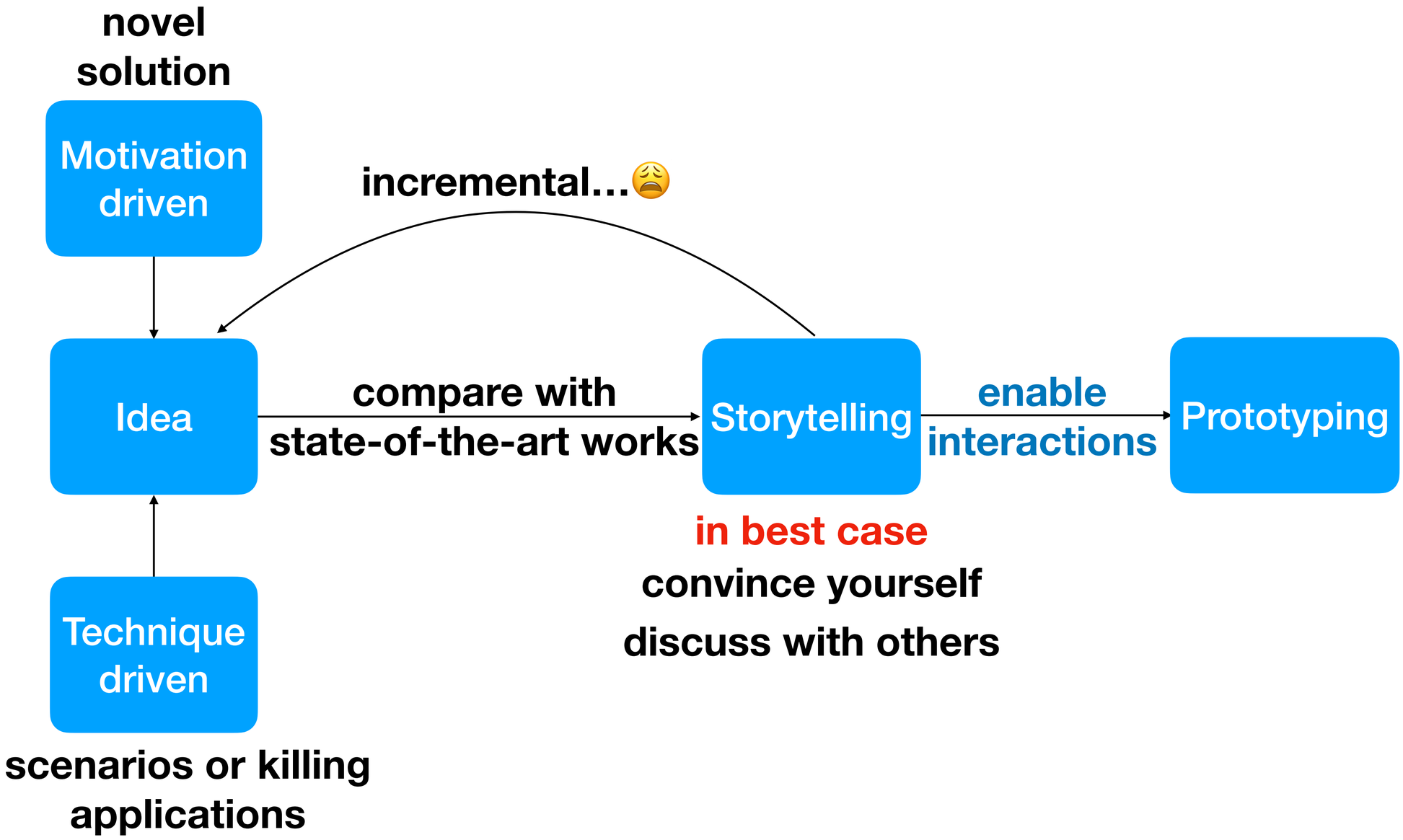
HCI Research Flow



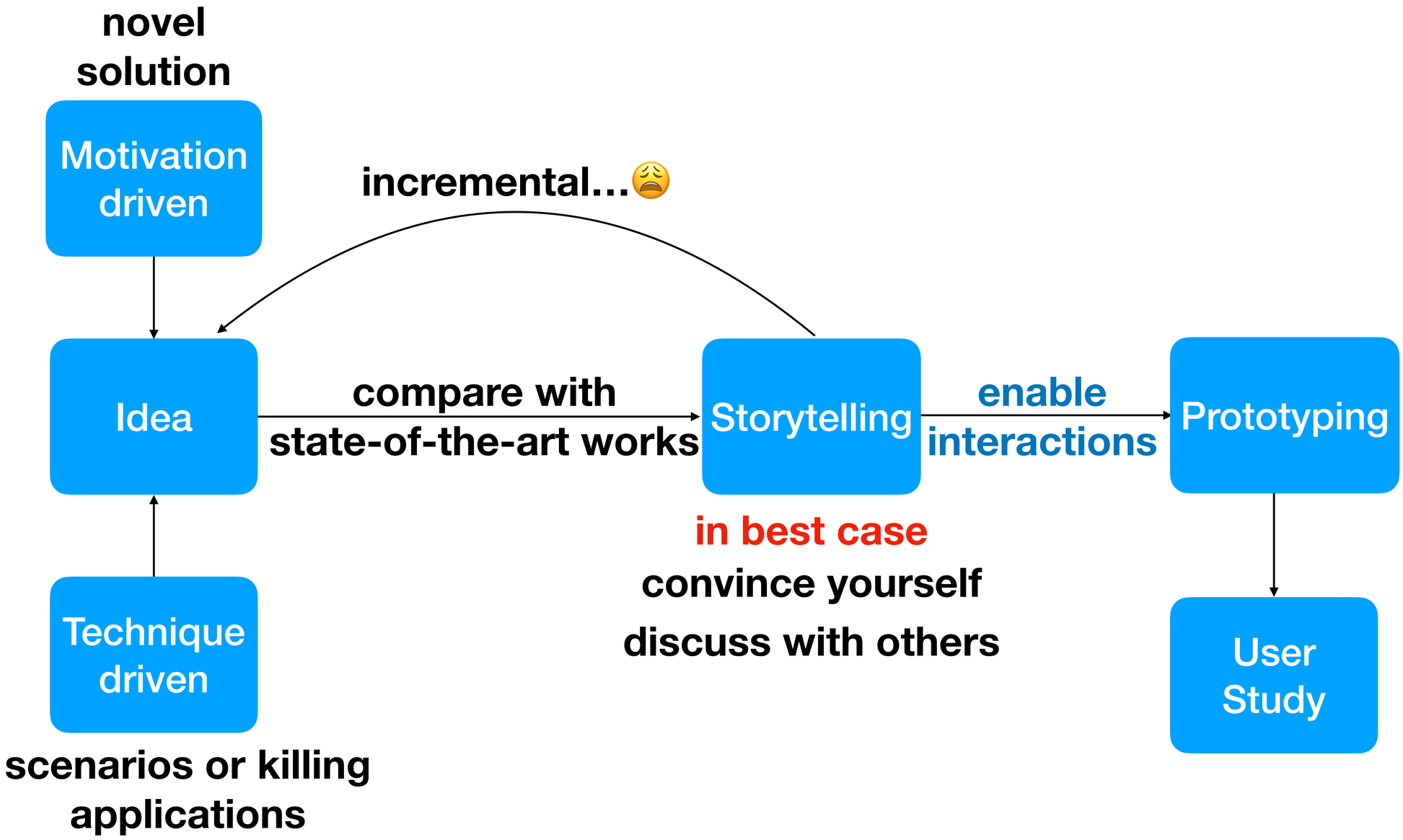
HCI Research Flow



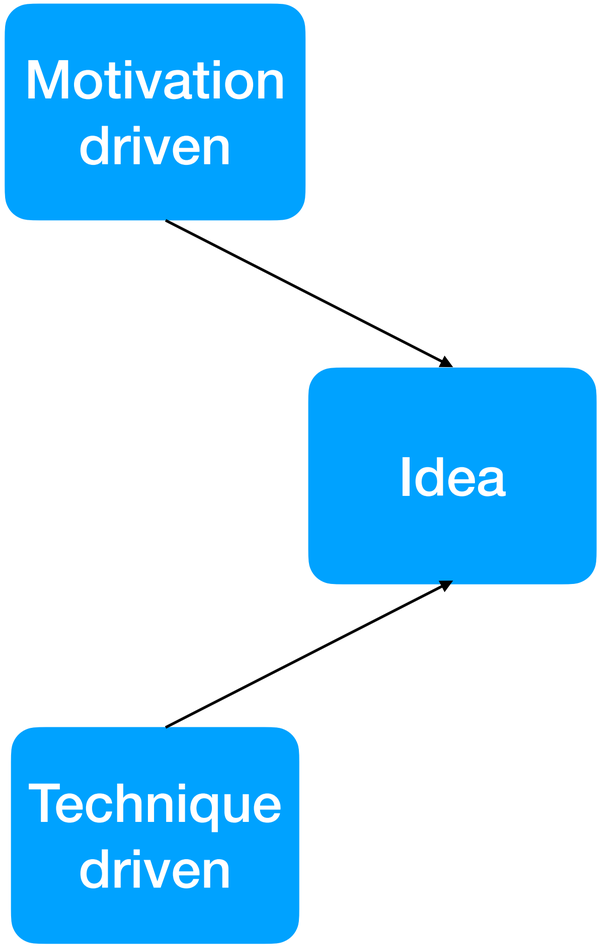
HCI Research Flow



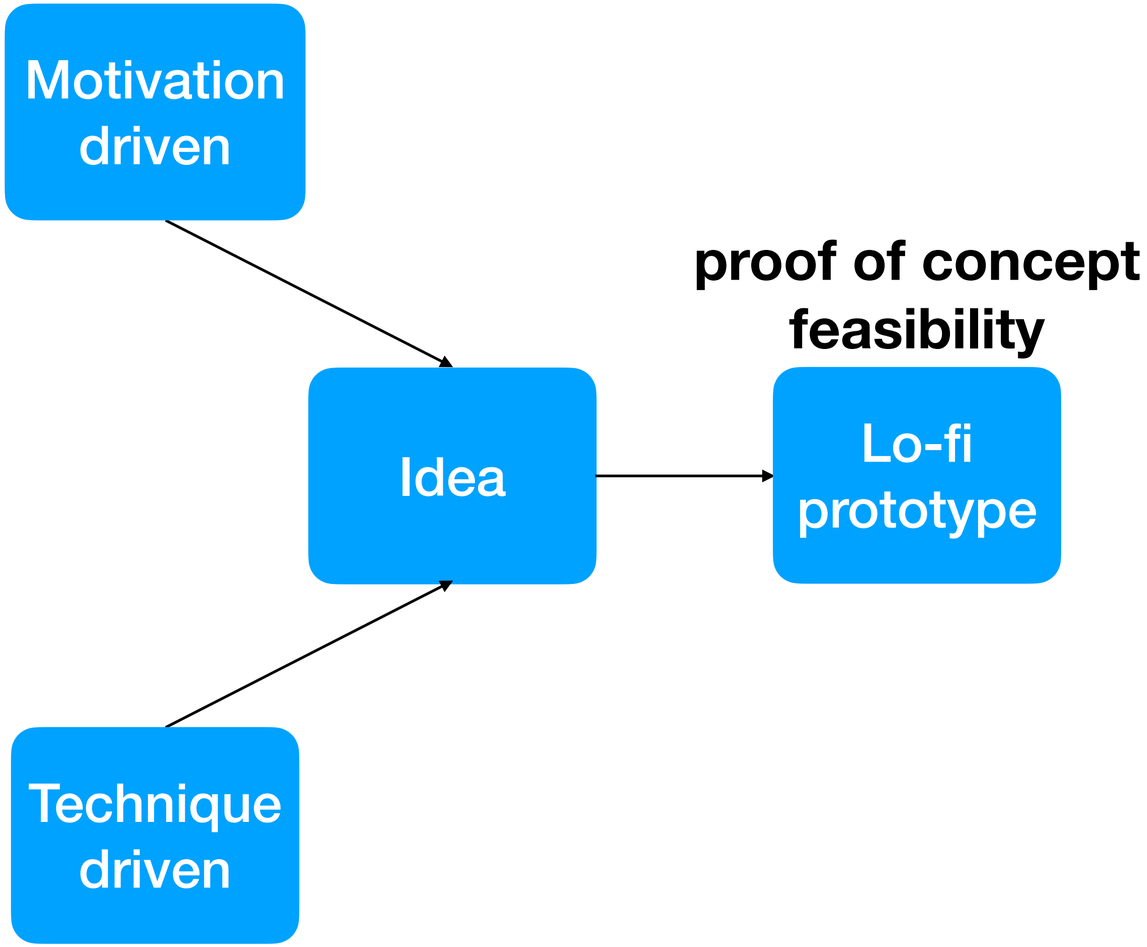
HCI Research Flow



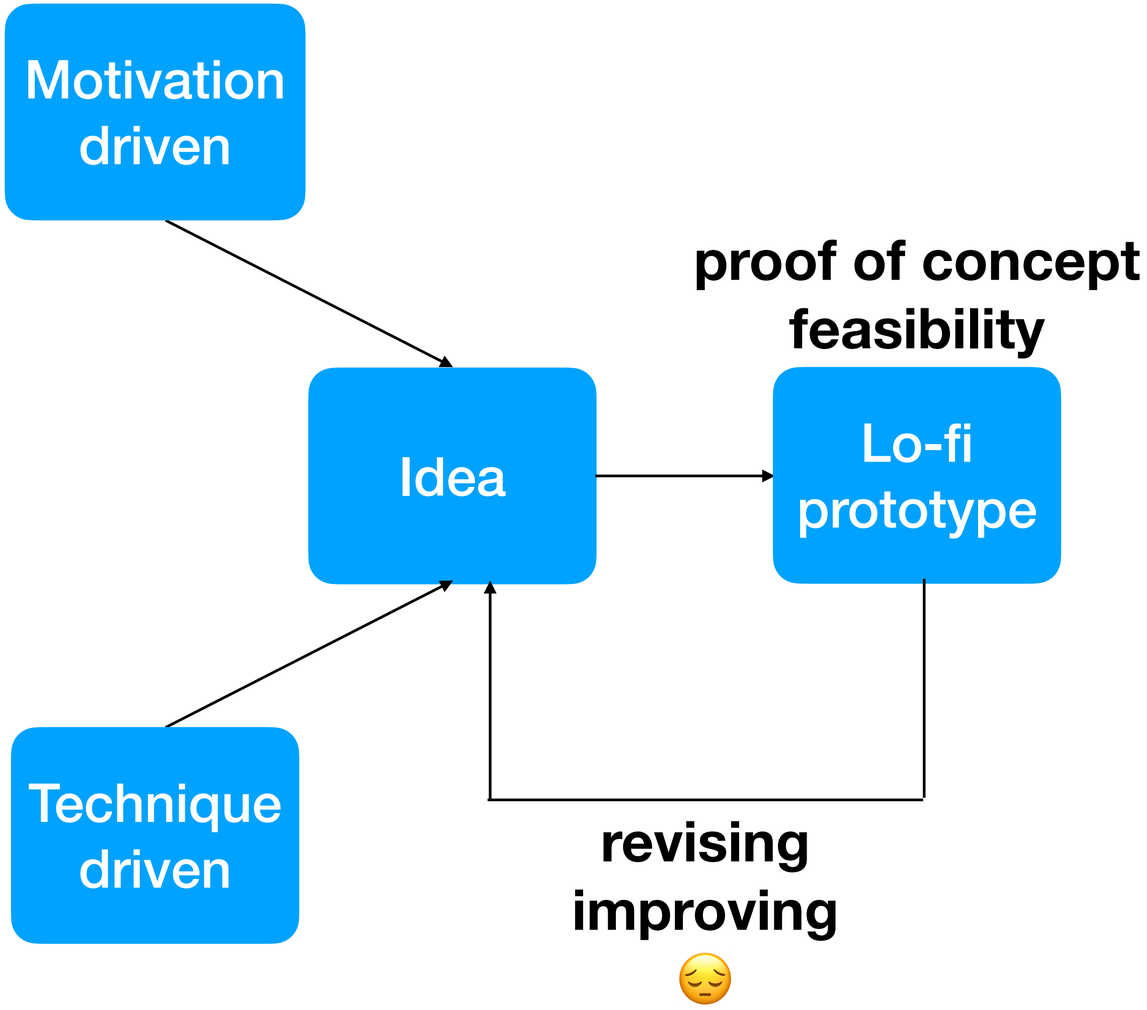
Prototyping Flow



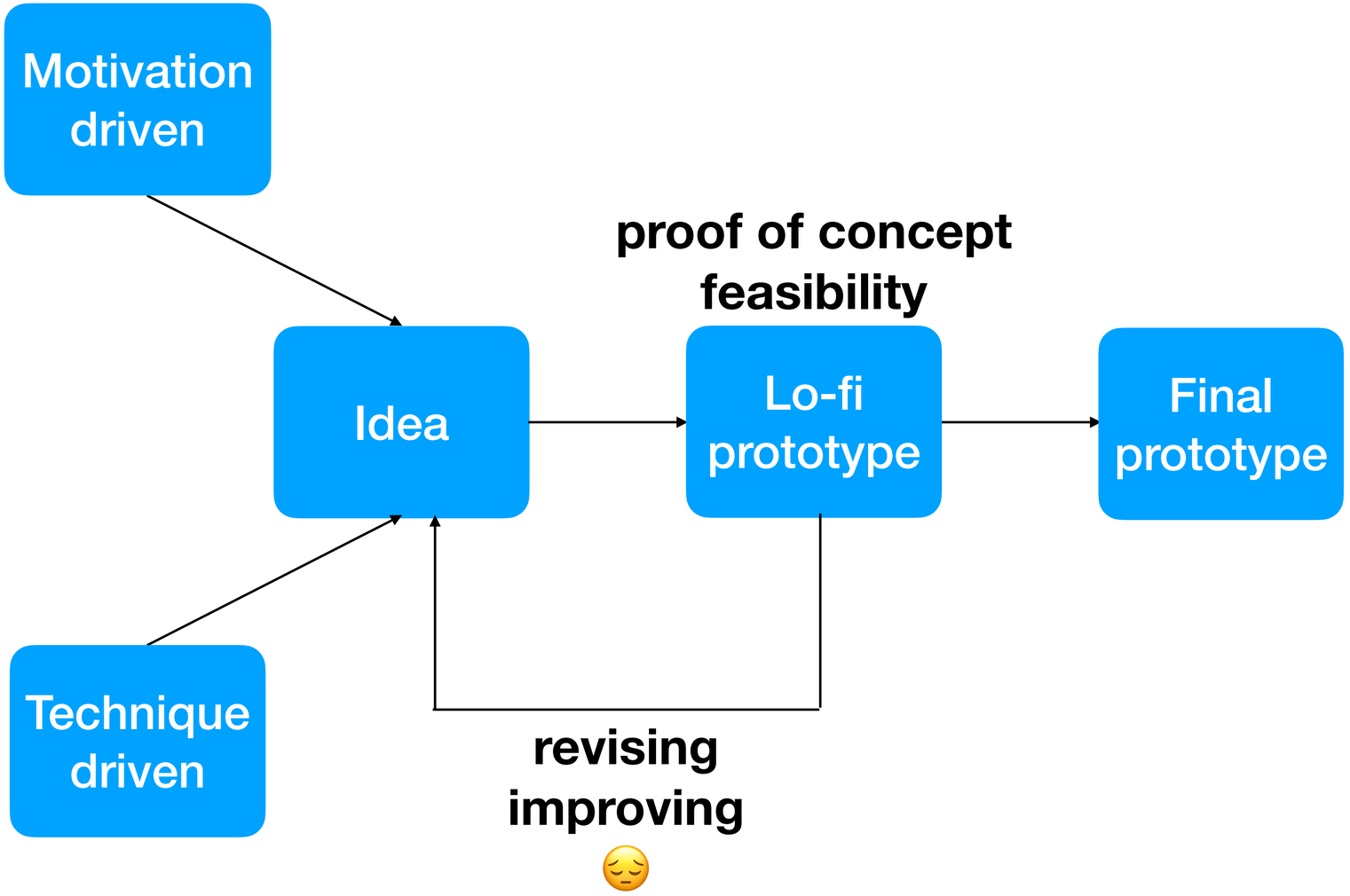
Prototyping Flow



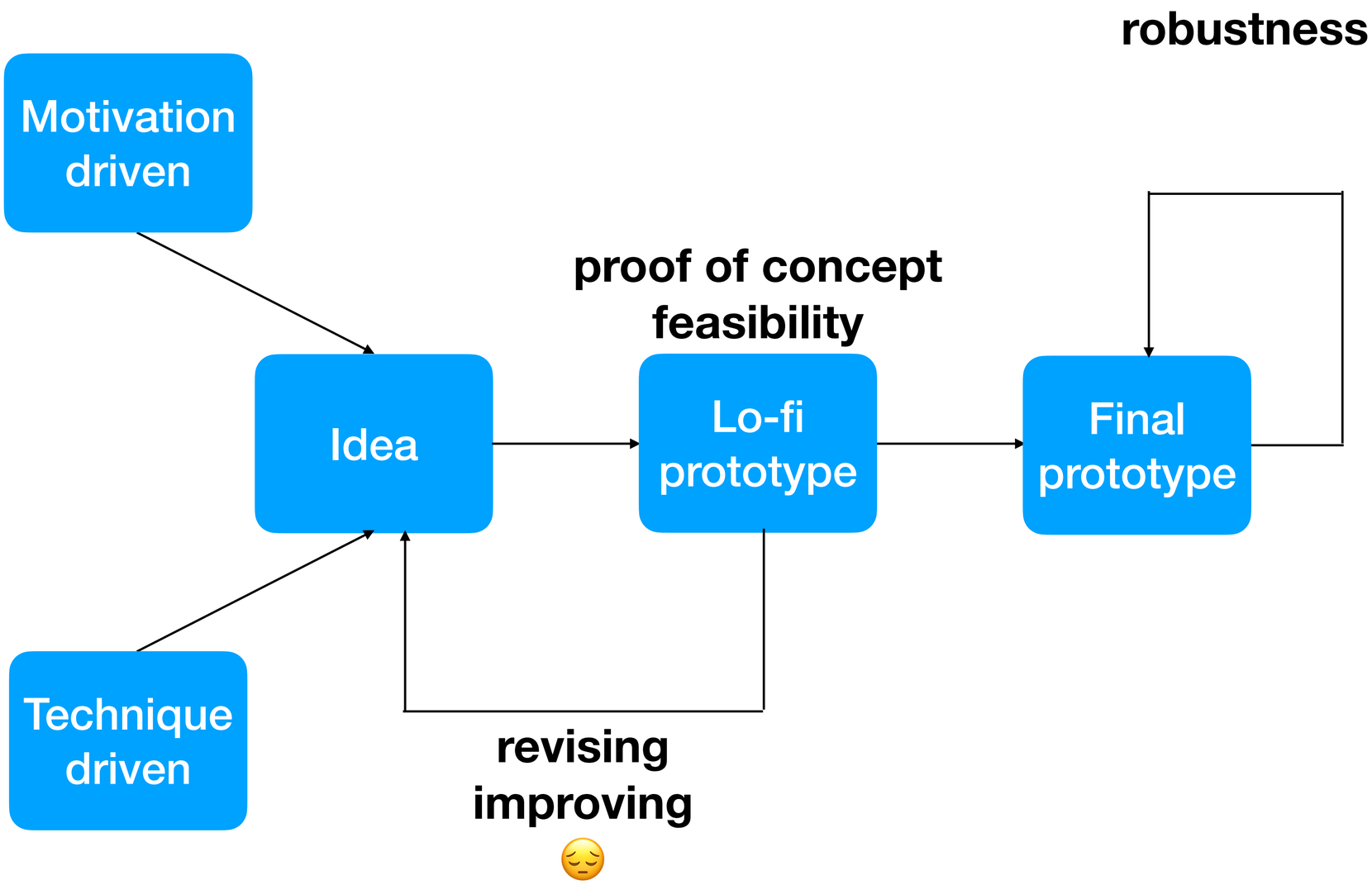
Prototyping Flow



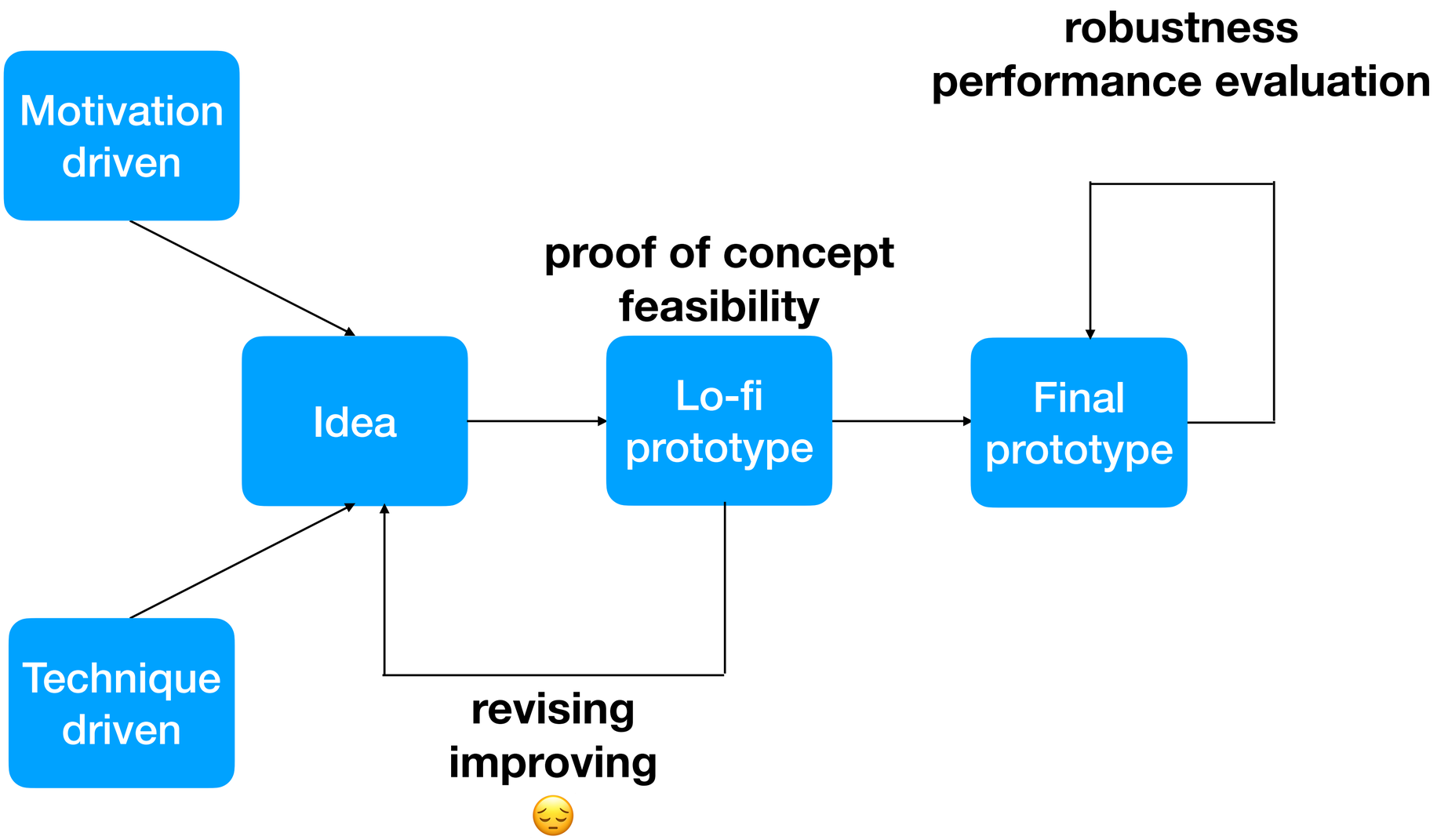
Prototyping Flow



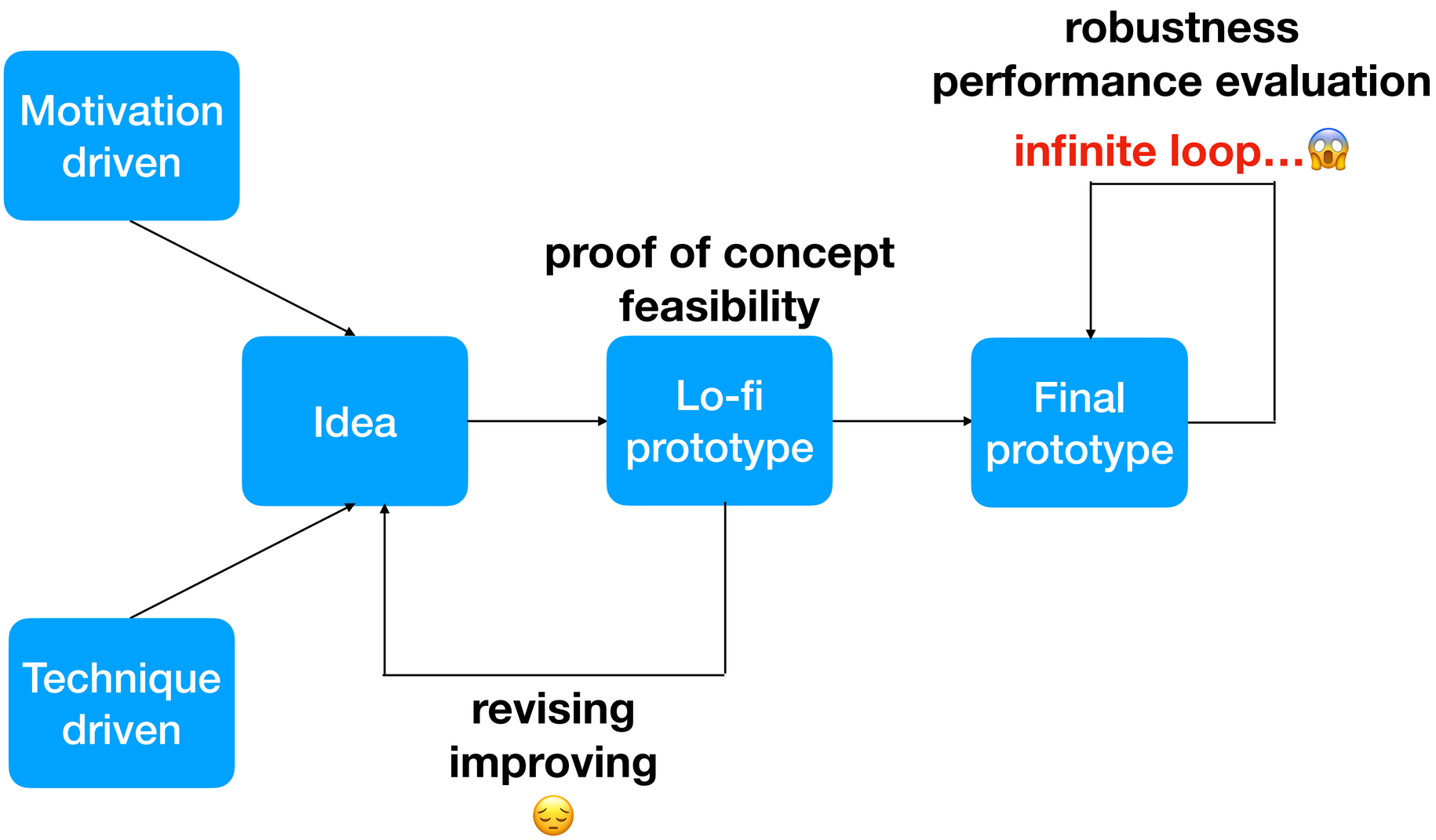
Prototyping Flow



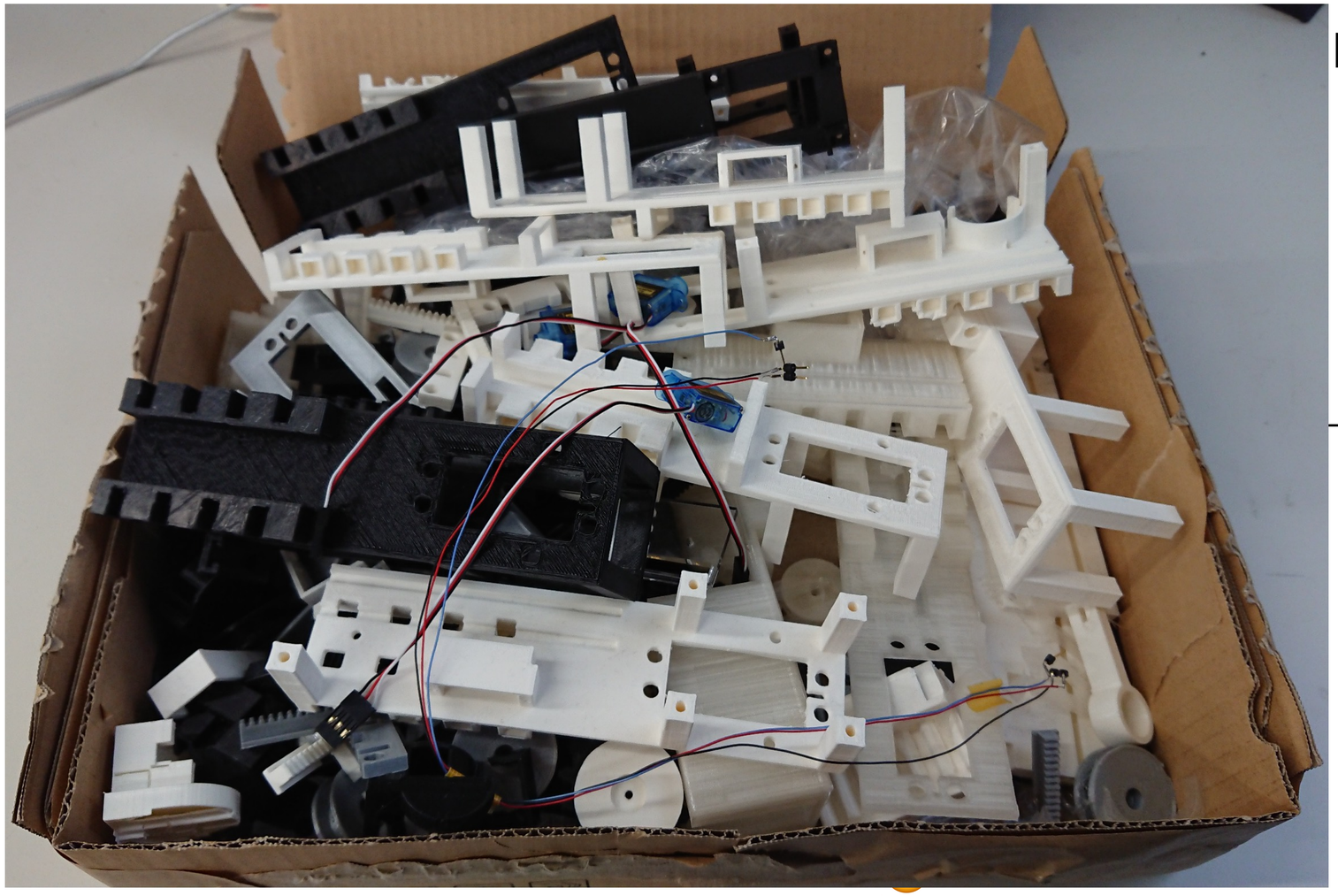
Prototyping Flow



Prototyping Flow

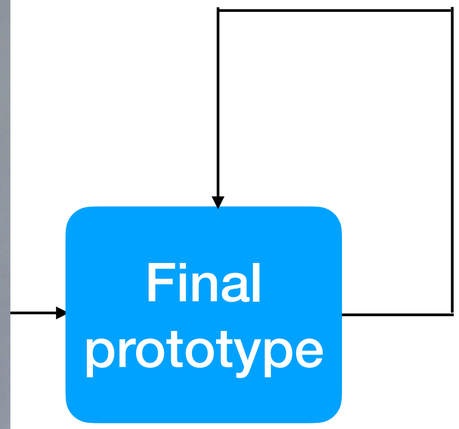


Prototyping Flow

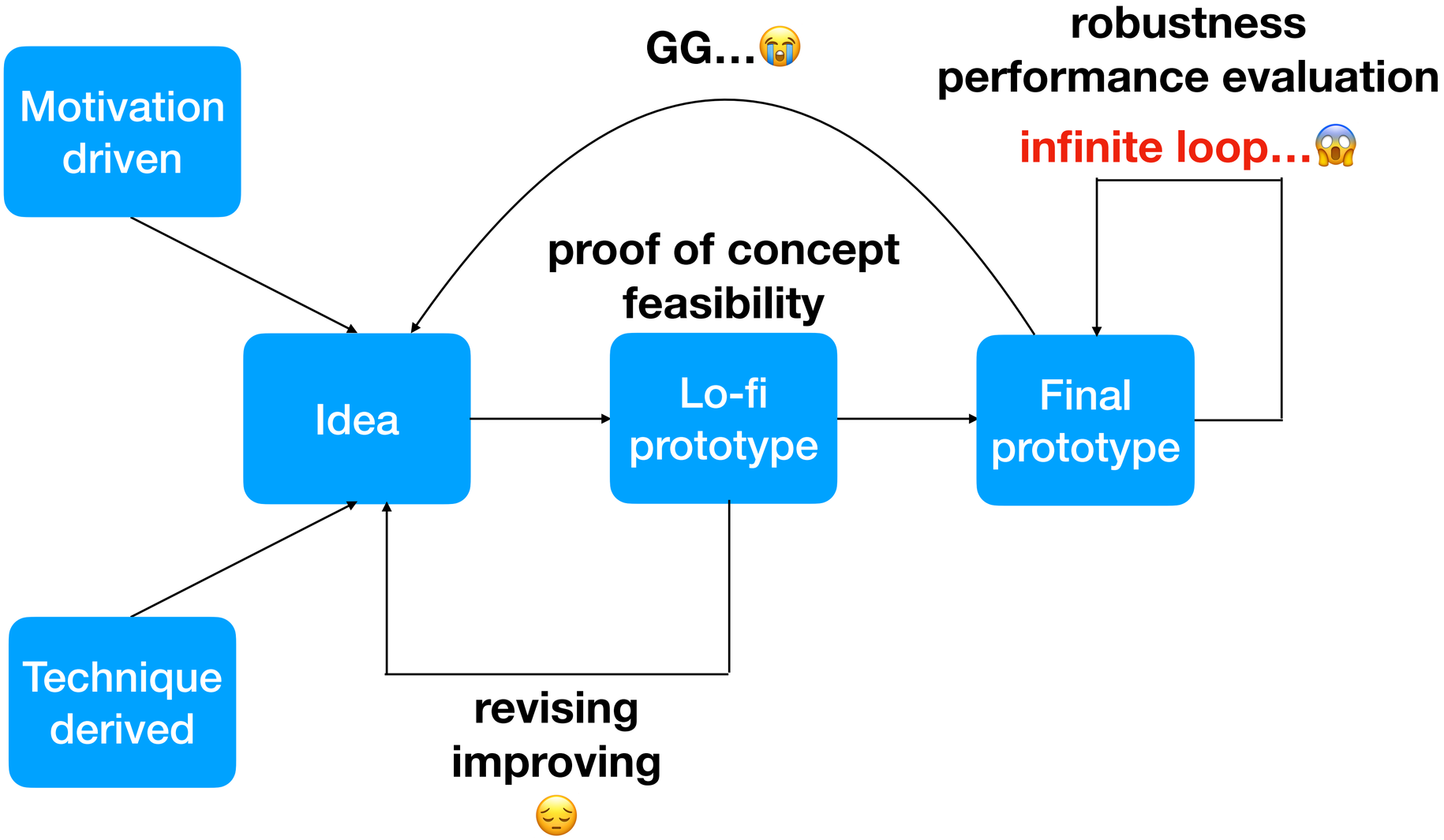


robustness
performance evaluation

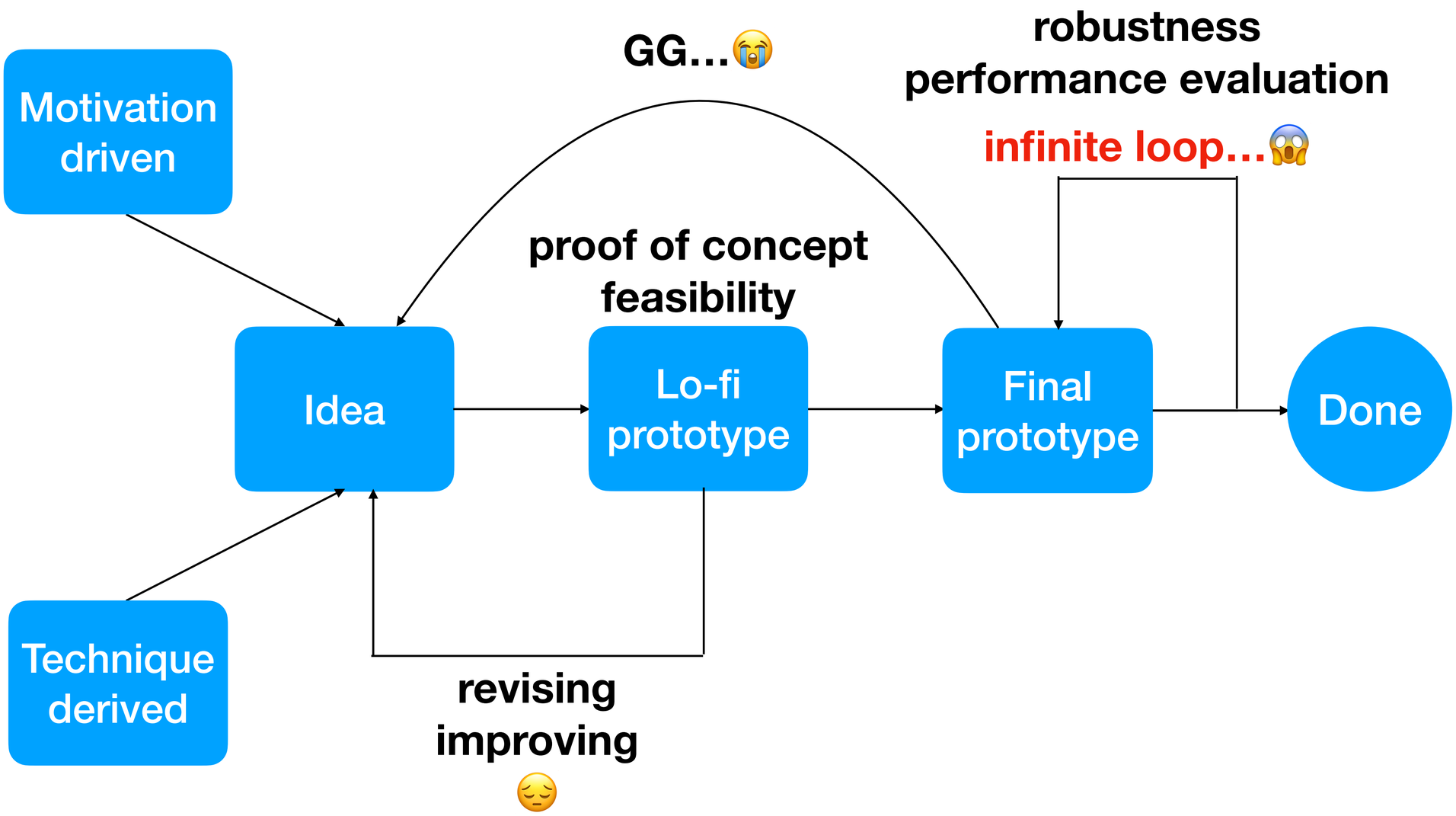
infinite loop…😱



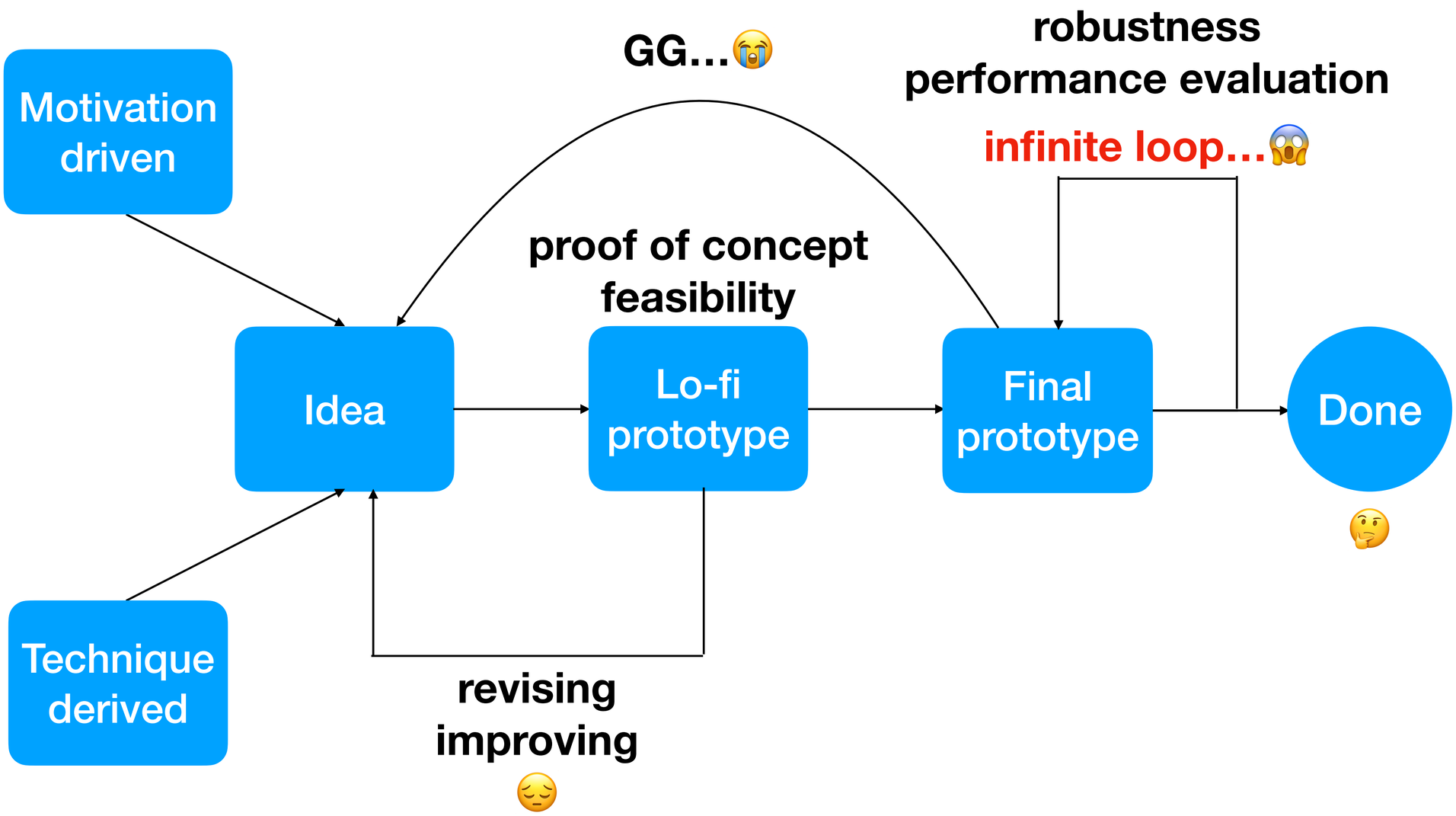
Prototyping Flow



Prototyping Flow



Prototyping Flow



HCI Researcher

HCI Researcher

Observation
Inspiration
Brainstorming

HCI Researcher

Observation
Inspiration
Brainstorming

- **Seeing**
- **Looking at**
- **Observing**

HCI Researcher

Observation
Inspiration
Brainstorming

- **Seeing**: aware something happens

HCI Researcher

Observation
Inspiration
Brainstorming

- **Seeing**: aware something happens
- **Looking at**: pay attention on something happening

HCI Researcher

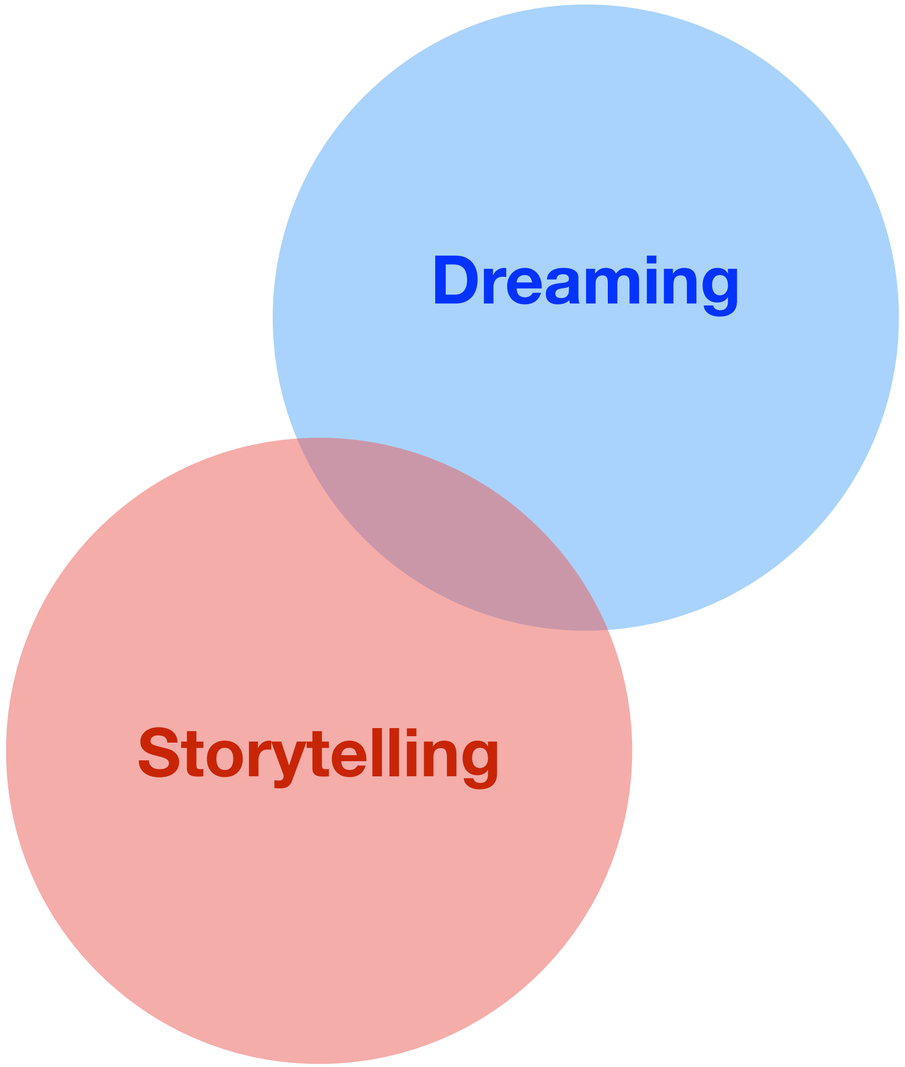
Observation
Inspiration
Brainstorming

- **Seeing**: aware something happens
- **Looking at**: pay attention on something happening
- **Observing**: think about how something happens & how to improve it

HCI Researcher

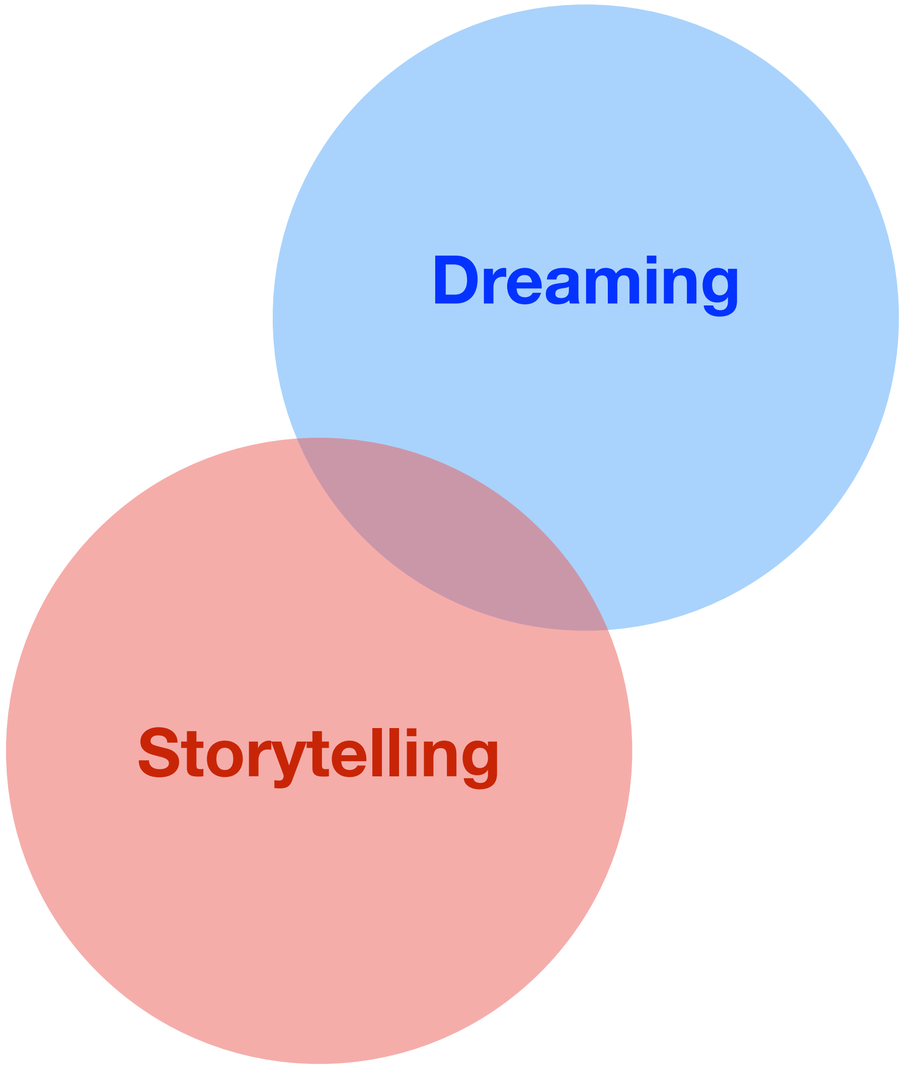
Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!

HCI Researcher



Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!

HCI Researcher

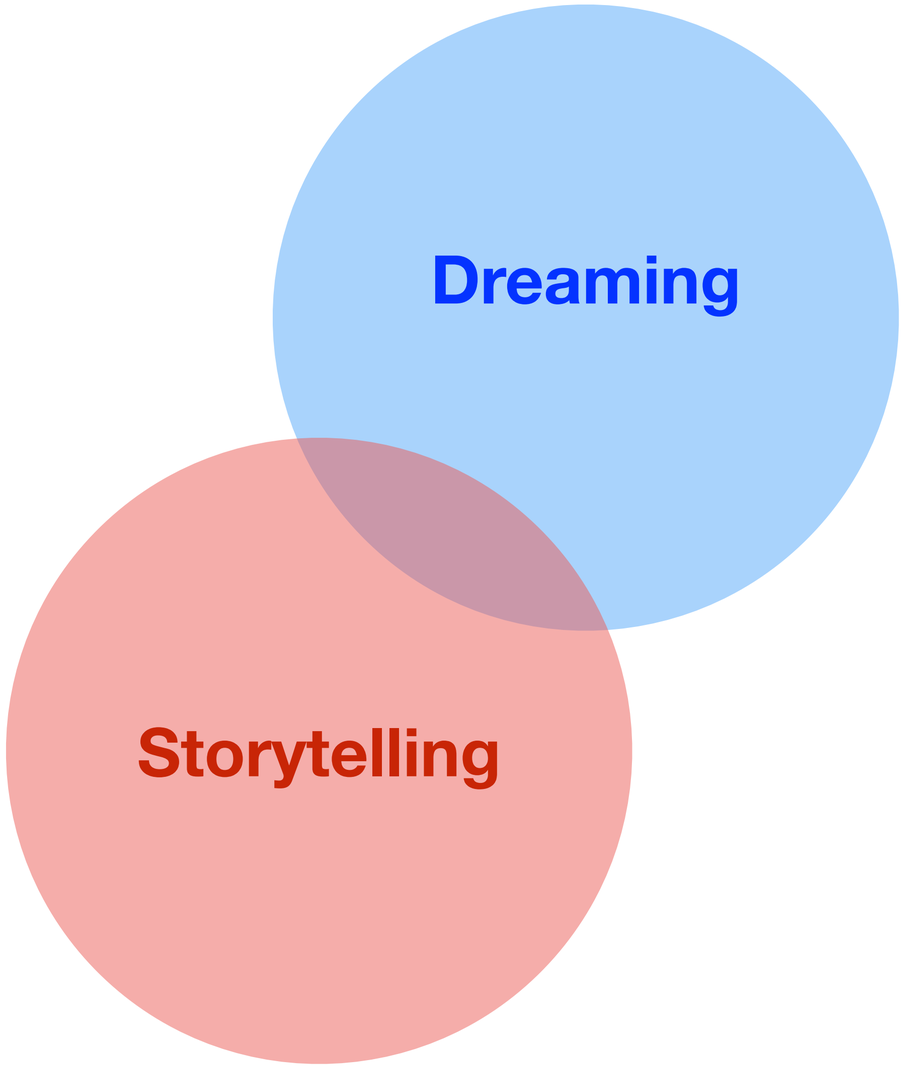


Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!

Evaluation
Organization
Rationality
Presentation

HCI Researcher

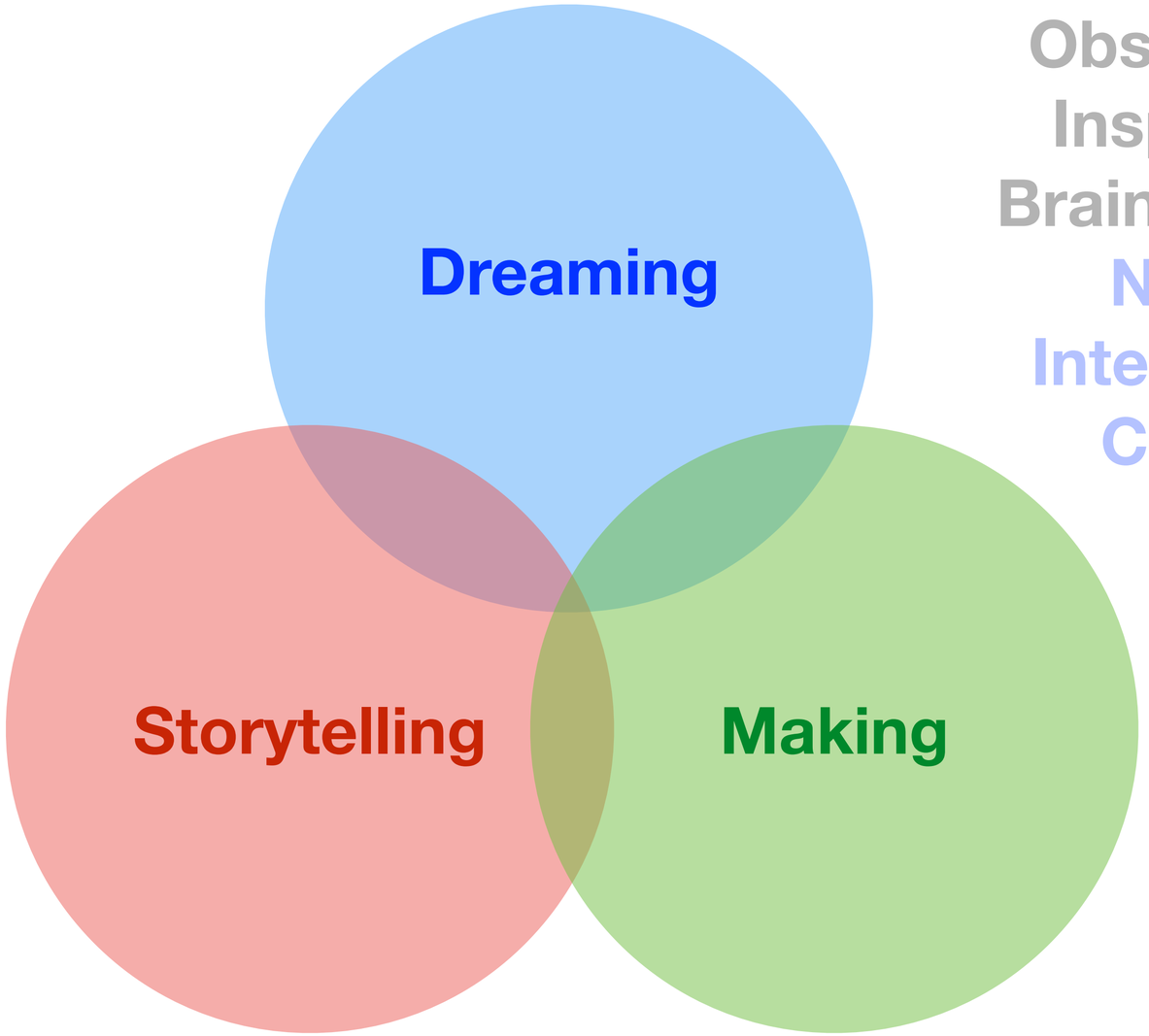
Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!



Insight
Vision
Evaluation
Organization
Rationality
Presentation

HCI Researcher

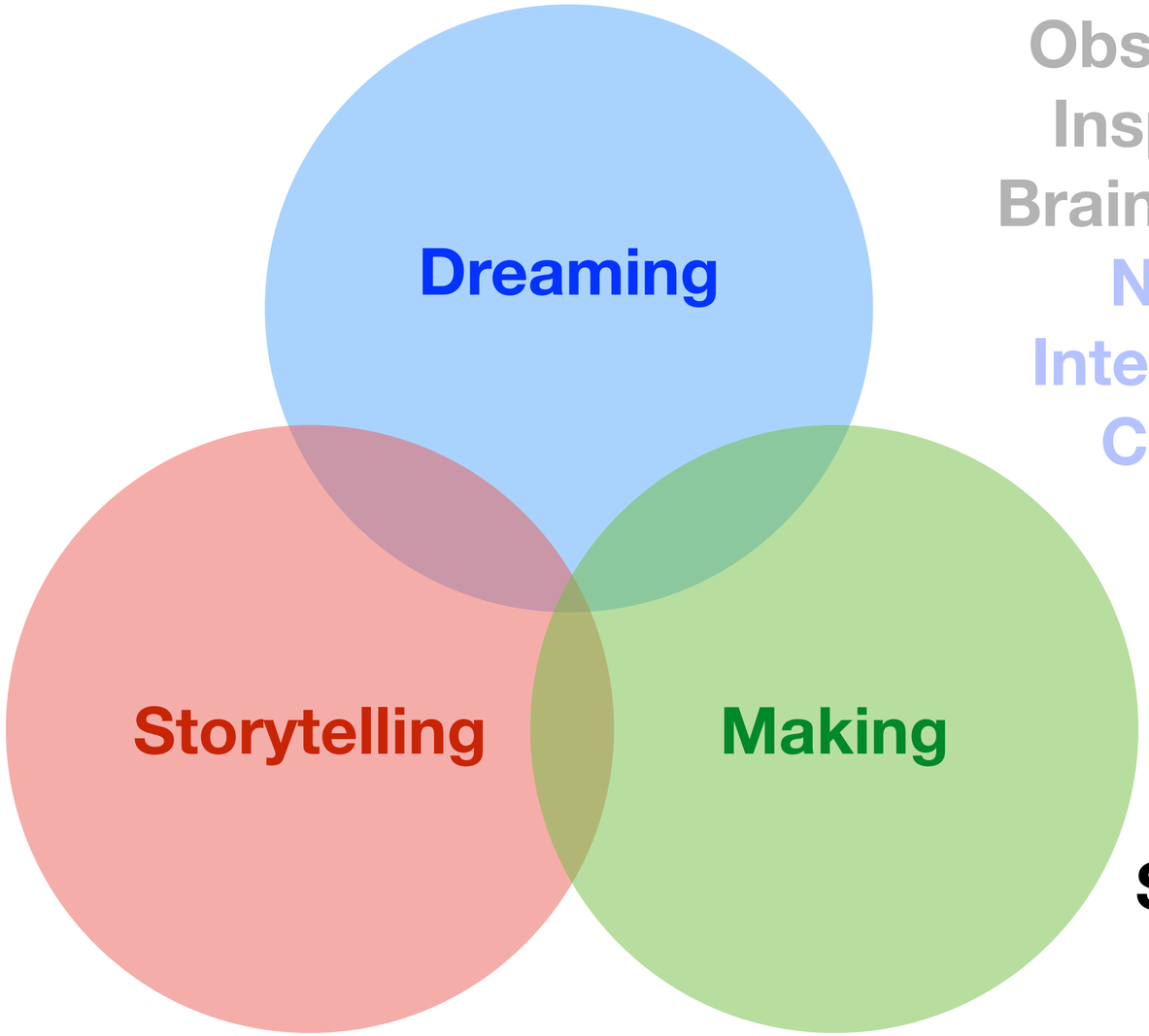
Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!



Insight
Vision
Evaluation
Organization
Rationality
Presentation

HCI Researcher

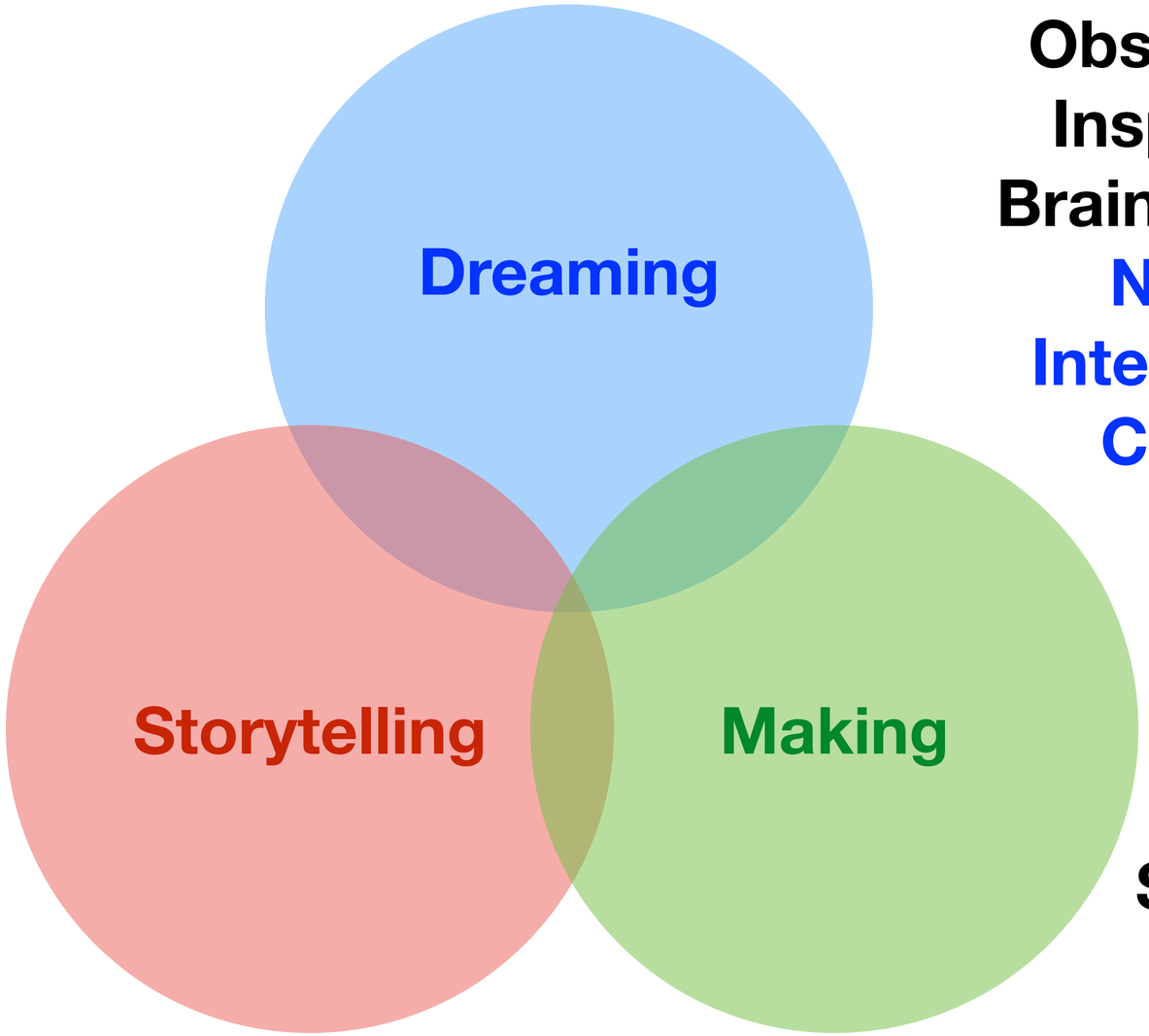
Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!



Insight
Vision
Evaluation
Organization
Rationality
Presentation

Searching
Learning

HCI Researcher



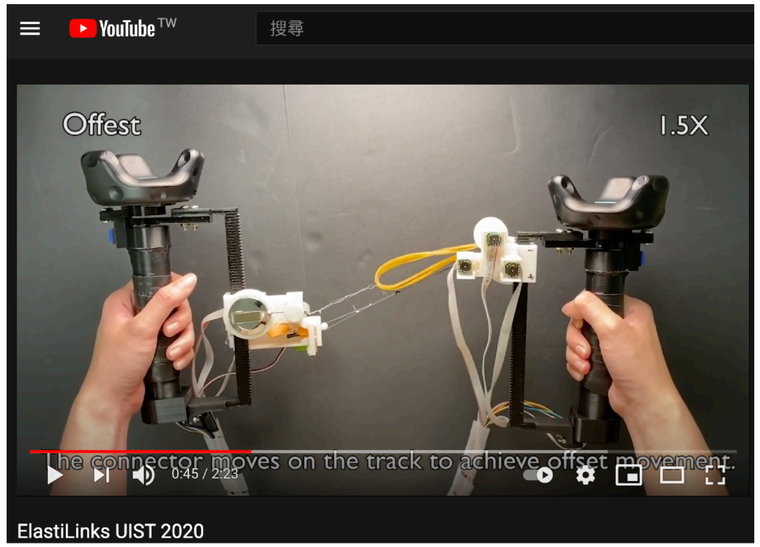
Observation
Inspiration
Brainstorming
Novel~
Interesting~
Crazy!!!

Insight
Vision
Evaluation
Organization
Rationality
Presentation

Searching
Learning

Paper Reading

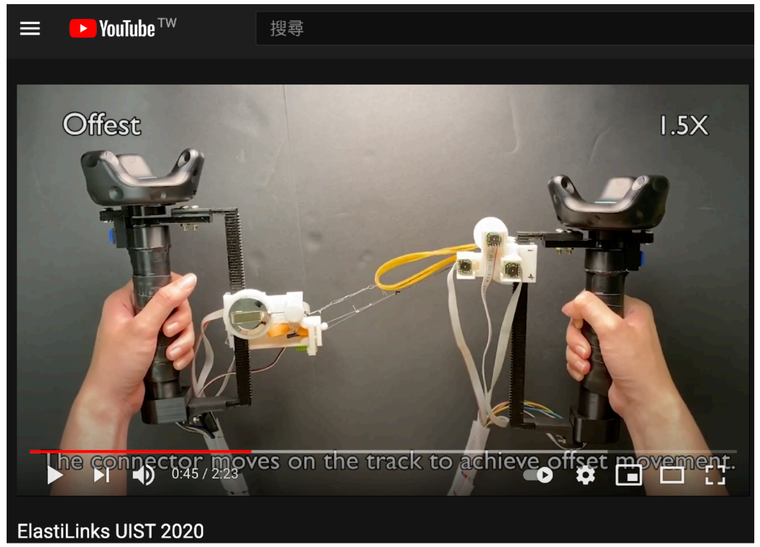
Video →



Paper Reading

Video ⟶ **Teaser**

understandable even without caption



Session 15A: ForceFeedback and Haptics in VR UIST '20, October 20–23, 2020, Virtual Event, USA

ElastiLinks: Force Feedback between VR Controllers with Dynamic Points of Application of Force

Tzu-Yun Wei*‡ Hsin-Ruey Tsai†‡ Yu-So Liao†§ Chieh Tsai†‡
Yi-Shan Chen*‡ Chi Wang*‡ Bing-Yu Chen*¶
*National Taiwan University †National Chengchi University
‡{f61031, hsnuhrt, tsaichieh850929, yishan13.0702, chi541323}@gmail.com
§{108753133}g.nccu.edu.tw ¶{robin}@ntu.edu.tw

Teaser

ABSTRACT
Force feedback is commonly used to enhance realism in virtual reality (VR). However, current works mainly focus on providing different force types or patterns, but do not investigate how a proper point of application of force (PAF), which means where the resultant force is applied to, affects users' experience. For example, users perceive resistive force without torque when pulling a virtual bow, but with torque when pulling a virtual slingshot. Therefore, we propose a set of handheld controllers, ElastiLinks, to provide force feedback between controllers with dynamic PAFs. A rotatable track on each controller provides a dynamic PAF, and two common types of force feedback, resistive force and impact, are produced by two links, respectively. We performed a force perception study to ascertain users' resistive and impact force level distinguishability between controllers. Based on the results, we conducted another perception study to understand users' distinguishability of PAF offset and rotation differences. Finally, we performed a VR experience study to prove that force feedback with dynamic PAFs enhances VR experience.

Author Keywords
Haptic feedback; force feedback; resistive force; impact; point of application of force; virtual reality

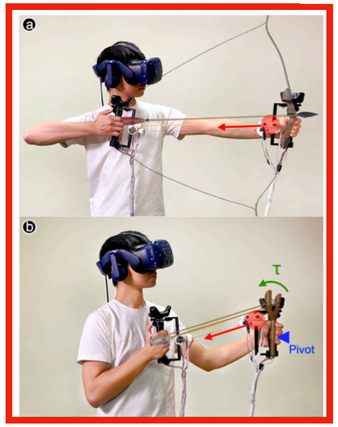
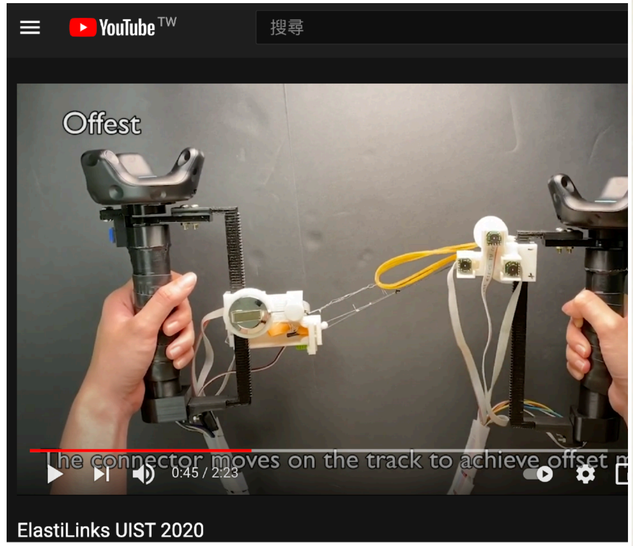


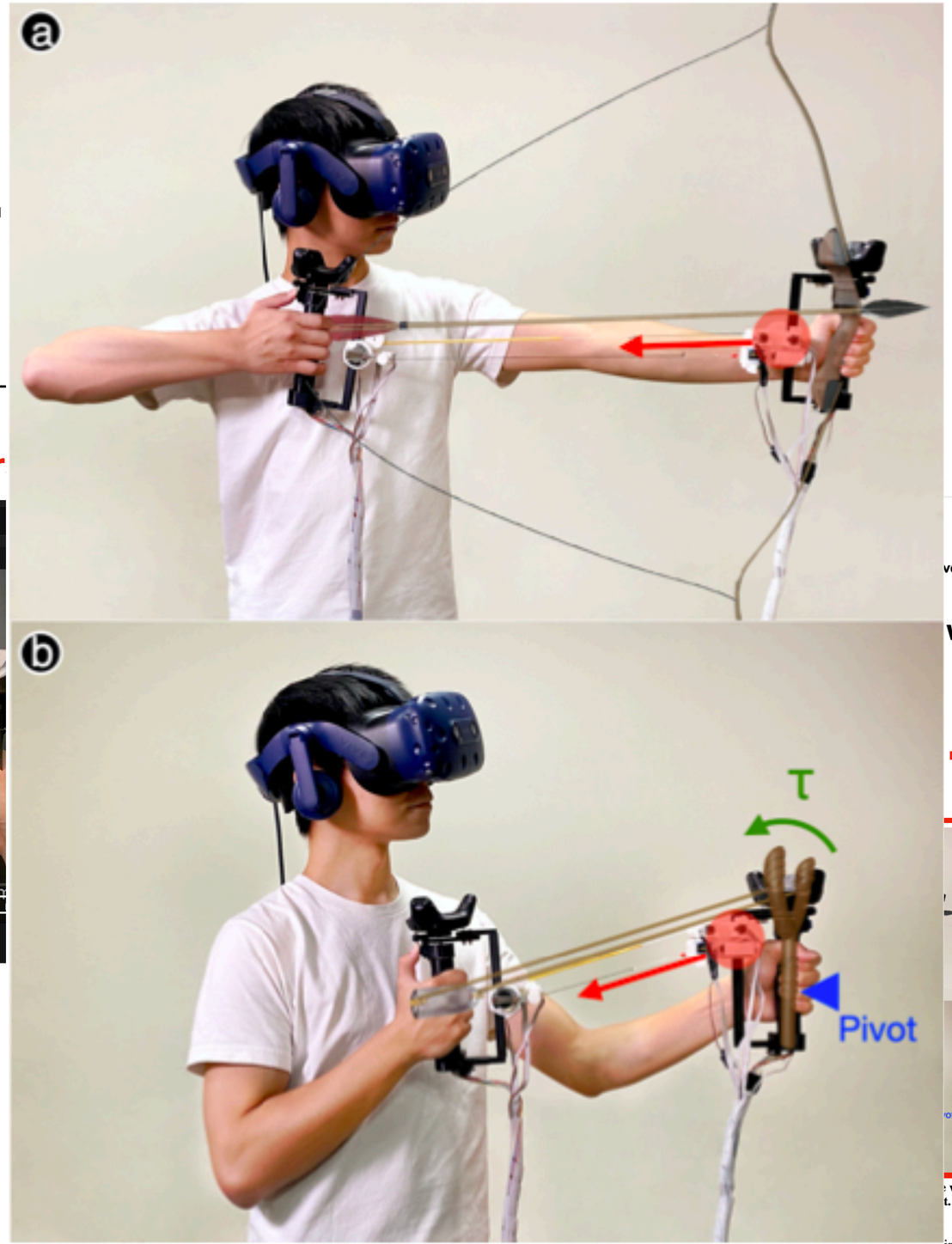
Figure 1. Users perceive resistive force (a) without torque when pulling a virtual bow, and (b) with torque when pulling a slingshot. The translucent red point is the point of application of force (PAF).

Pa

Video

under





vent, USA

with

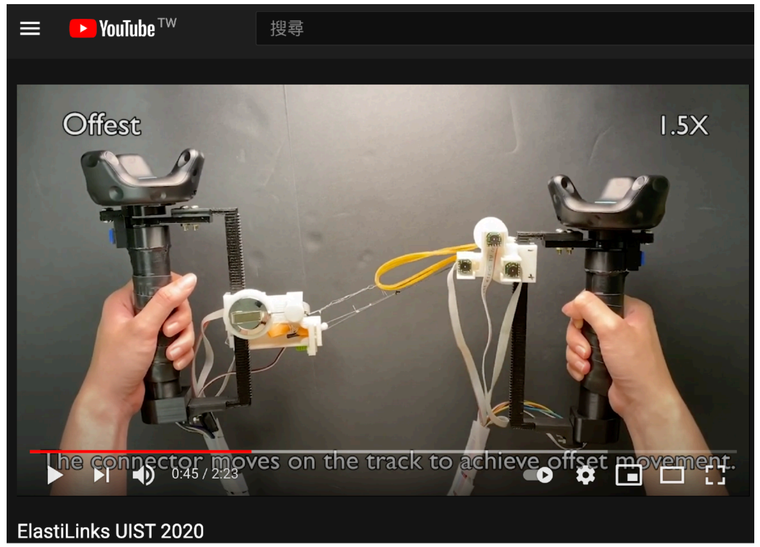
Teaser

when pulling
t. The translu-

Paper Reading

Video → **Teaser** → **Abstract**

understandable even without caption



Session 15A: ForceFeedback and Haptics in VR UIST '20, October 20–23, 2020, Virtual Event, USA

ElastiLinks: Force Feedback between VR Controllers with Dynamic Points of Application of Force

un Wei*‡ Hsin-Ruey Tsai†‡ Yu-So Liao†§ Chieh Tsai†‡
Yi-Shan Chen*‡ Chi Wang*‡ Bing-Yu Chen*¶
nal Taiwan University †National Chengchi University
nuhrt, tsaichieh850929, yishan13.0702, chi541323}@gmail.com
108753133}g.nccu.edu.tw ¶{robin}@ntu.edu.tw

Abstract

ABSTRACT
Force feedback is commonly used to enhance realism in virtual reality (VR). However, current works mainly focus on providing different force types or patterns, but do not investigate how a proper point of application of force (PAF), which means where the resultant force is applied to, affects users' experience. For example, users perceive resistive force without torque when pulling a virtual bow, but with torque when pulling a virtual slingshot. Therefore, we propose a set of handheld controllers, ElastiLinks, to provide force feedback between controllers with dynamic PAFs. A rotatable track on each controller provides a dynamic PAF, and two common types of force feedback, resistive force and impact, are produced by two links, respectively. We performed a force perception study to ascertain users' resistive and impact force level distinguishability between controllers. Based on the results, we conducted another perception study to understand users' distinguishability of PAF offset and rotation differences. Finally, we performed a VR experience study to prove that force feedback with dynamic PAFs enhances VR experience.

Author Keywords
Haptic feedback; force feedback; resistive force; impact; point of application of force; virtual reality

Teaser

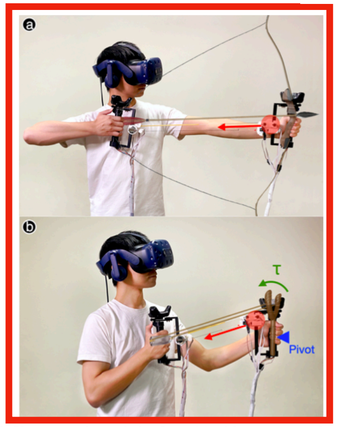


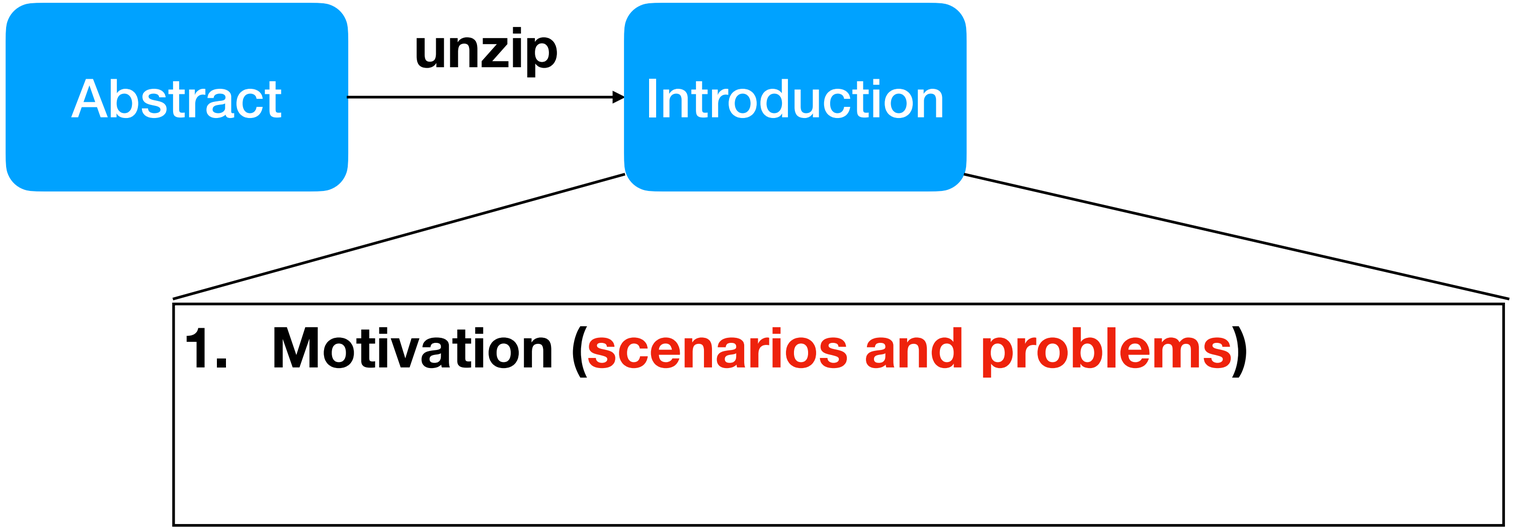
Figure 1. Users perceive resistive force (a) without torque when pulling a virtual bow, and (b) with torque when pulling a slingshot. The translucent red point is the point of application of force (PAF).

Paper Reading

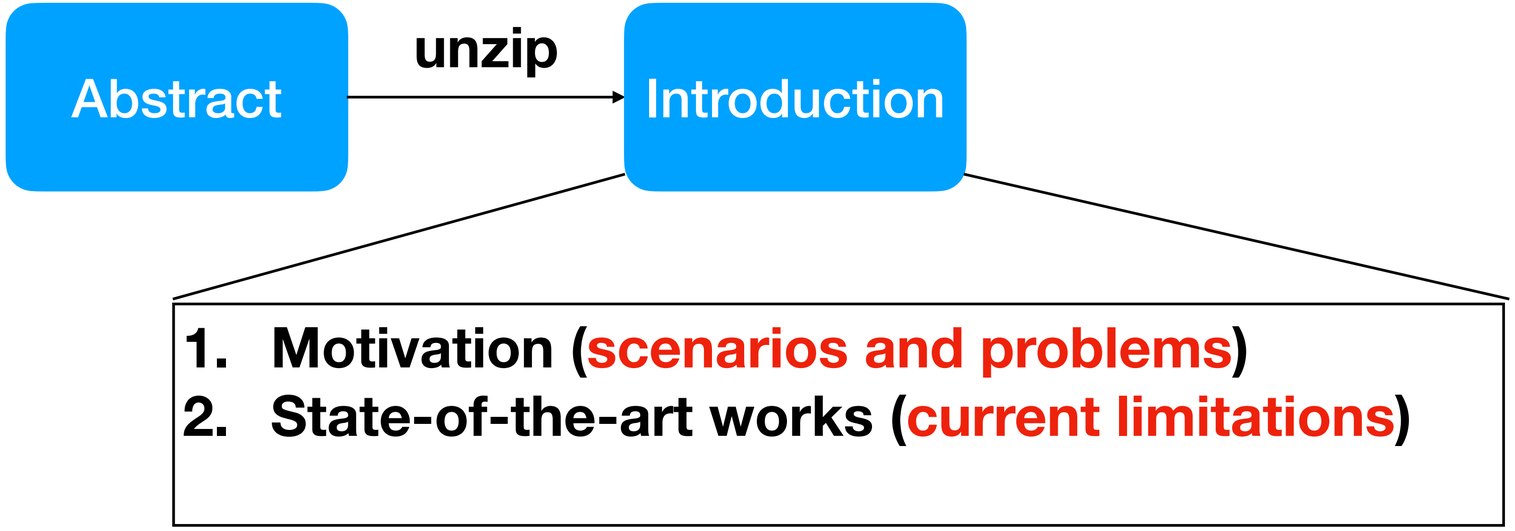
Abstract

Paper Reading

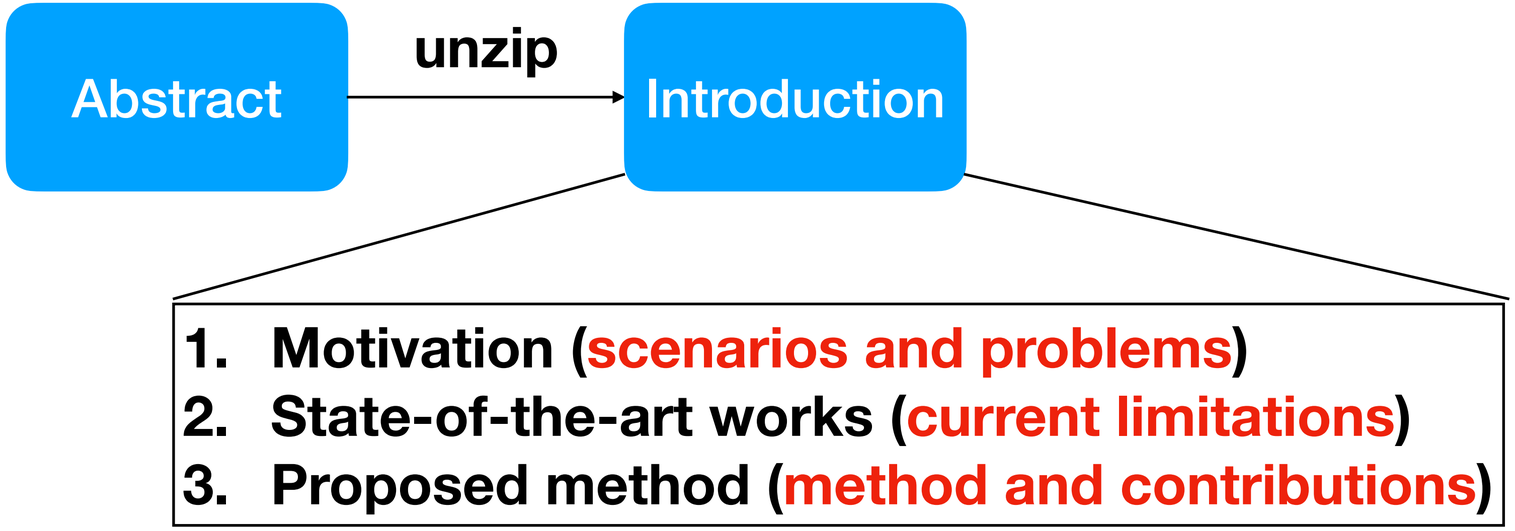
Paper Reading



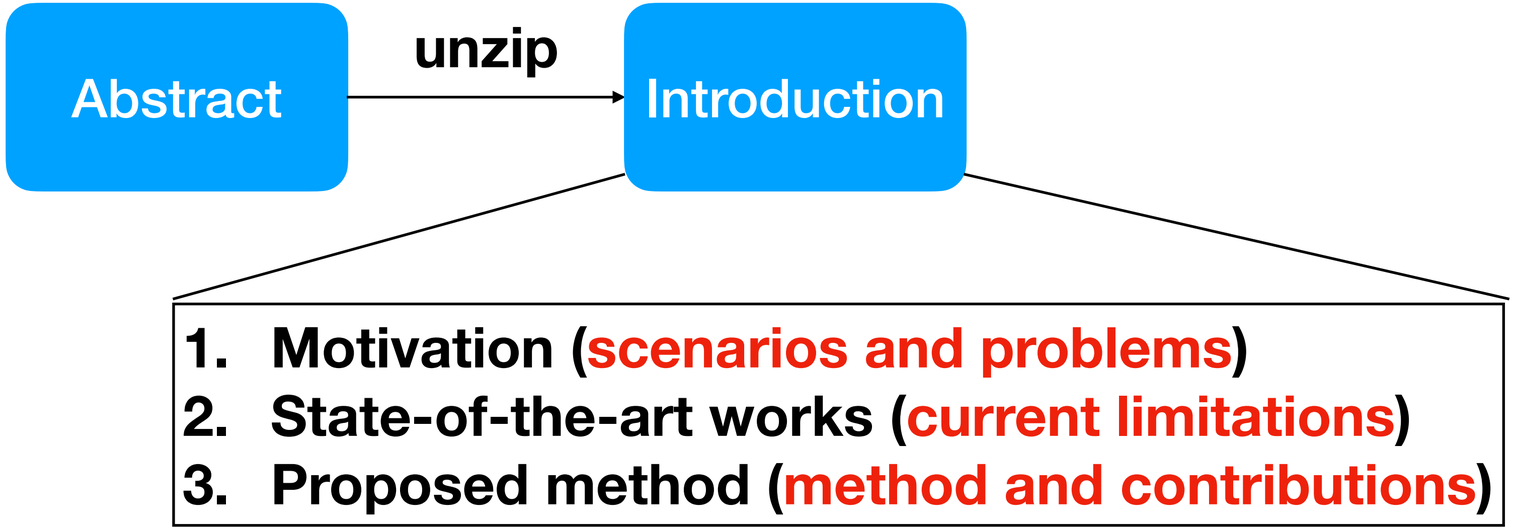
Paper Reading



Paper Reading



Paper Reading



Situation
Complication
Question
Answer

Paper Reading

Abstract → unzip → Introduction

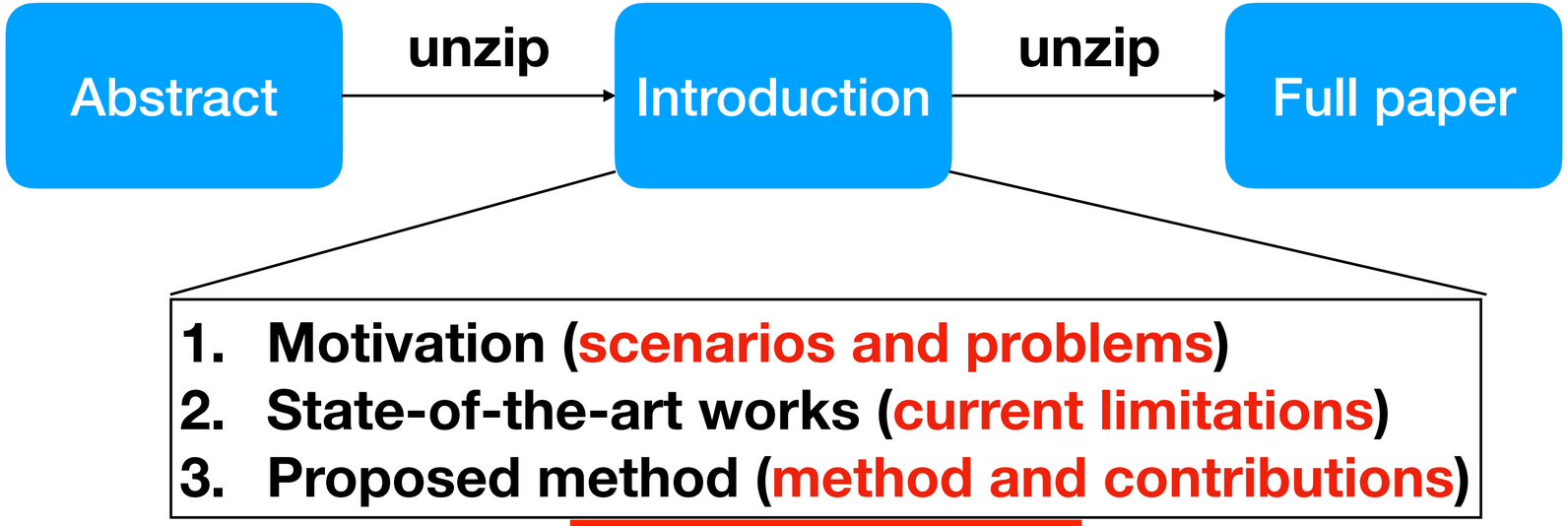
1. Motivation (scenarios and problems)
2. State-of-the-art works (current limitations)
3. Proposed method (method and contributions)

Situation
Complication
Question
Answer

→

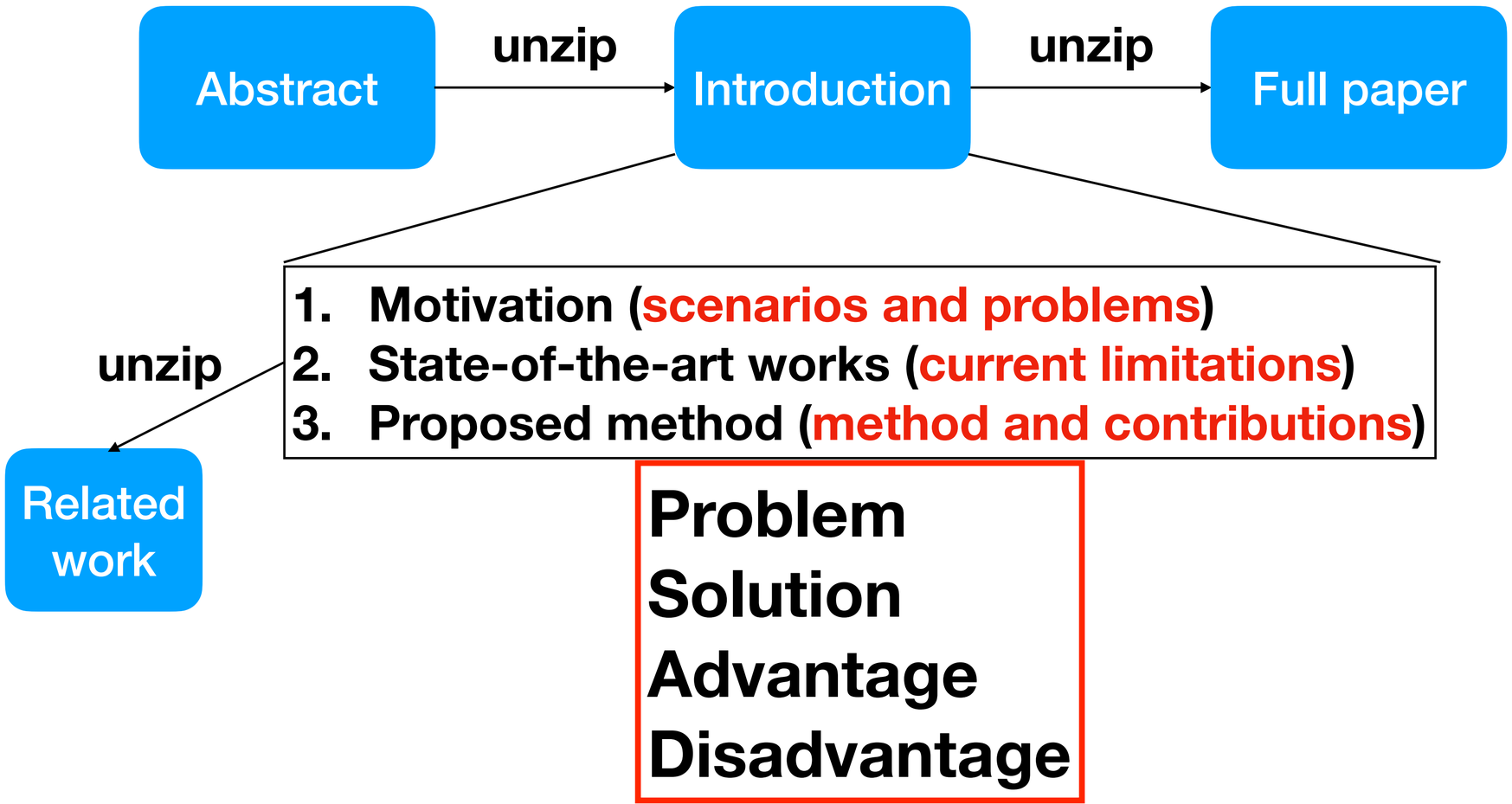
Problem
Solution
Advantage
Disadvantage

Paper Reading

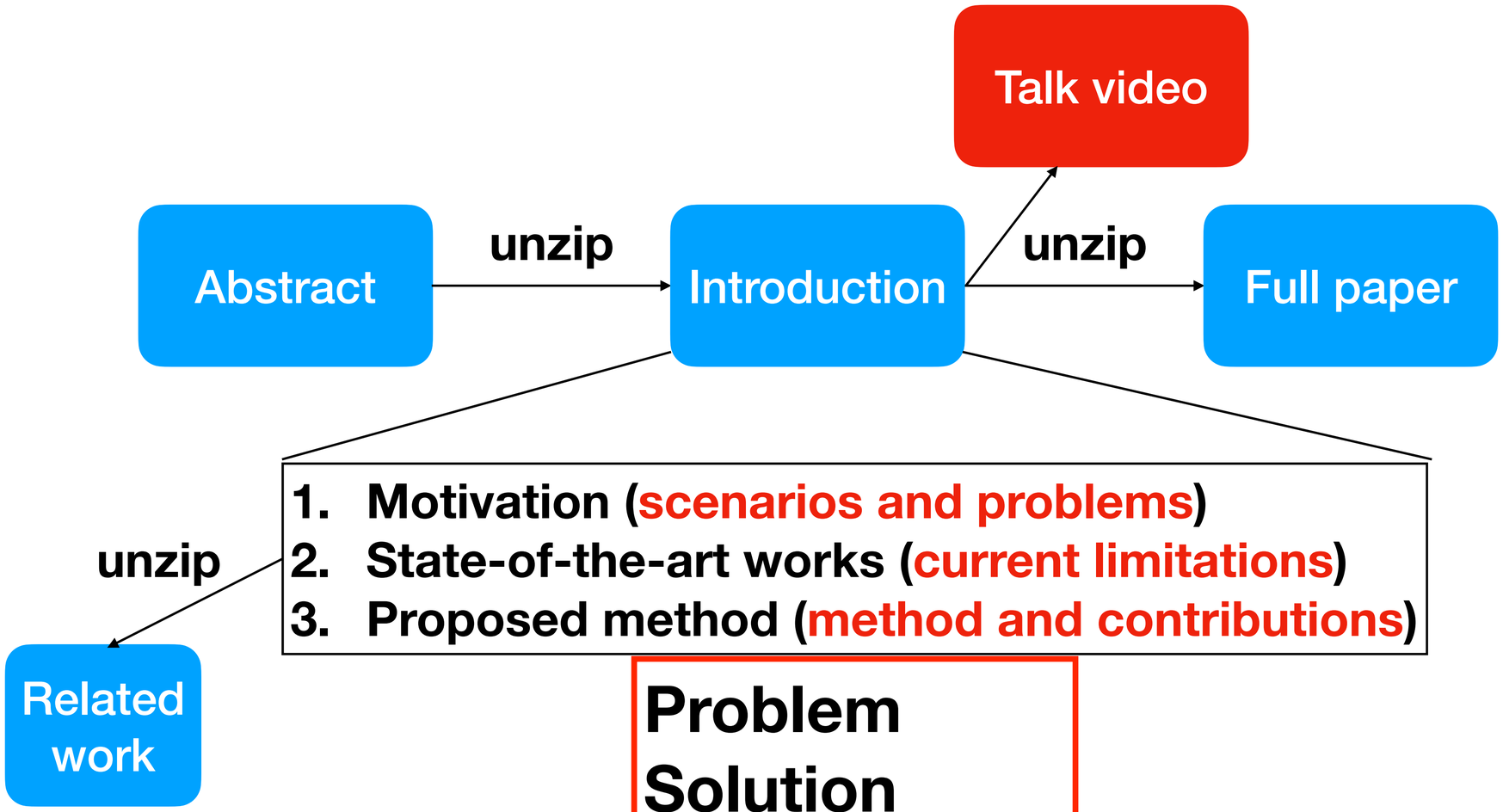


Problem
Solution
Advantage
Disadvantage

Paper Reading



Paper Reading



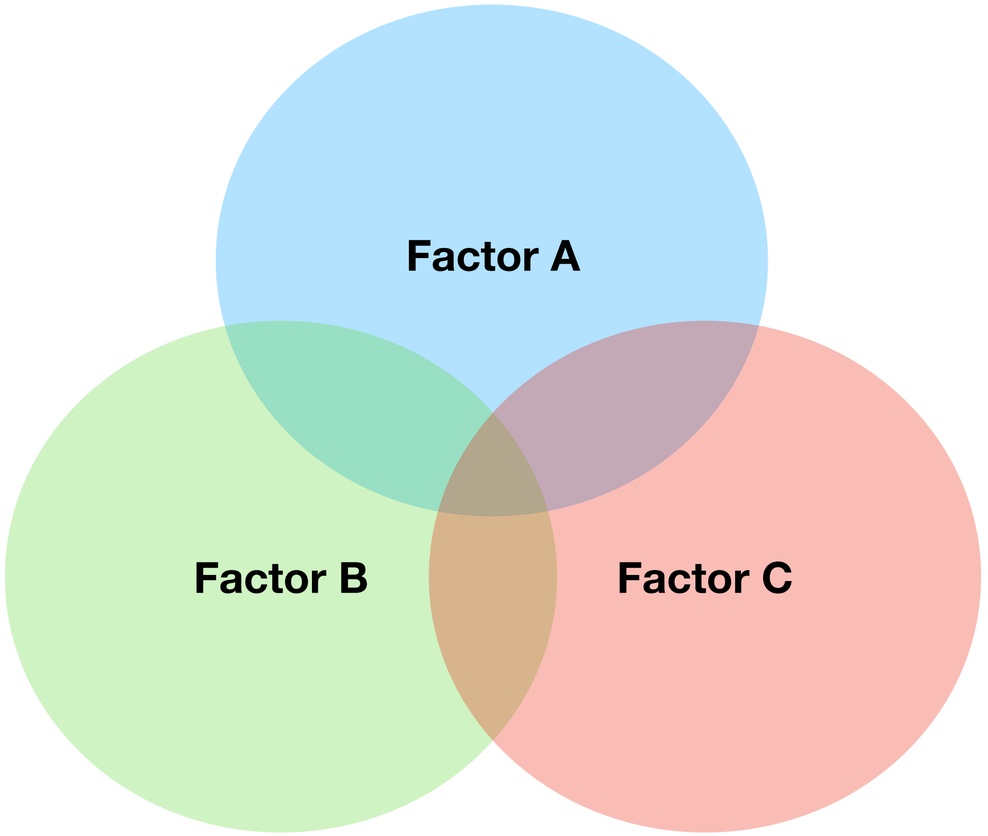
1. Motivation (scenarios and problems)
2. State-of-the-art works (current limitations)
3. Proposed method (method and contributions)

Problem
Solution
Advantage
Disadvantage

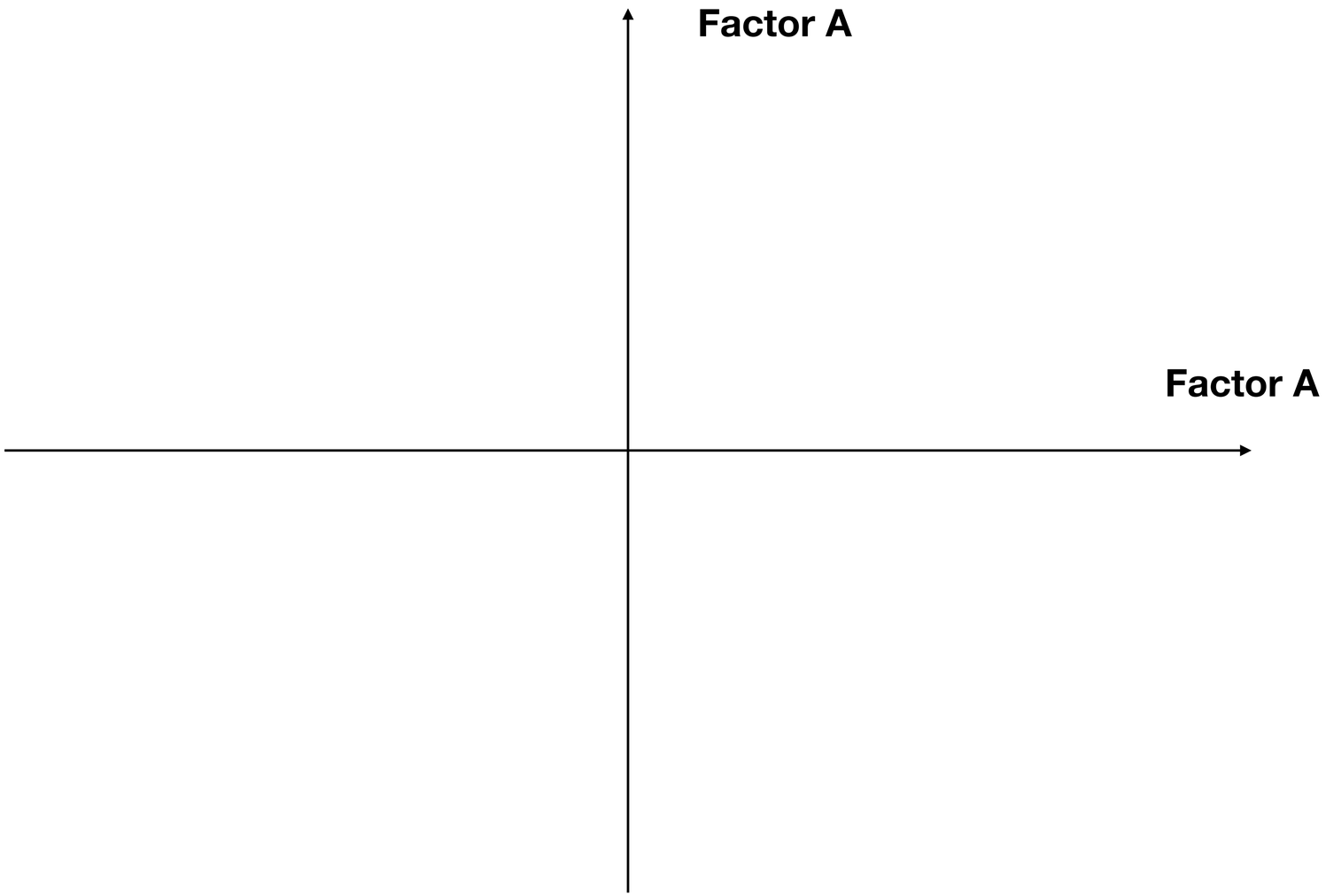
Paper Outline

- Introduction
- Related Work
- Proposed Design/Method
- Experiment/Study/Evaluation
- Limitations and Future Work
- Conclusion

Related Work

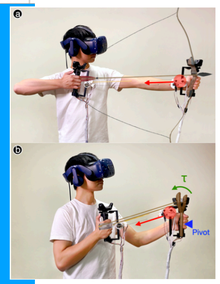


Related Work



Related Work - Category

	A	B	C	D	E	F
1	Name	Feedback type	Device actuator	Body parts	Device form	Device power type
2	ElastImpact					
3	ElastOscillation					
4	Force Jacket					
5	FacePush					
6	LevioPole					
7	Thor's Hammer					
8	Pull-ups					
9	Wind-Blaster					
10	ExoInterfaces					
11	Elastic-Arm					
12	GyroVR					
13	Aarnio					
14	Virtual Walls					
15	ElasticVR					
16	Drag:on					
17	TilePop					
18	PuPoP					
19	DextrES					
20	Dexmo					
21	Wolverine					
22	PoCoPo					
23	inForm					
24	Wireality					
25	SPIDAR					
26	HandMorph					
27	PaCaPa					
28	Haptic Links					



Experiments

- Pilot Study/Pre-Study (to choose proper parameters/factors, adjust the study flow and foresee the results)
- Human Factor Study (human ability)
- Algorithm/Device/System Performance Evaluation (computer ability)
- User Study (human-computer interaction)

Research

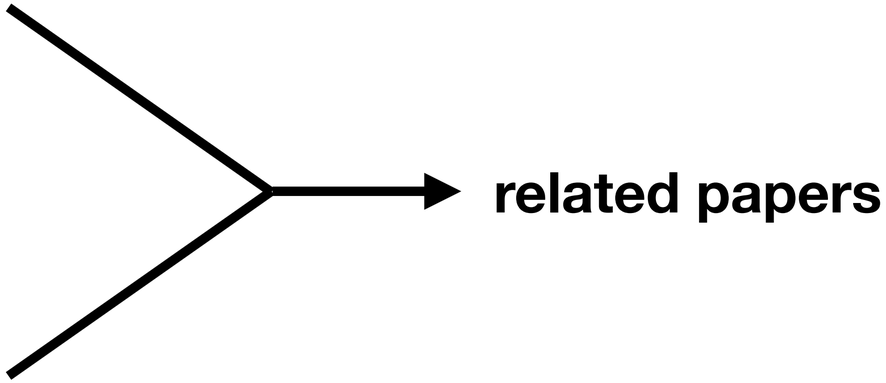
- Research = re + search

How to find papers

- Google Scholar
- ACM Digital Library
- IEEE Xplore Digital Library

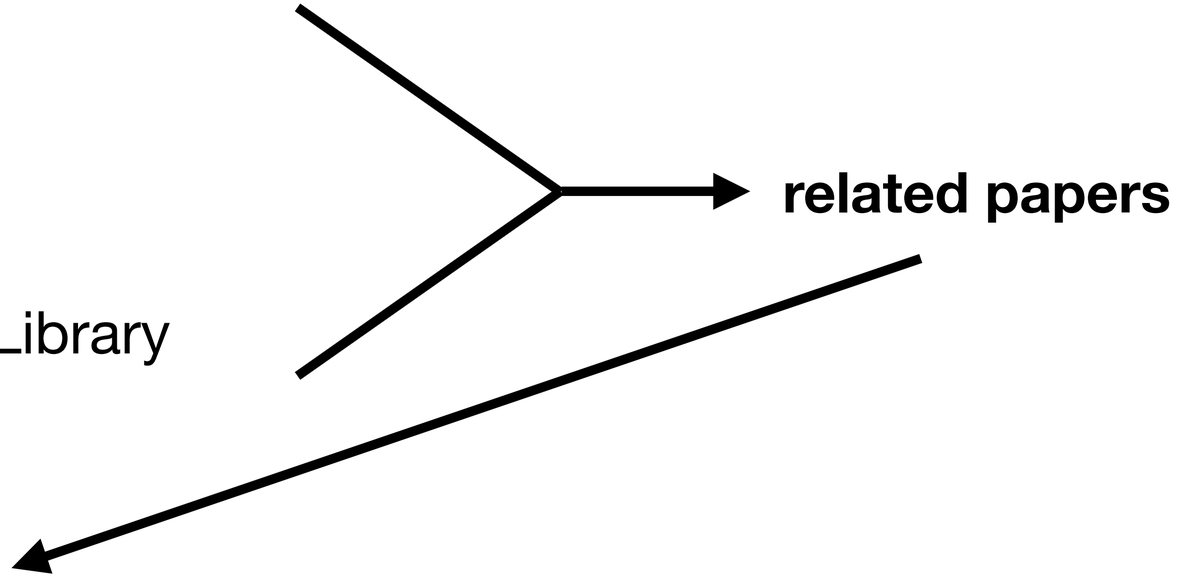
How to find papers

- Google Scholar
- ACM Digital Library
- IEEE Xplore Digital Library



How to find papers

- Google Scholar
- ACM Digital Library
- IEEE Xplore Digital Library
- References
- Citations



Review Policy

2 Associate Chairs (ACs) + 2 external reviewers

Reviews

Reviewer	Role	Score	Review
Reviewer 4	1AC	3.5	review 4
Reviewer 2	2AC	4	review 2
Reviewer 1	reviewer	4	review 1
Reviewer 3	reviewer	2.5	review 3

Reviews

Reviewer	Role	Score	Review
Reviewer 4	AC	3.5	review 4
Reviewer 3	2AC	4	review 3
Reviewer 1	reviewer	4	review 1
Reviewer 2	reviewer	3.5	review 2

Score the works in 5 points
(1: strongly reject, 2: reject, 3: borderline, 4: accept, 5: strongly accept)

Evaluating a paper

- Novel idea (innovative, insightful)
- Technical contribution (novel techniques, difficult to implement)
- Complete studies (deep investigating, insightful results)

Review

- Summarizing in 1~2 (3 at most) concise sentences to describe the work and contributions
- Evaluating based on each section or each contribution
- What’s the differences between this work and the 1~3 closest works
- Finding drawbacks or logic flaws
- **Overclaiming is unacceptable**
- Scoring based on the comments

Review Assignment Form (1/3)

1. Score: 1~5 (1: strongly reject, 2: reject, 3: borderline, 4: accept, 5: strongly accept)
2. Summary: 1~2 (3 at most) concise sentences to describe the work and contributions
3. Problem: the problems/motivations this work solves, why they are important
4. Solution: the solution or method proposed in the work
5. Advantage: the advantages of the proposed method
6. Disadvantage: the disadvantages of the proposed method
7. Future direction: the possible future directions, ideas inspired by this work, or important research questions not discussed

Review Assignment Form (2/3)

- Review content:

  summary, flaws in each section, contributions

Review Assignment Form (3/3)

- Categorize close related works in 2~4 groups (in an excel file with brief introduction)

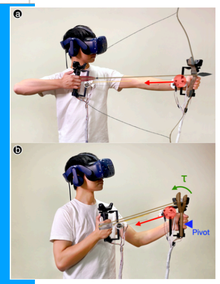
- Compare the difference:

  (this work & the closest work) v.s. (the closest work & its previous closest work)

  (W_n & W_{n-1}) v.s. (W_{n-1} & W_{n-2})

Related Work - Category

	A	B	C	D	E	F
1	Name	Feedback type	Device actuator	Body parts	Device form	Device power type
2	ElastImpact					
3	ElastOscillation					
4	Force Jacket					
5	FacePush					
6	LevioPole					
7	Thor's Hammer					
8	Pull-ups					
9	Wind-Blaster					
10	ExoInterfaces					
11	Elastic-Arm					
12	GyroVR					
13	Aarnio					
14	Virtual Walls					
15	ElasticVR					
16	Drag:on					
17	TilePop					
18	PuPoP					
19	DextrES					
20	Dexmo					
21	Wolverine					
22	PoCoPo					
23	inForm					
24	Wireality					
25	SPIDAR					
26	HandMorph					
27	PaCaPa					
28	Haptic Links					



Paper pool

- transPAF: Rendering Omnidirectional Impact Feedback with Dynamic Point of Application of Force All Round a Controller, CHI 2023 (https://dl.acm.org/doi/abs/10.1145/3544548.3581092)
- QuadStretcher: A Forearm-Worn Skin Stretch Display for Bare-Hand Interaction in AR/VR, CHI 2024 (https://dl.acm.org/doi/10.1145/3613904.3642067)
- PaCaPa: A Handheld VR Device for Rendering Size, Shape, and Stiffness of Virtual Objects in Tool-based Interactions, CHI 2019 (https://dl.acm.org/doi/10.1145/3290605.3300682)
- Wireality: Enabling Complex Tangible Geometries in Virtual Reality with Worn Multi-String Haptics, CHI 2020 (https://dl.acm.org/doi/abs/10.1145/3313831.3376470)
- GuideBand: Intuitive 3D Multilevel Force Guidance on a Wristband in Virtual Reality, CHI 2021 (https://dl.acm.org/doi/10.1145/3411764.3445262)
- Touch&Fold: A Foldable Haptic Actuator for Rendering Touch in Mixed Reality, CHI 2021 (https://dl.acm.org/doi/10.1145/3411764.3445099)
- GamesBond: Bimanual Haptic Illusion of Physically Connected Objects for Immersive VR Using Grip Deformation, CHI 2021 (https://dl.acm.org/doi/10.1145/3411764.3445727)
- FingerX: Rendering Haptic Shapes of Virtual Objects Augmented by Real Objects using Extendable and Withdrawable Supports on Fingers, CHI 2022 (https://dl.acm.org/doi/10.1145/3491102.3517489)
- Altering Perceived Softness of Real Rigid Objects by Restricting Fingerpad Deformation, UIST 2021 (https://dl.acm.org/doi/10.1145/3472749.3474800)

Arduino Installation

Go to: https://www.arduino.cc/en/Main/Software

Download the Arduino IDE

ARDUINO 1.8.10

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board.
Refer to the Getting Started page for Installation instructions.

Windows Installer, for Windows XP and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10
Get

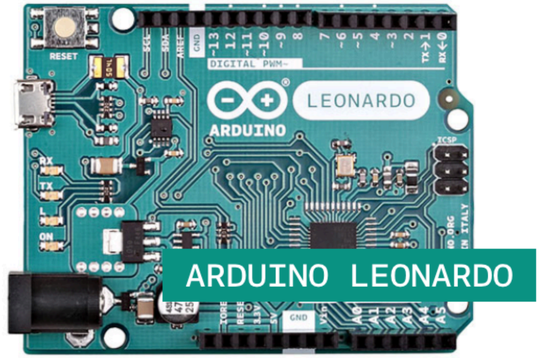
Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

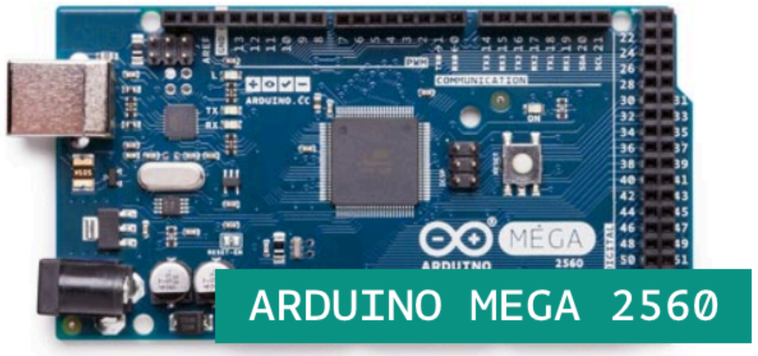
Release Notes
Source Code
Checksums (sha512)

Arduino

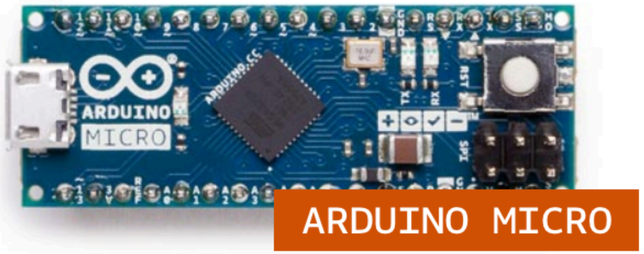
ARDUINO UNO



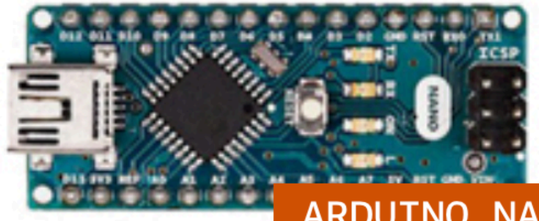
ARDUINO LEONARDO



ARDUINO MEGA 2560



ARDUINO MICRO

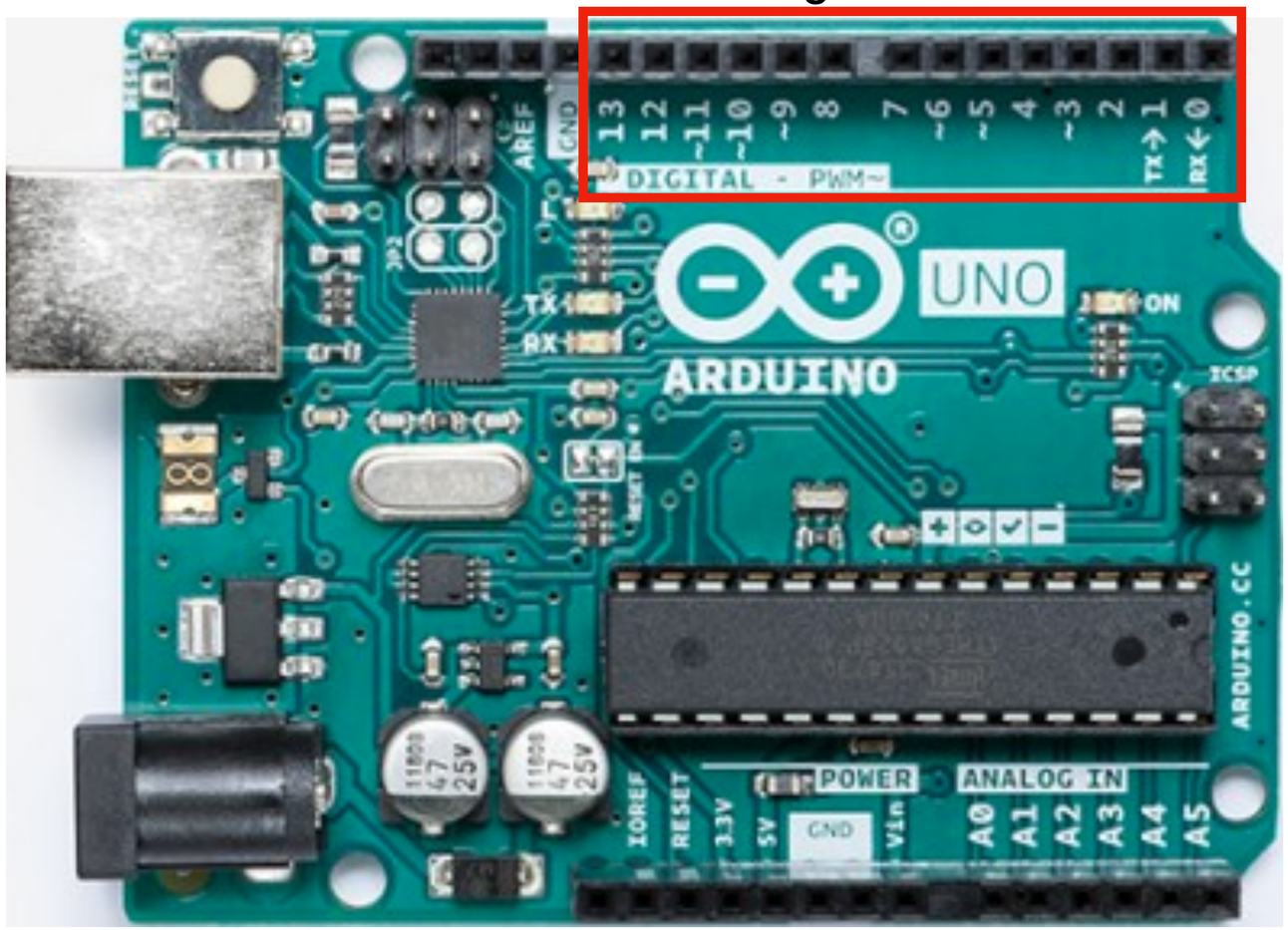


ARDUINO NANO

[arduino.cc]

Arduino

Digital I/O Pins



Arduino

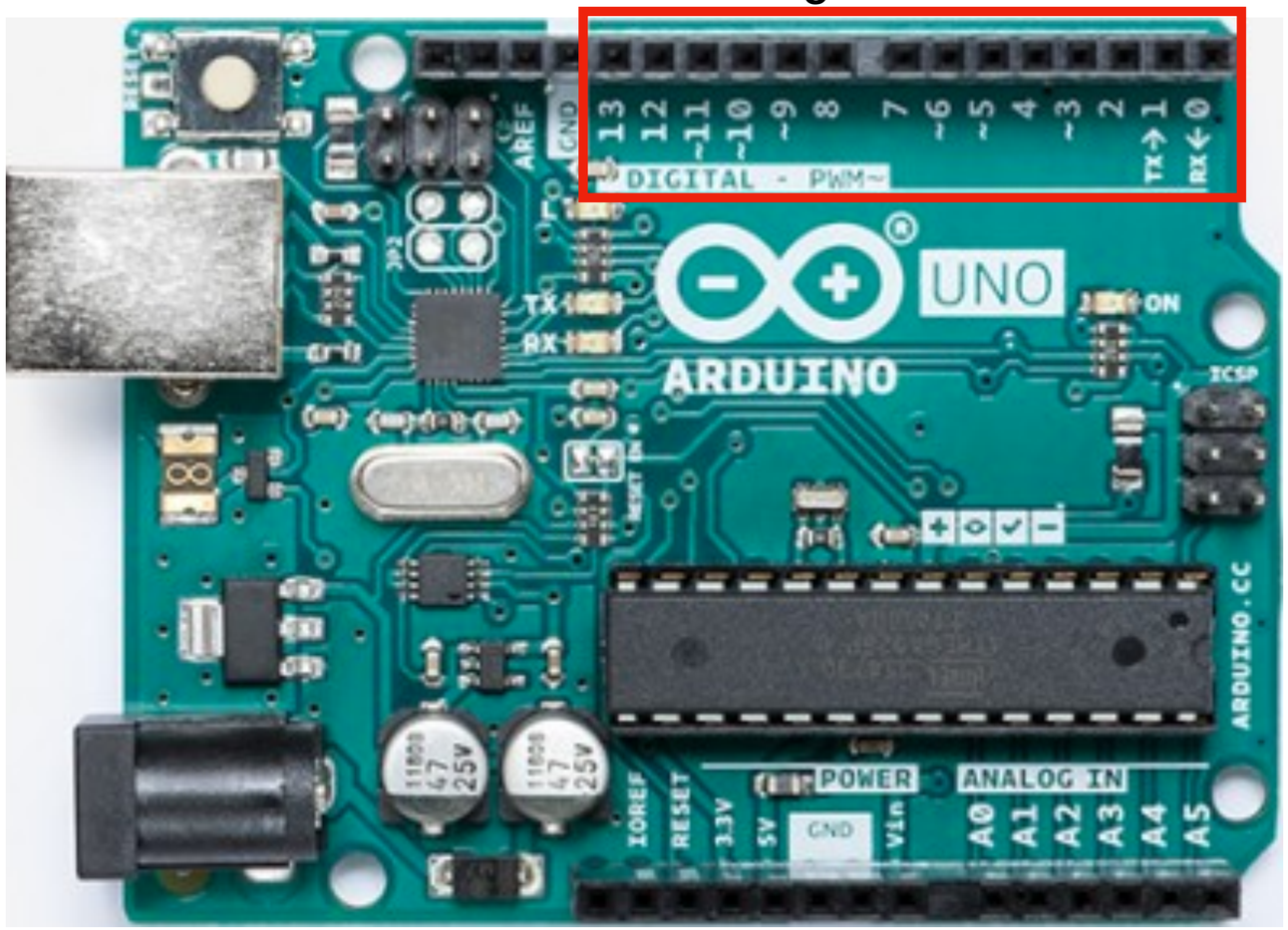
Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Digital I/O Pins



Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

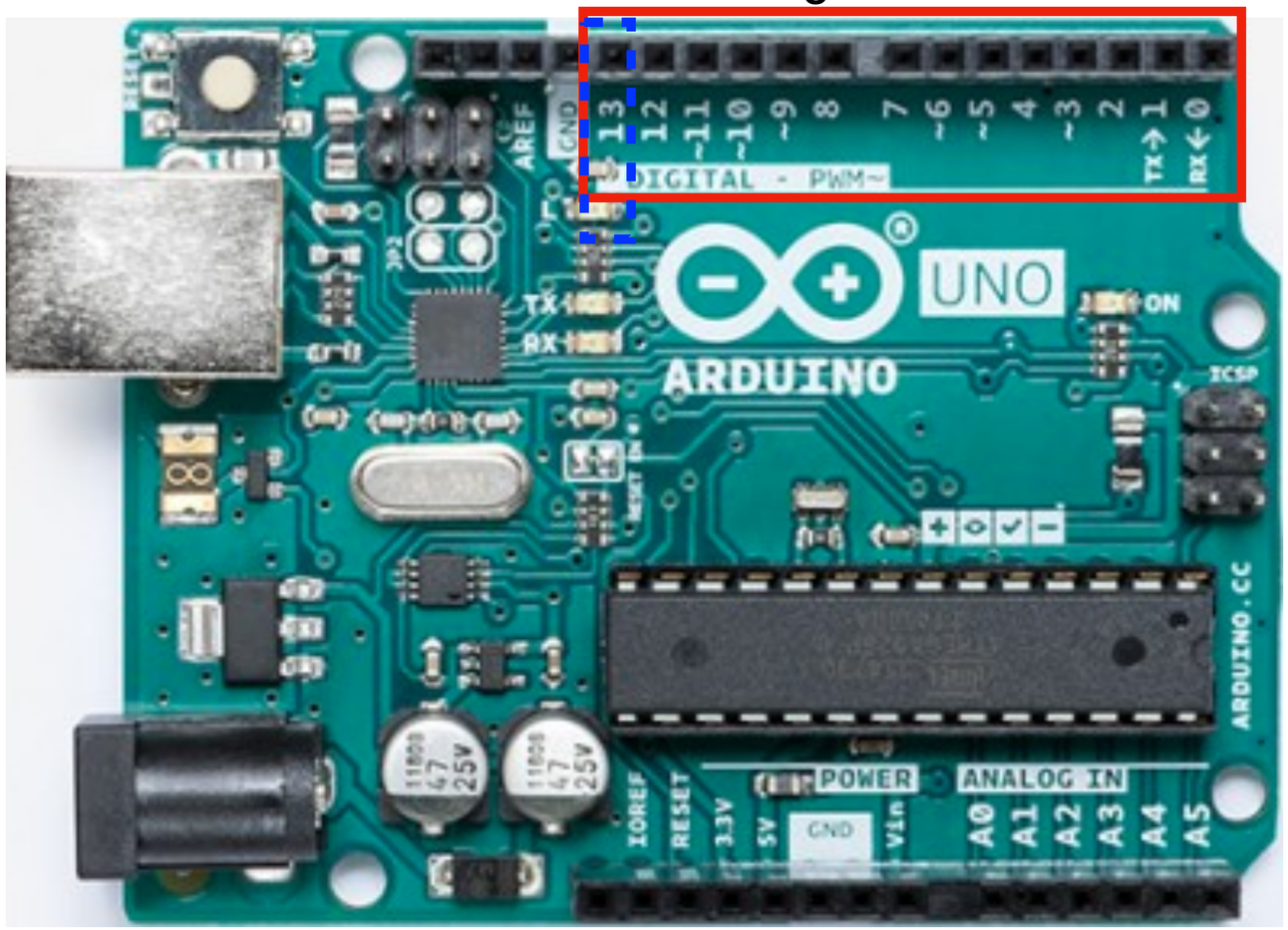
[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

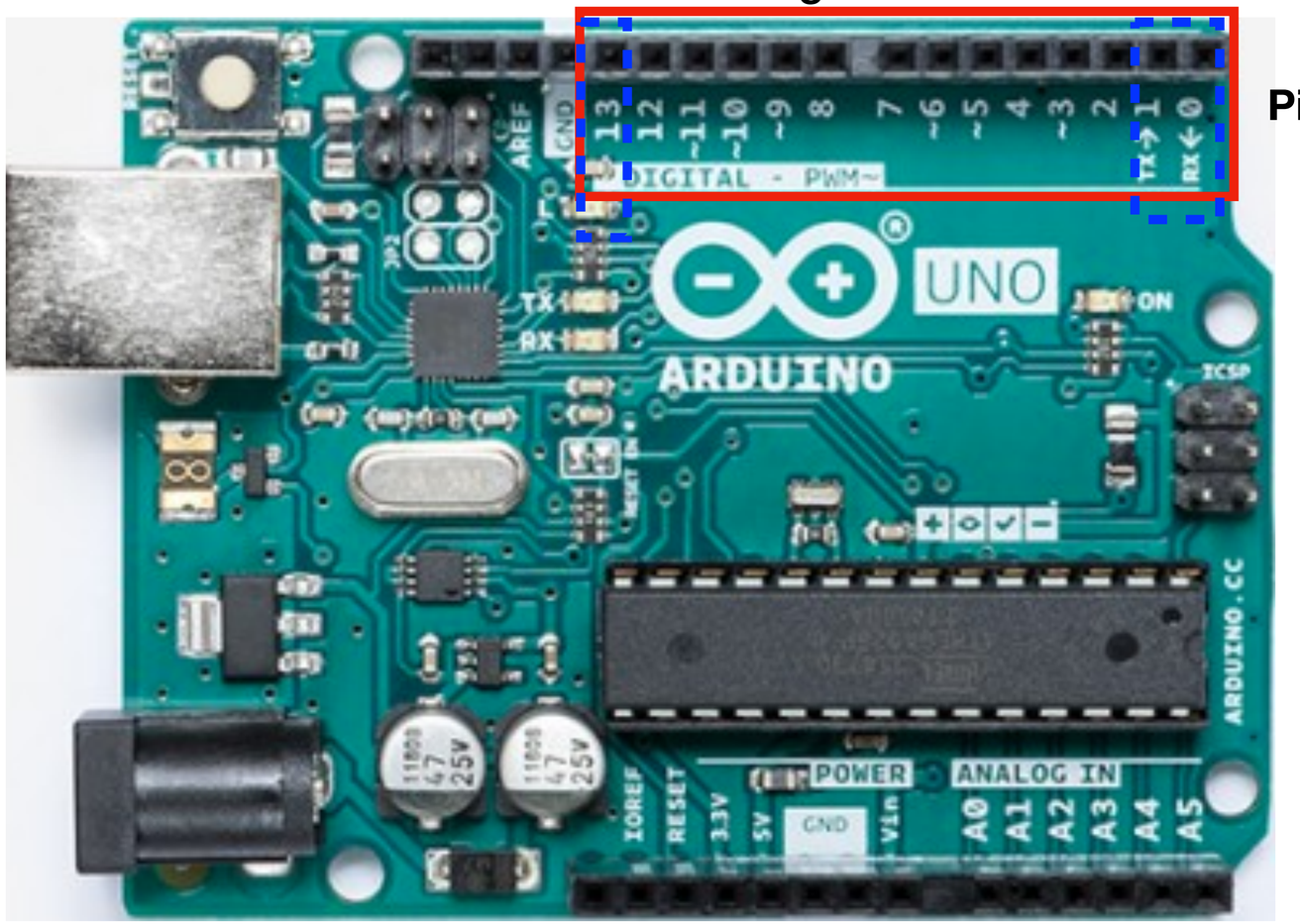
[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



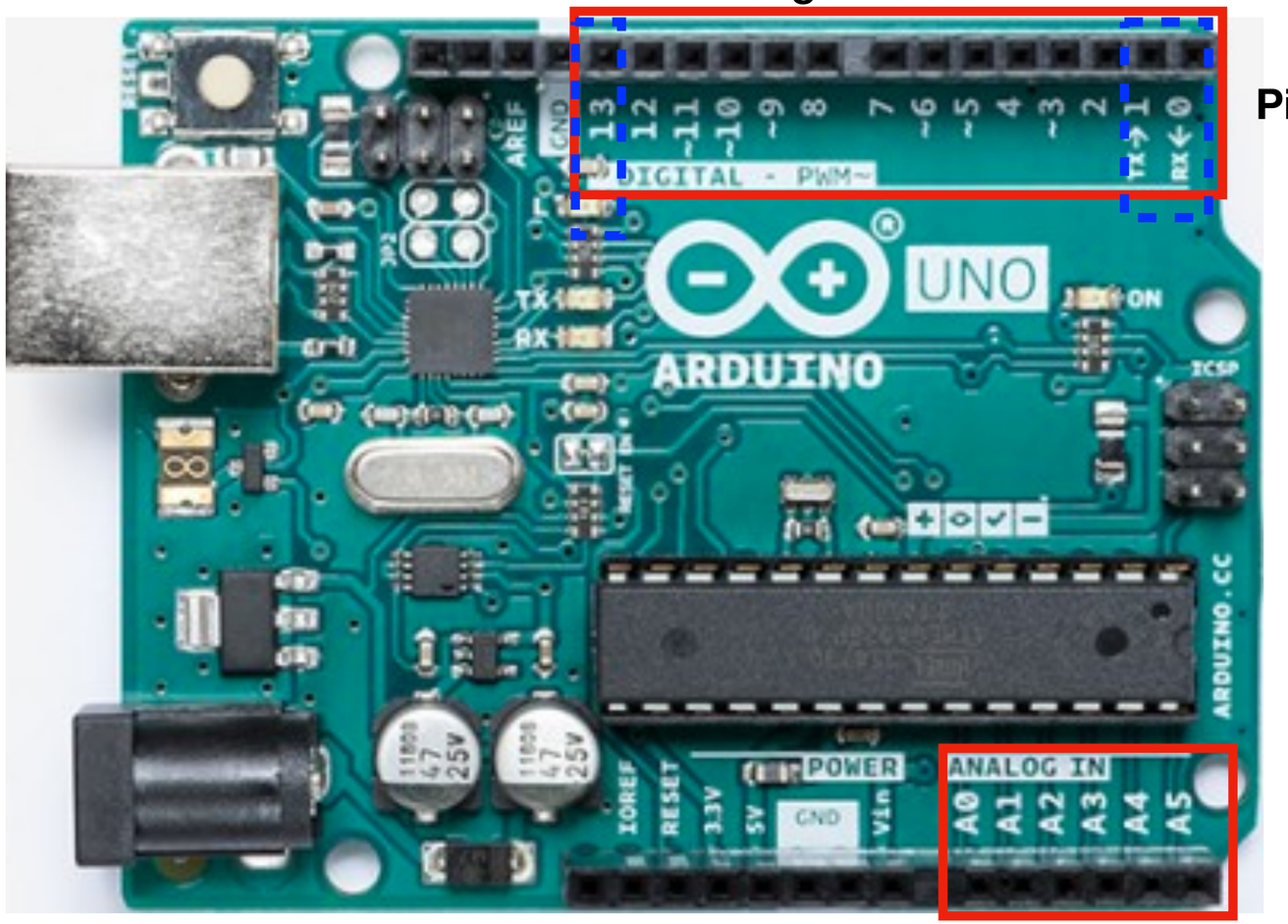
Pin 0, 1: Rx, Tx

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

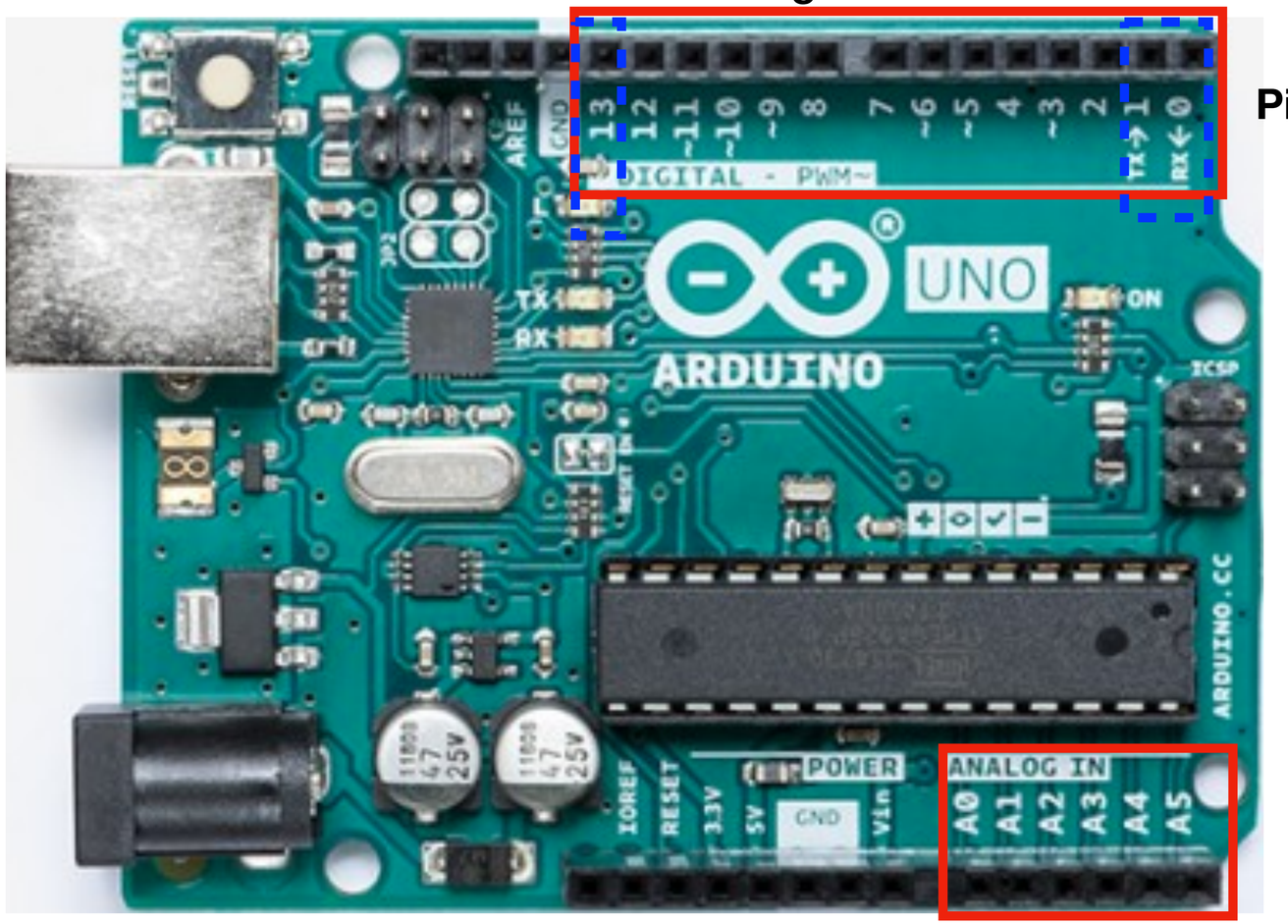
Analog Input Pins
(0~1023)

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

Analog-to-Digital
Converter (ADC)

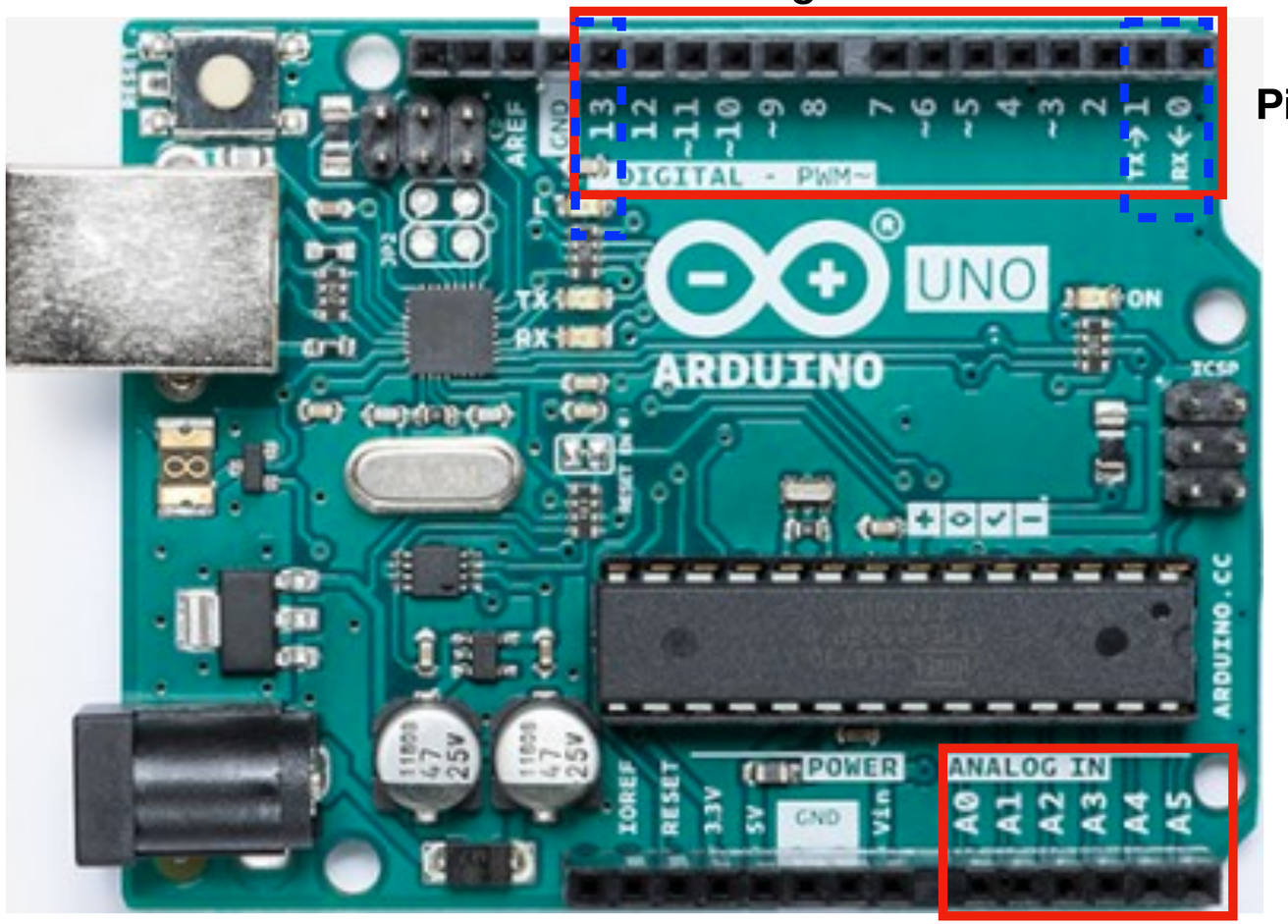
Analog Input Pins
(0~1023)

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

Analog-to-Digital
Converter (ADC)

Analog Input Pins & Digital I/O Pins
(0~1023)

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

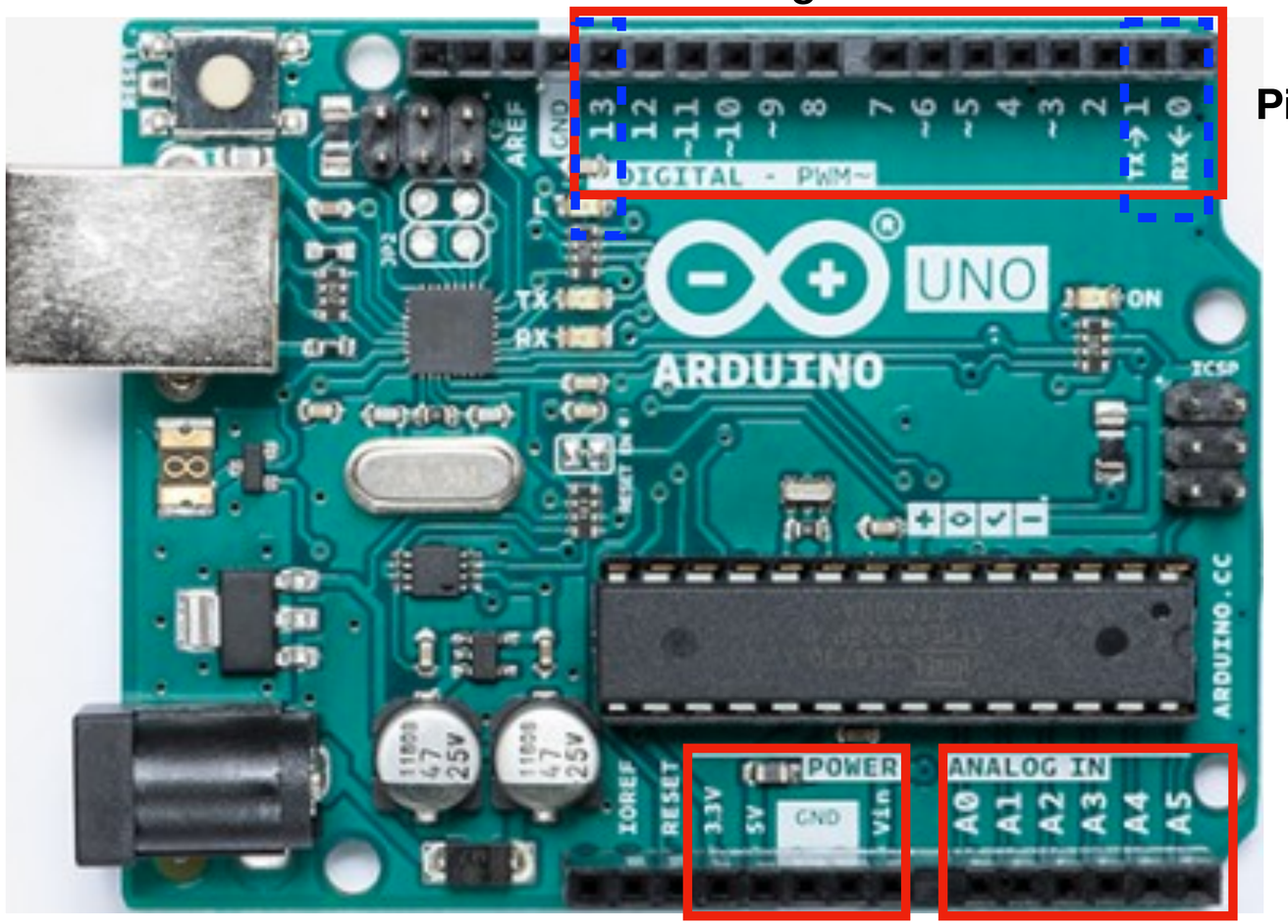
[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

Analog-to-Digital
Converter (ADC)

Power
5V/3.3V, GND

Analog Input Pins
(0~1023)

& Digital I/O Pins

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

[arduino.cc]

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

[arduino.cc]

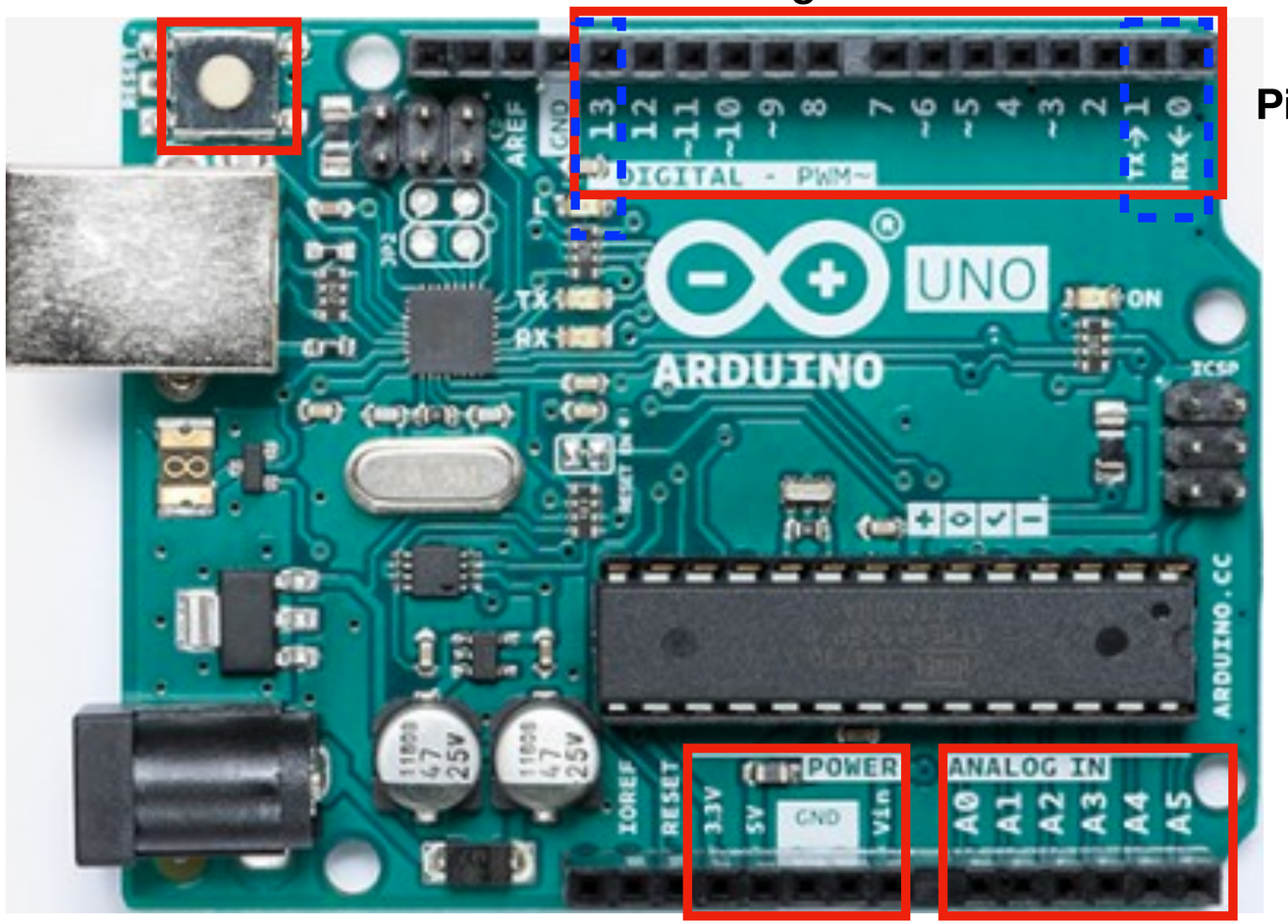
Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Reset

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

Analog-to-Digital
Converter (ADC)

Power
5V/3.3V, GND

Analog Input Pins
(0~1023)

& Digital I/O Pins

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

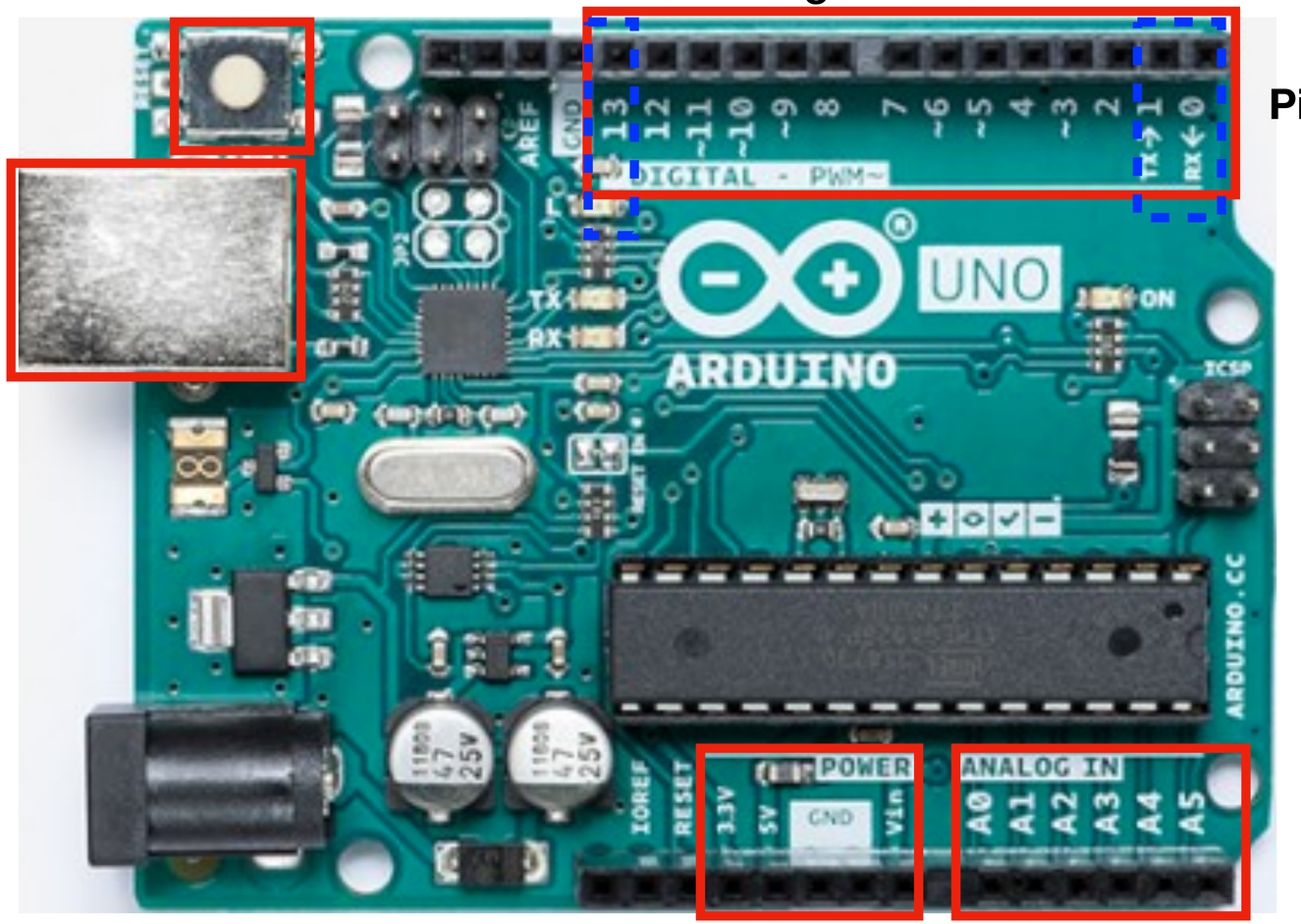
Reset

Pin 13: LED

Digital I/O Pins

Pin 0, 1: Rx, Tx

USB
Port



Analog-to-Digital
Converter (ADC)

Power
5V/3.3V, GND

Analog Input Pins
(0~1023)

& Digital I/O Pins

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

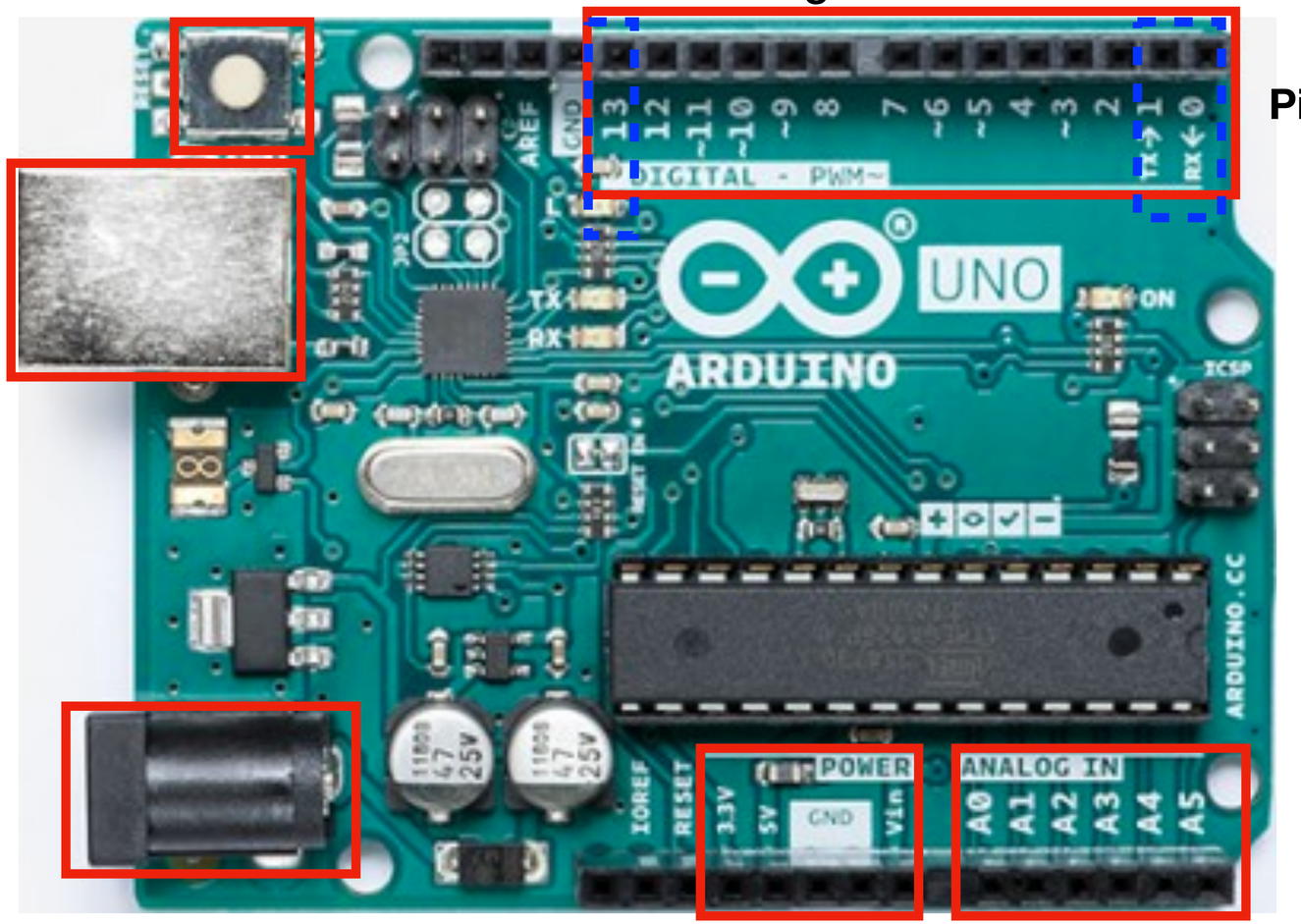
Reset

Pin 13: LED

Digital I/O Pins

Pin 0, 1: Rx, Tx

USB
Port



7~12V
Input

Analog-to-Digital
Converter (ADC)

Power
5V/3.3V, GND

Analog Input Pins
(0~1023)

& Digital I/O Pins

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

[arduino.cc]

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

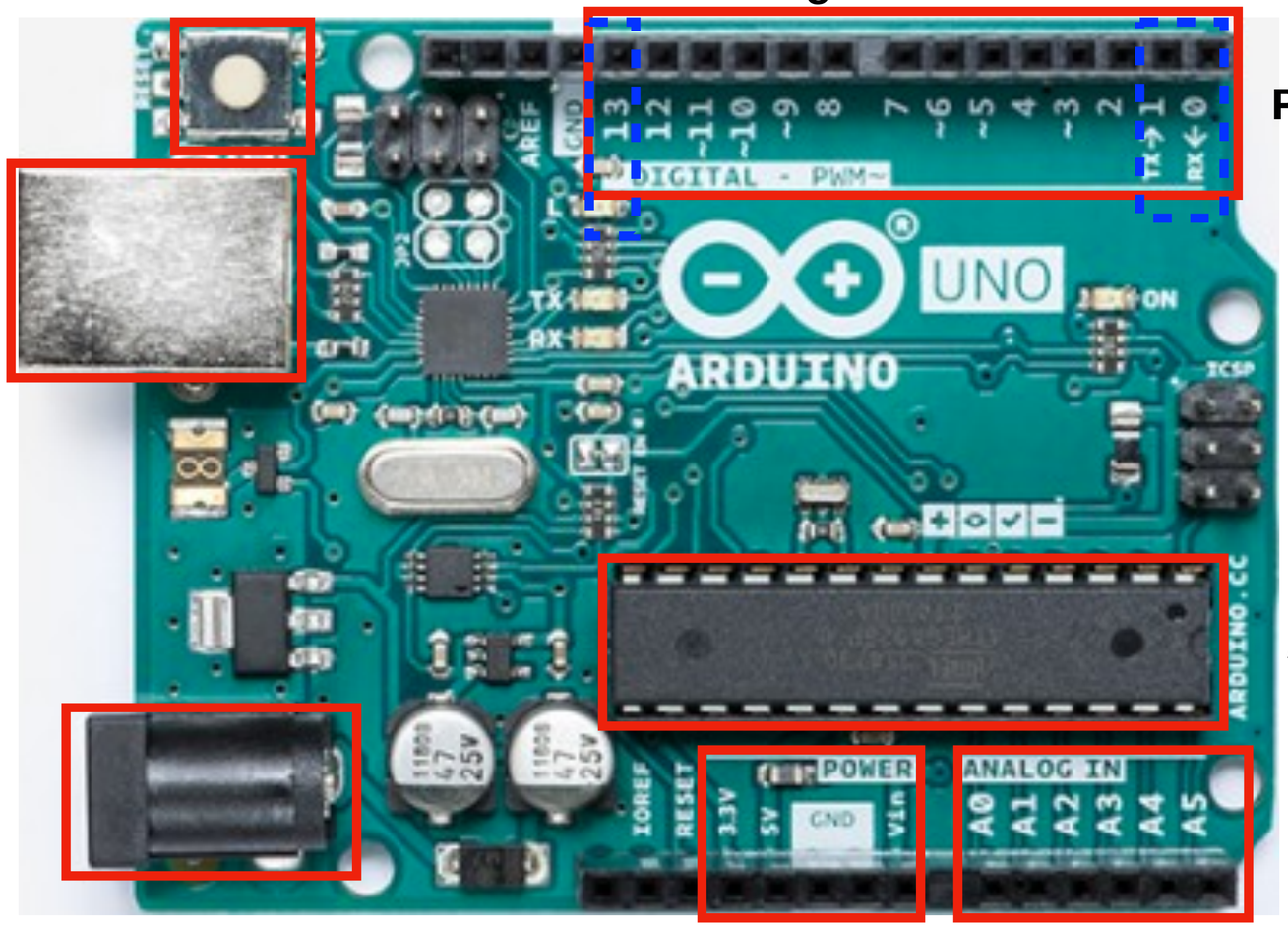
Reset

Pin 13: LED

Digital I/O Pins

Pin 0, 1: Rx, Tx

USB
Port



ATmega328P

7~12V
Input

Analog-to-Digital
Converter (ADC)

Power
5V/3.3V, GND

Analog Input Pins
(0~1023)

& Digital I/O Pins

Arduino

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

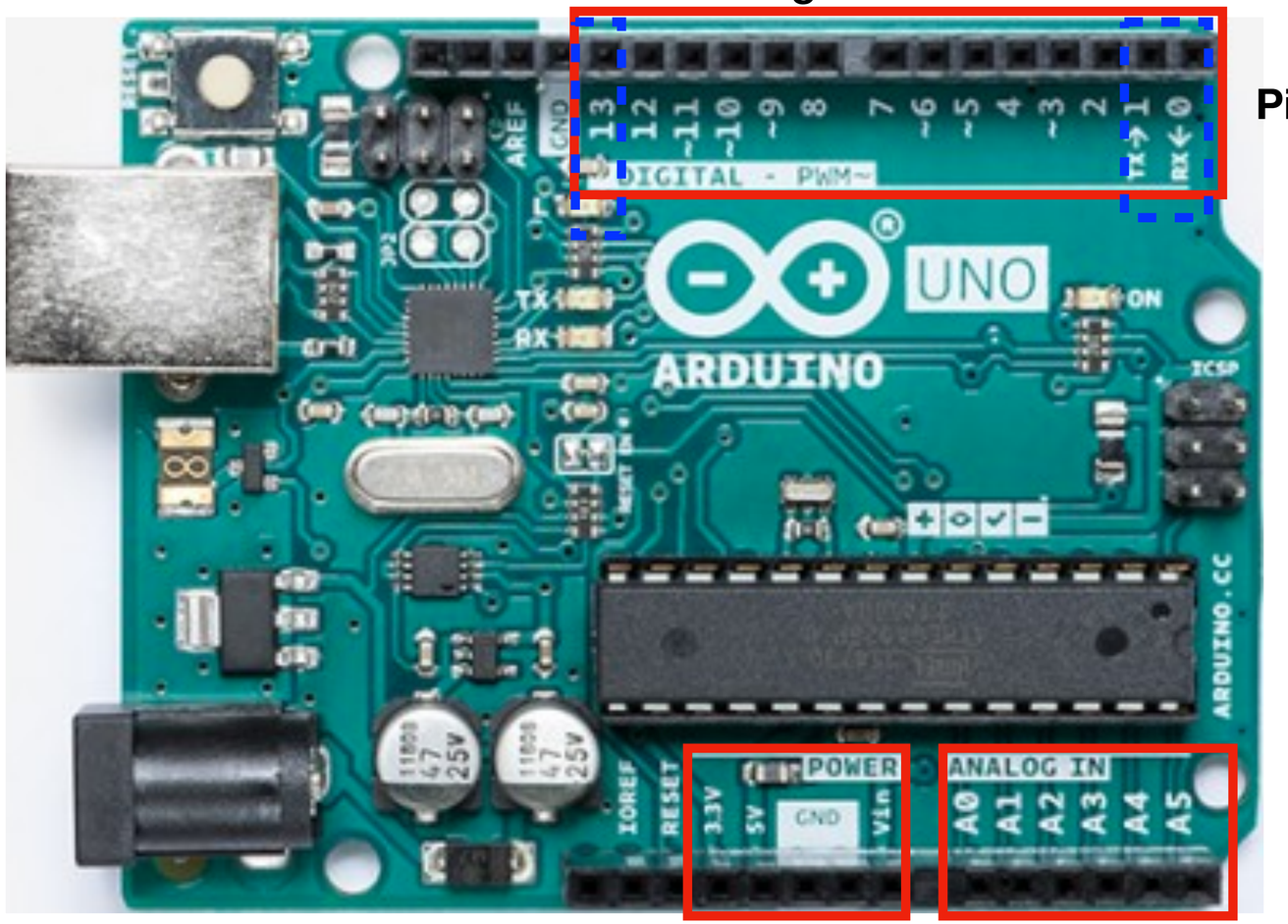
[arduino.cc]

Arduino

~: PWM Pins (3, 5, 6, 9,10, 11)
(0~255)

Pin 13: LED

Digital I/O Pins



Pin 0, 1: Rx, Tx

Power
5V/3.3V, GND

Analog Input Pins & Digital I/O Pins
(0~1023)

Arduino and Raspberry Pi Comparison

Arduino	Raspberry Pi
One can build own Arduino from scratch or with specifications as per one's need	Development environment includes anything with Linux support
Less powerful clock speed (16 MHz)	Significantly more powerful (900 MHz)
Projects are not made to run videos	Has graphics port and can run videos
It can run only one program at a time	Multitasking in a program is possible
0.002MB and no external memory support	512MB with external memory support
Large support network online to assist	Has a more fragmented community
Arduino UNO R3 board cost < Rs1500	Raspberry Pi B costs around Rs3000

[https://bit.ly/3YbIYAb]

Raspberry Pi	Arduino
Microcomputer	Microcontroller
Needs an operating system	Does not need an operating system
Complicated	Simple
Video out, Camera, Ethernet ports, Wifi, Bluetooth, USB, I2C, SPI, UART etc. on board	USB only for power and serial in/out, I2C, SPI, UART
Best for general computer	Best for small tasks that constantly repeat
Capable of performing a huge range of tasks	Optimised for sensing and controlling the world around it
Best for more advanced makers	Best for beginners
Programmed in many languages, including C/C++, Python, Ruby	Programmed in C/C++
Relatively high power consumption	Relatively low power consumption

[https://bit.ly/3IATnzD]

Arduino Installation

Done?

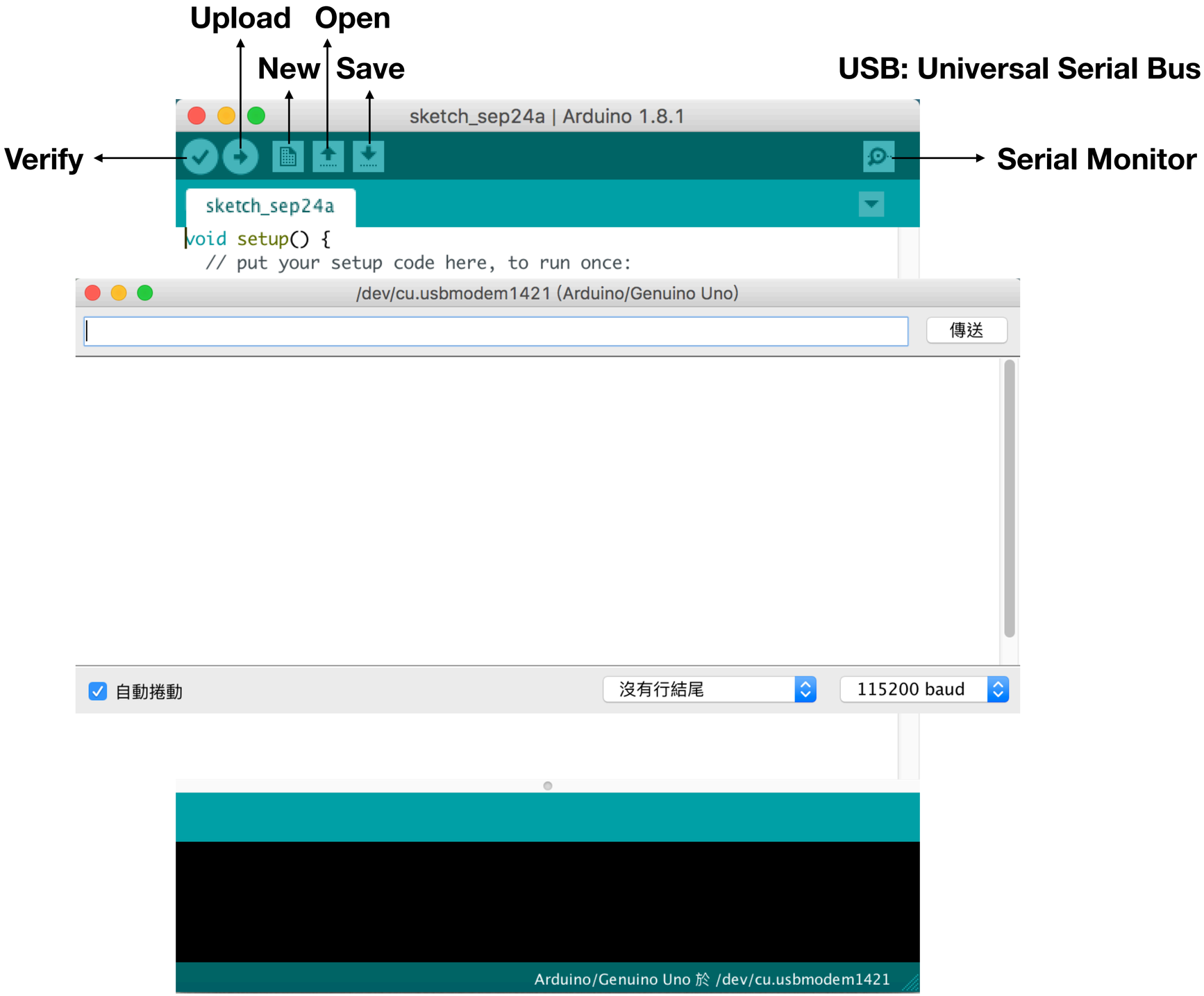
sketch_sep24a | Arduino 1.8.1

sketch_sep24a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

Upload
Open
New
Save
USB: Universal Serial Bus
Verify
Serial Monitor
sketch_sep24a | Arduino 1.8.1
sketch_sep24a
void setup() {
// put your setup code here, to run once:
}
void loop() {
// put your main code here, to run repeatedly:
}
Arduino/Genuino Uno 於 /dev/cu.usbmodem1421


Upload
Open
New
Save
USB: Universal Serial Bus
Verify
Serial Monitor
sketch_sep24a | Arduino 1.8.1
sketch_sep24a
void setup() {
// put your setup code here, to run once:
/dev/cu.usbmodem1421 (Arduino/Genuino Uno)
傳送
自動捲動
沒有行結尾
115200 baud
Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

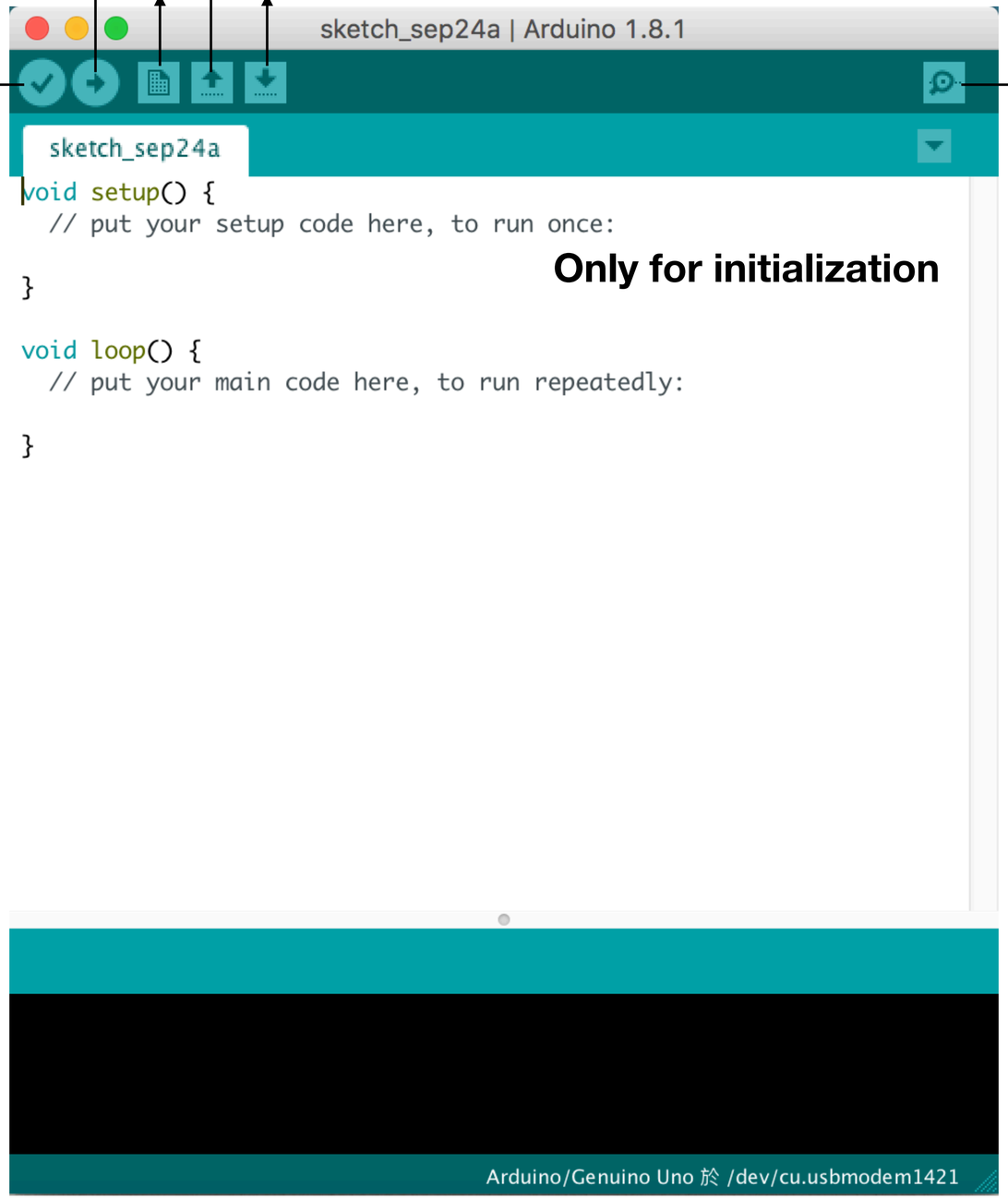
Upload Open

New Save

USB: Universal Serial Bus

Verify

Serial Monitor



Only for initialization

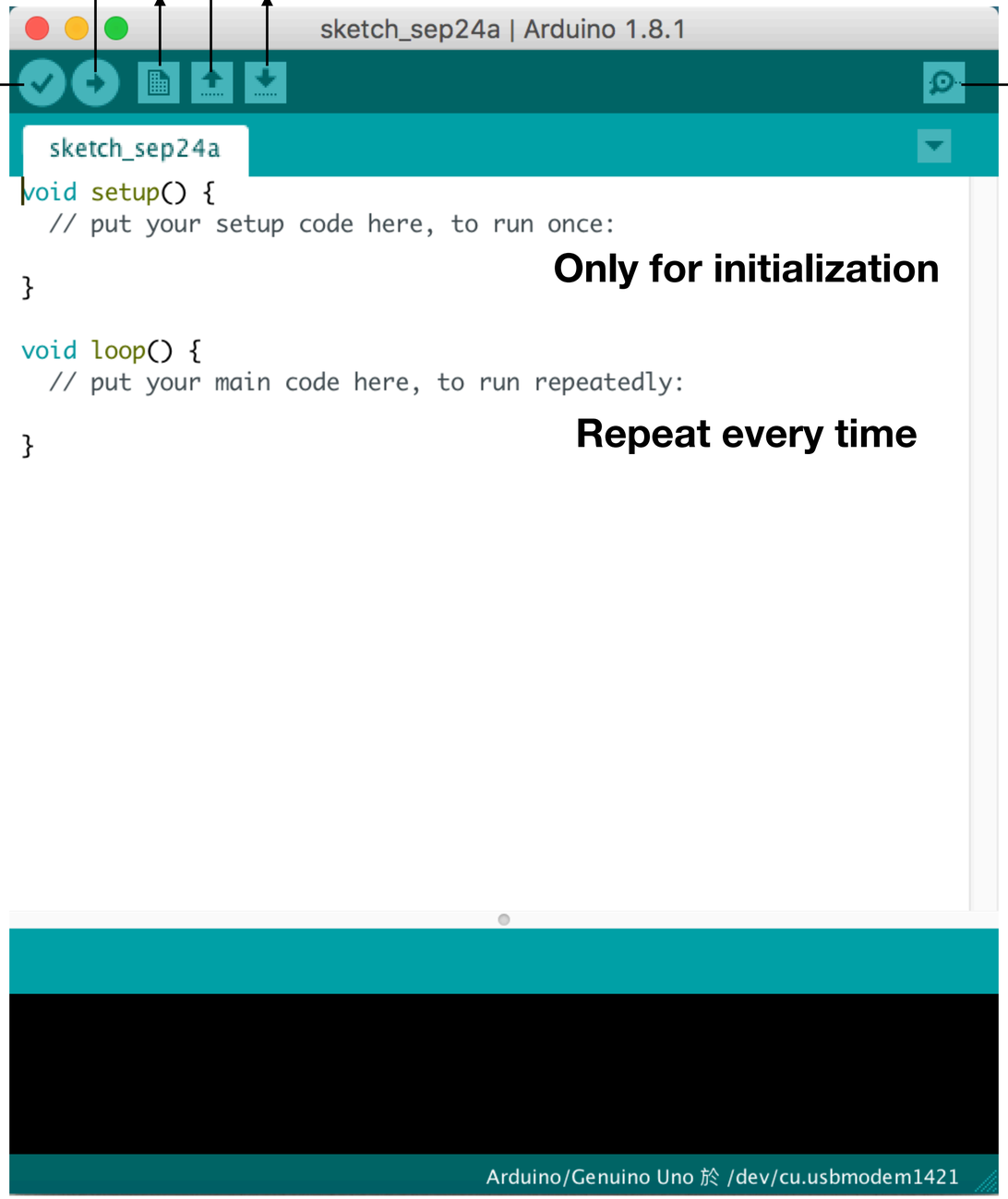
Upload Open

New Save

USB: Universal Serial Bus

Verify

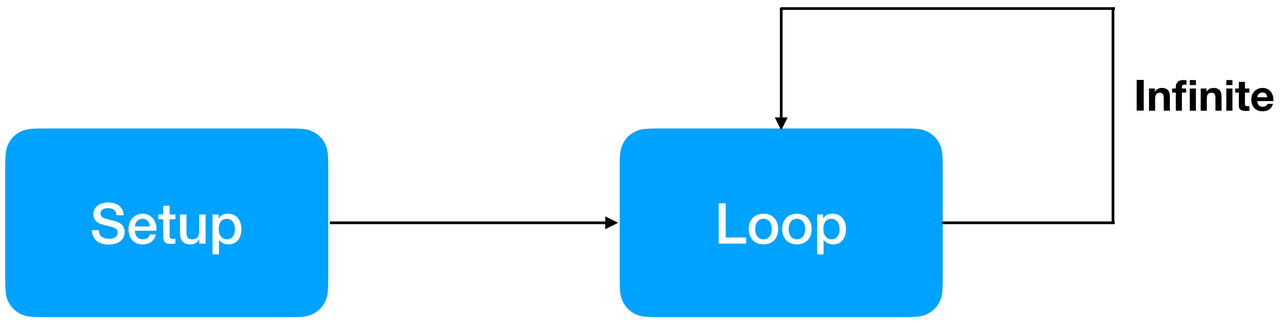
Serial Monitor



Only for initialization

Repeat every time

Flow Chart



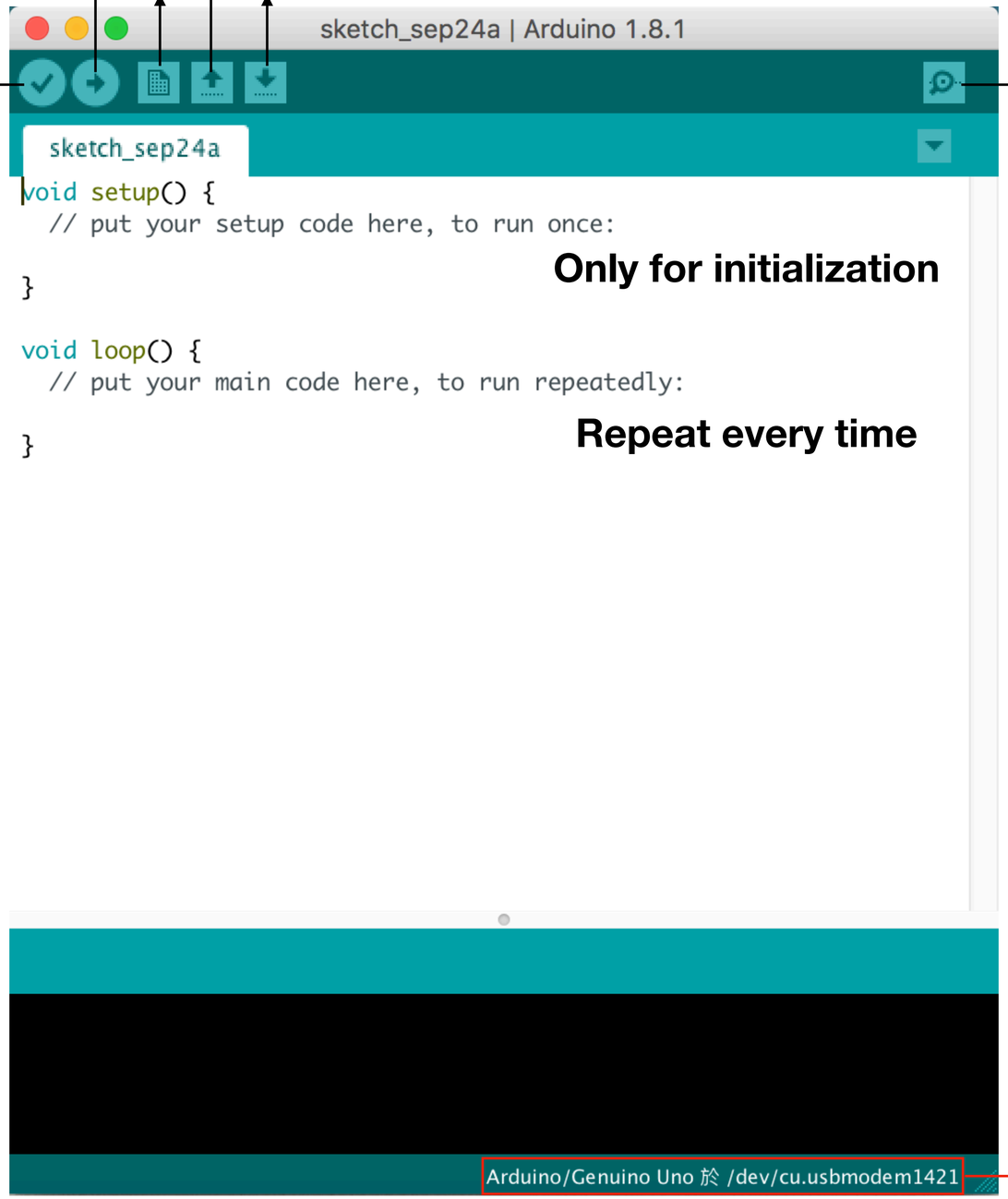
Upload Open

New Save

USB: Universal Serial Bus

Verify

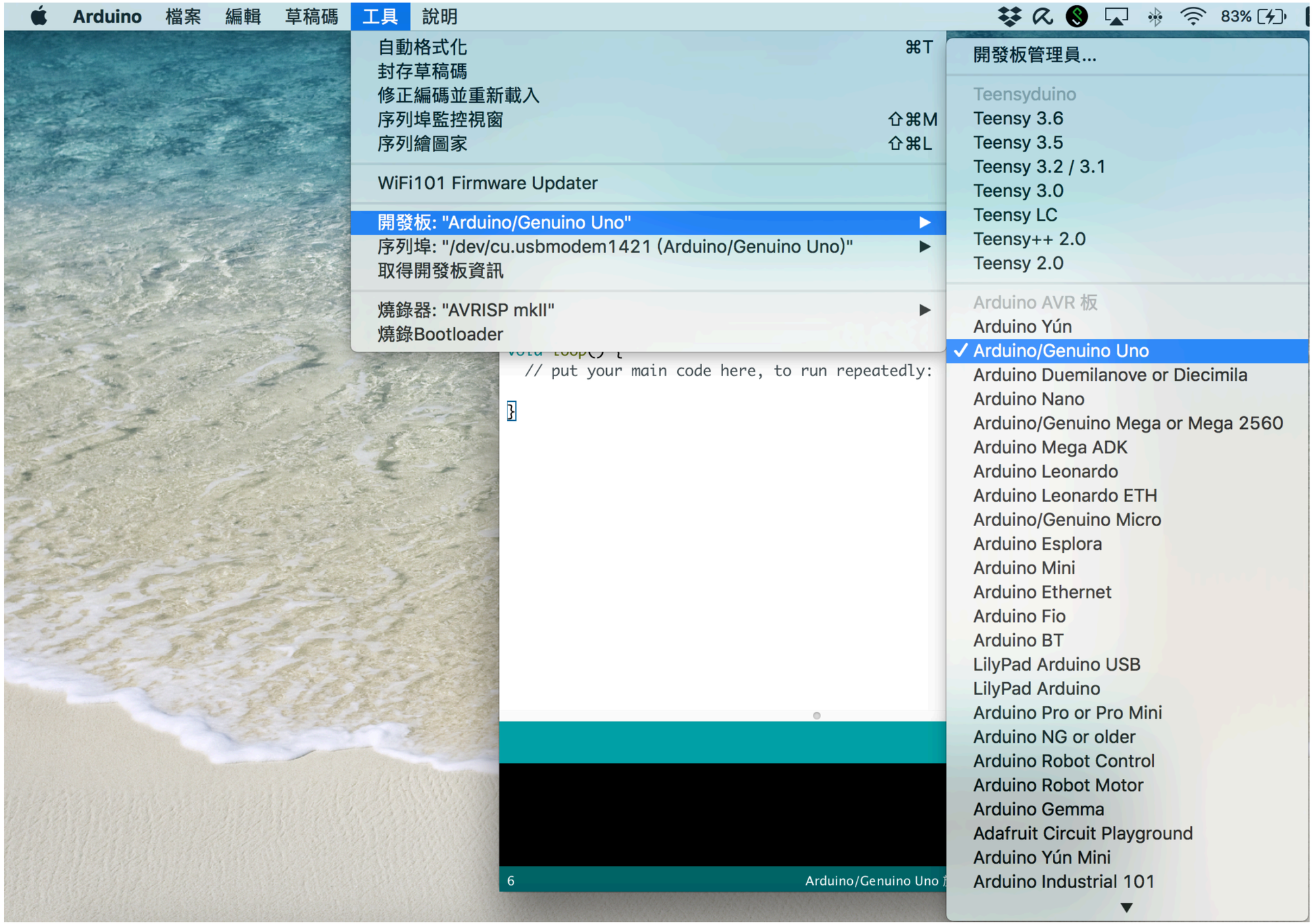
Serial Monitor

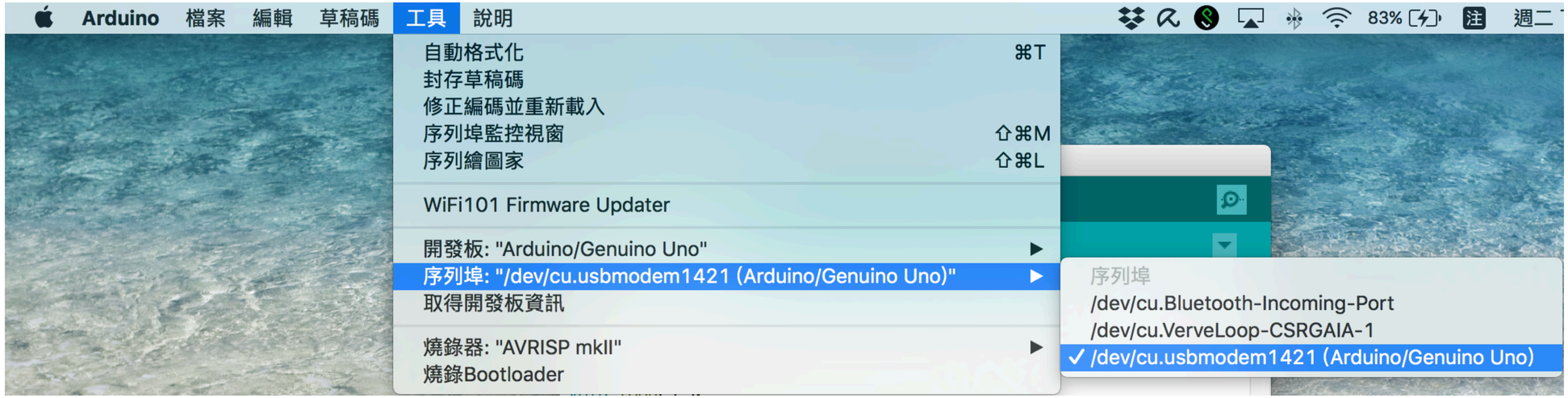


Only for initialization

Repeat every time

Arduino Board & Port


Arduino
檔案
編輯
草稿碼
工具
說明
83%
自動格式化 ⌘T
封存草稿碼
修正編碼並重新載入
序列埠監控視窗 ⇧⌘M
序列繪圖家 ⇧⌘L
WiFi101 Firmware Updater
開發板: "Arduino/Genuino Uno"
序列埠: "/dev/cu.usbmodem1421 (Arduino/Genuino Uno)"
取得開發板資訊
燒錄器: "AVRISP mkII"
燒錄Bootloader
// put your main code here, to run repeatedly:
}
6
Arduino/Genuino Uno
開發板管理員...
Teensyduino
Teensy 3.6
Teensy 3.5
Teensy 3.2 / 3.1
Teensy 3.0
Teensy LC
Teensy++ 2.0
Teensy 2.0
Arduino AVR 板
Arduino Yún
✓ Arduino/Genuino Uno
Arduino Duemilanove or Diecimila
Arduino Nano
Arduino/Genuino Mega or Mega 2560
Arduino Mega ADK
Arduino Leonardo
Arduino Leonardo ETH
Arduino/Genuino Micro
Arduino Esplora
Arduino Mini
Arduino Ethernet
Arduino Fio
Arduino BT
LilyPad Arduino USB
LilyPad Arduino
Arduino Pro or Pro Mini
Arduino NG or older
Arduino Robot Control
Arduino Robot Motor
Arduino Gemma
Adafruit Circuit Playground
Arduino Yún Mini
Arduino Industrial 101


Arduino
檔案
編輯
草稿碼
工具
說明
83%
週二
自動格式化 ⌘T
封存草稿碼
修正編碼並重新載入
序列埠監控視窗 ⇧⌘M
序列繪圖家 ⇧⌘L
WiFi101 Firmware Updater
開發板: "Arduino/Genuino Uno"
序列埠: "/dev/cu.usbmodem1421 (Arduino/Genuino Uno)"
取得開發板資訊
燒錄器: "AVRISP mkII"
燒錄Bootloader
序列埠
/dev/cu.Bluetooth-Incoming-Port
/dev/cu.VerveLoop-CSRGAIA-1
/dev/cu.usbmodem1421 (Arduino/Genuino Uno)

Arduino - LED

LED | Arduino 1.8.1

LED

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(13, HIGH);
  delay(500);
  digitalWrite(13, LOW);
  delay(200);
}
```

上傳完畢。

```
草稿碼使用了 948 bytes (2%) 的程式儲存空間。上限為 32256 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 20
```

1 Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

https://bit.ly/2I4BPFo

Arduino - LED

LED | Arduino 1.8.1

LED

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(13, HIGH);
  delay(500);
  digitalWrite(13, LOW);
  delay(200);
}
```

Set Pin Mode
(Pin number, Input/Output)

上傳完畢。

```
草稿碼使用了 948 bytes (2%) 的程式儲存空間。上限為 32256 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 20
```

1 Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

https://bit.ly/2I4BPFo

Arduino - LED

LED | Arduino 1.8.1

LED

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(13, HIGH);
  delay(500);
  digitalWrite(13, LOW);
  delay(200);
}
```

Set Pin Mode
(Pin number, Input/Output)

digitalWrite(pin, value)
https://bit.ly/2lhUc9O

上傳完畢。

```
草稿碼使用了 948 bytes (2%) 的程式儲存空間。上限為 32256 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 20
```

1 Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

https://bit.ly/2l4BPFo

Arduino - LED

LED | Arduino 1.8.1

LED

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(13, HIGH);
  delay(500);
  digitalWrite(13, LOW);
  delay(200);
}
```

Set Pin Mode
(Pin number, Input/Output)

digitalWrite(pin, value)
https://bit.ly/2lhUc9O

Delay in ms

上傳完畢。

```
草稿碼使用了 948 bytes (2%) 的程式儲存空間。上限為 32256 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 20
```

1 Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

https://bit.ly/2l4BPFo

Arduino - LED

Comment

Delay in ms

LED | Arduino 1.8.1

LED

```
void setup() {
  // put your setup code here, to run once:
  pinMode(13, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(13, HIGH);
  delay(500);
  digitalWrite(13, LOW);
  delay(200);
}
```

Set Pin Mode
(Pin number, Input/Output)

digitalWrite(pin, value)
https://bit.ly/2y1d7YM

上傳完畢。

草稿碼使用了 948 bytes (2%) 的程式儲存空間。上限為 32256 bytes。
全域變數使用了 9 bytes (0%) 的動態記憶體，剩餘 2039 bytes 給區域變數。上限為 20

1 Arduino/Genuino Uno 於 /dev/cu.usbmodem1421

https://bit.ly/2l4BPFo

SOS on LED

SOS（摩斯電碼：···---···）

1000ms and repeat

https://bit.ly/2l94h9a

[https://zh.wikipedia.org/wiki/SOS]

SOS on LED

SOS（摩斯電碼：···---···）

1000ms and repeat

Show on computer

S...

O---

S...

SOS on LED

LED-SOS

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:
  Serial.println("SOS");
  digitalWrite(ledPin, HIGH);   //S:...
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(100);
  digitalWrite(ledPin, HIGH);
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(100);
  digitalWrite(ledPin, HIGH);
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(200);
```

Set baud rate
(data rate in bits per second)

SOS on LED

LED-SOS

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:
  Serial.println("SOS");
  digitalWrite(ledPin, HIGH);   //S:...
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(100);
  digitalWrite(ledPin, HIGH);
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(100);
  digitalWrite(ledPin, HIGH);
  delay(100);
  digitalWrite(ledPin, LOW);
  delay(200);
```

Set baud rate
(data rate in bits per second)

Print ASCII text

SOS on LED

SOS（摩斯電碼：···---···）

1000ms and repeat

Show on computer

S...

O---

S...

https://bit.ly/2loTfMO

Arduino I/O

Arduino board

Arduino I/O

Computer Serial Monitor

Arduino board

Digit/Analog pins

Arduino I/O

Arduino I/O

Computer
Serial Monitor
Arduino
board
digitalWrite()
Digit/Analog
pins

Arduino I/O

Computer
Serial Monitor
Arduino
board
digitalWrite()
digitalRead()
Digit/Analog
pins

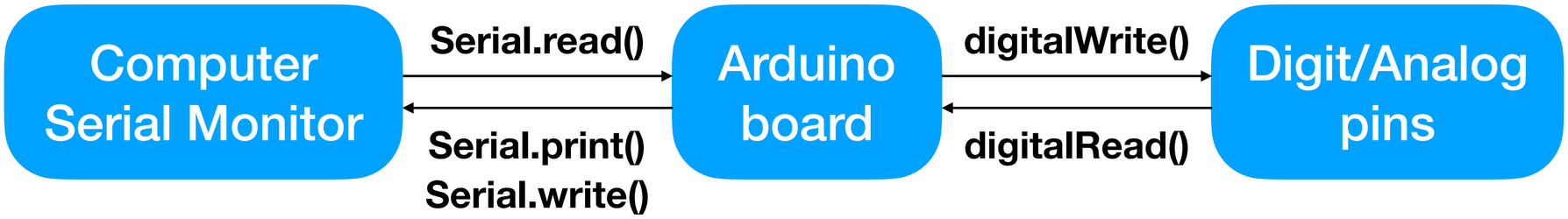
Arduino I/O

Computer
Serial Monitor
Serial.print()
Arduino
board
digitalWrite()
digitalRead()
Digit/Analog
pins

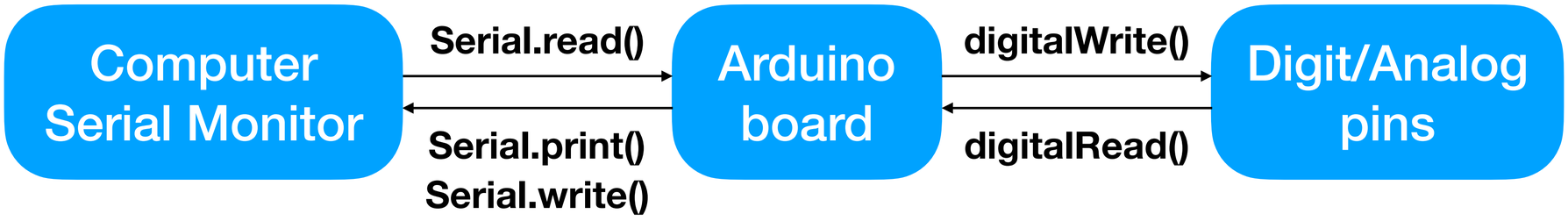
Arduino I/O

Computer Serial Monitor
Serial.read()
Serial.print()
Arduino board
digitalWrite()
digitalRead()
Digit/Analog pins

Arduino I/O

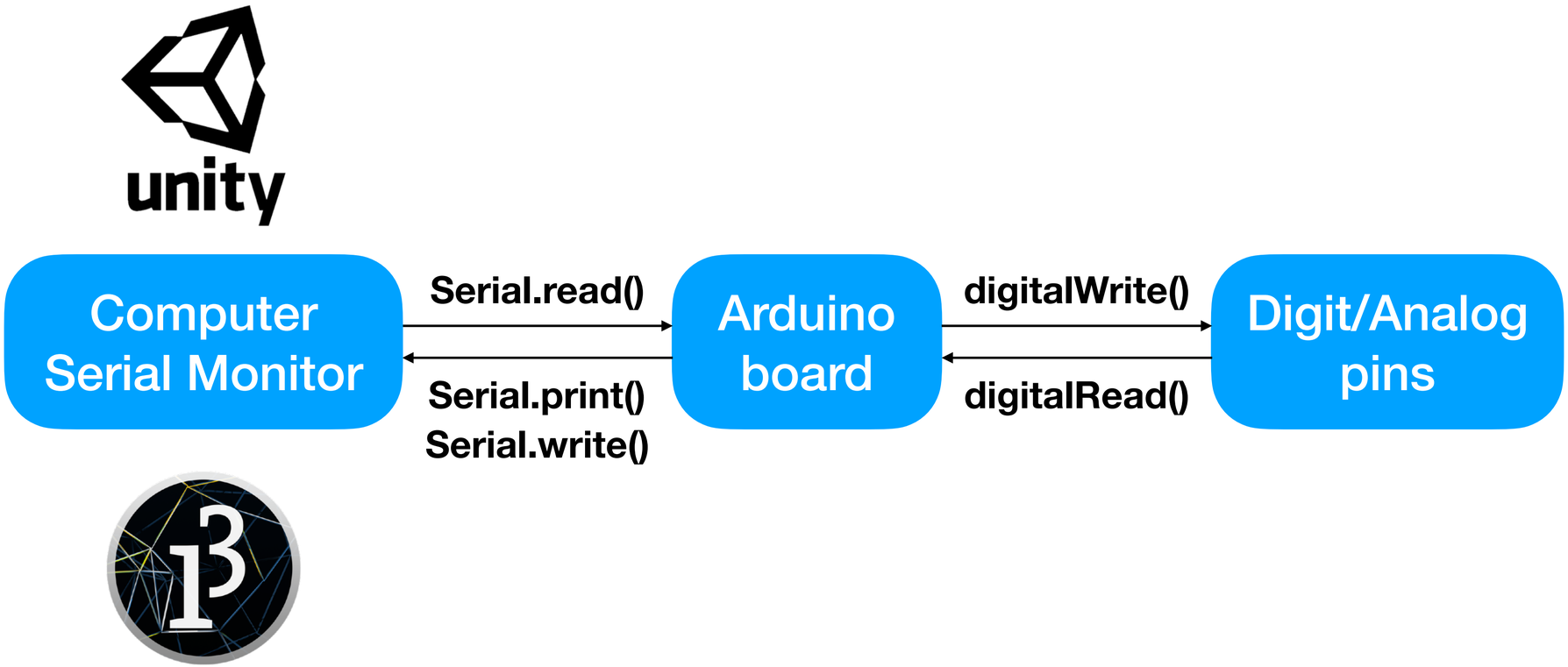

Computer
Serial Monitor
Serial.read()
Serial.print()
Serial.write()
Arduino
board
digitalWrite()
digitalRead()
Digit/Analog
pins

Arduino I/O

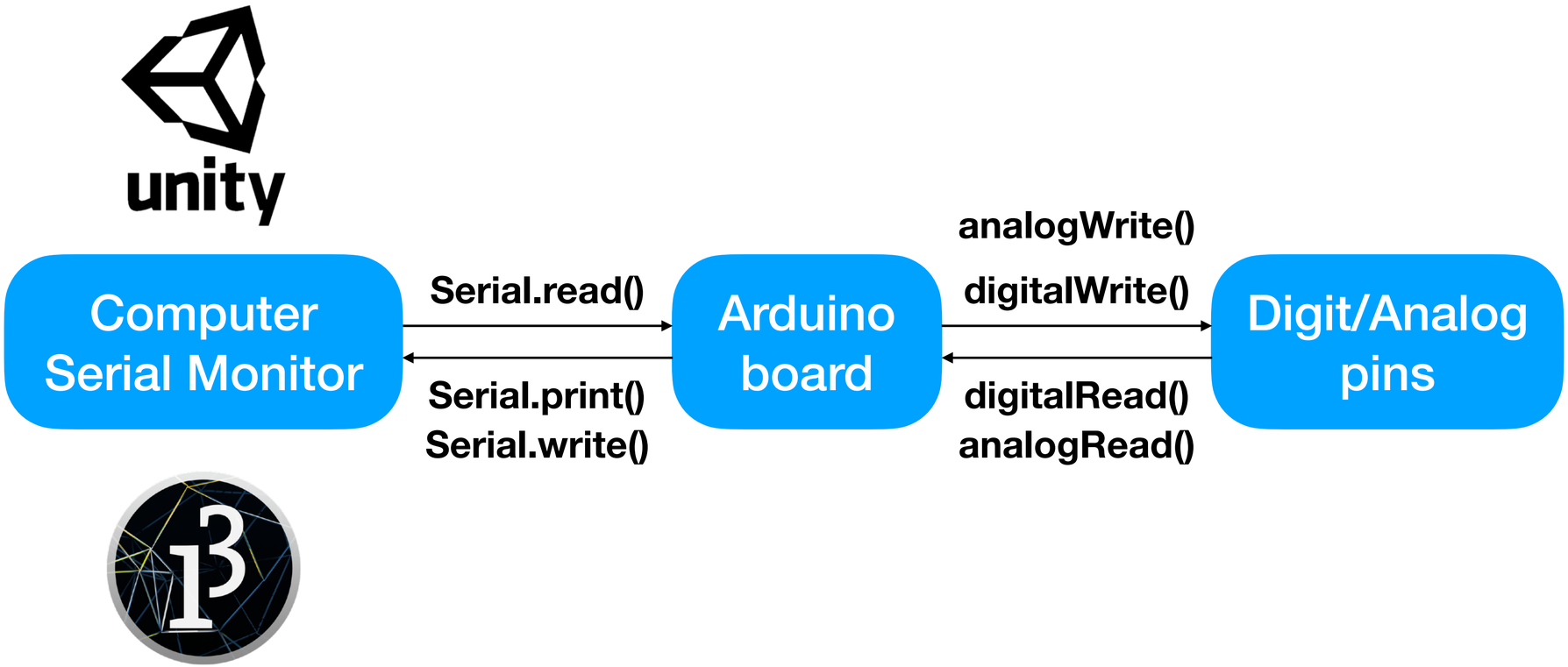


Serial.print(): ASCII text
Serial.write(): bytes

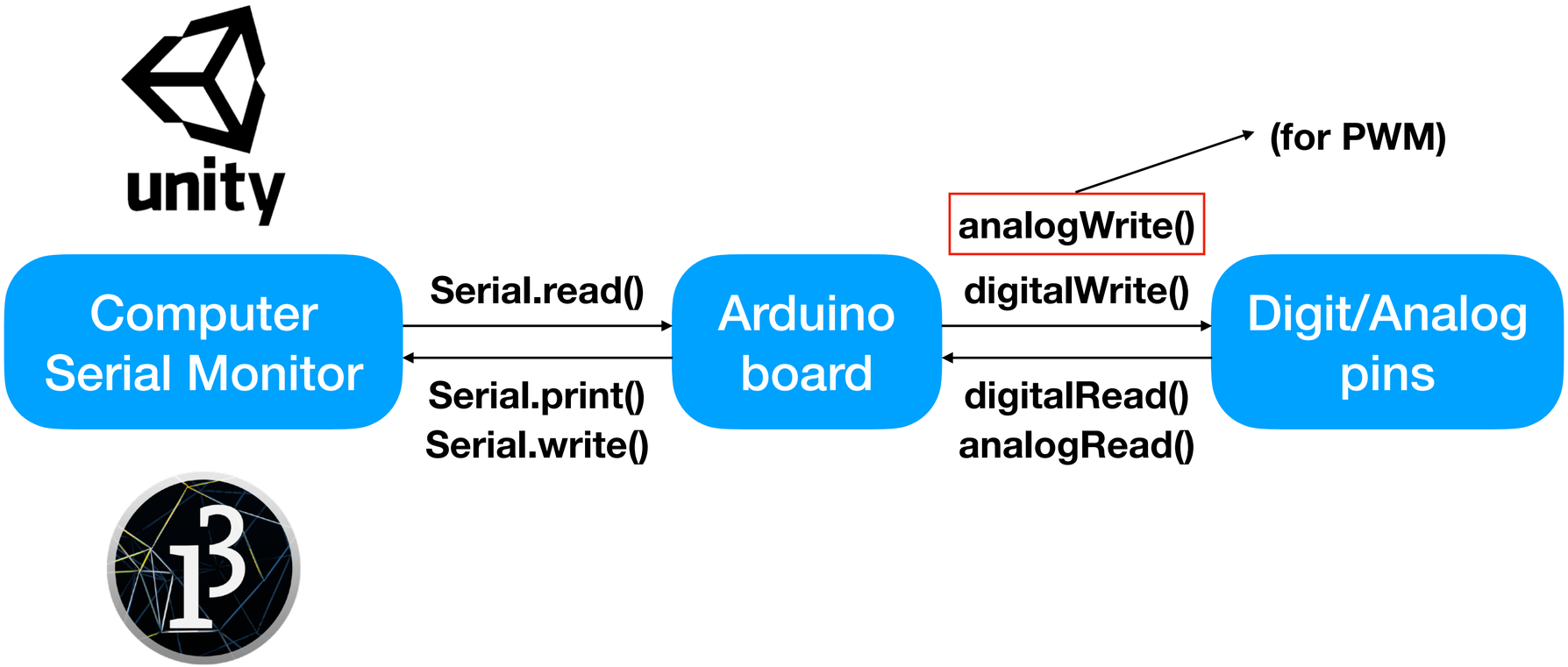
Arduino I/O


unity
Computer Serial Monitor
Serial.read()
Serial.print()
Serial.write()
Arduino board
digitalWrite()
digitalRead()
Digit/Analog pins

Arduino I/O


unity
Computer Serial Monitor
Serial.read()
Serial.print()
Serial.write()
Arduino board
analogWrite()
digitalWrite()
digitalRead()
analogRead()
Digit/Analog pins

Arduino I/O


unity
Computer Serial Monitor
Serial.read()
Serial.print()
Serial.write()
Arduino board
analogWrite()
(for PWM)
digitalWrite()
digitalRead()
analogRead()
Digit/Analog pins

Arduino I/O

Computer Serial Monitor
Serial.read()
Serial.print()
Arduino board
digitalWrite()
digitalRead()
Digit/Analog pins

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();       //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();       //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Bytes/chars in the buffer (64 byte)

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();        //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Read from the buffer

Bytes/chars in the buffer (64 byte)

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();        //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Read from the buffer

Bytes/chars in the buffer (64 byte)

Enter ‘s’: start SOS
Enter ‘e’: end SOS

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();        //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Read from the buffer

Bytes/chars in the buffer (64 byte)

Enter ‘s’: start SOS
Enter ‘e’: end SOS

https://bit.ly/2mYnMBT

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    char inst = Serial.read();        //instruction from monitor
    Serial.print("Instruction: ");
    Serial.println(inst);
  }
```

Read from the buffer

Bytes/chars in the buffer (64 byte)

Enter a number for SOS execution times, e.g., 12 times

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    int num = Serial.parseInt();
    Serial.print("Instruction: ");
    Serial.println(num);
  }
```

Read from the buffer

https://bit.ly/2lsz8gK

Enter a number for SOS execution times

Arduino - Serial.read()

```
int ledPin = 13;

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:

  if (Serial.available() > 0) {
    int num = Serial.parseInt();
    Serial.print("Instruction: ");
    Serial.println(num);
  }
```

Read from the buffer

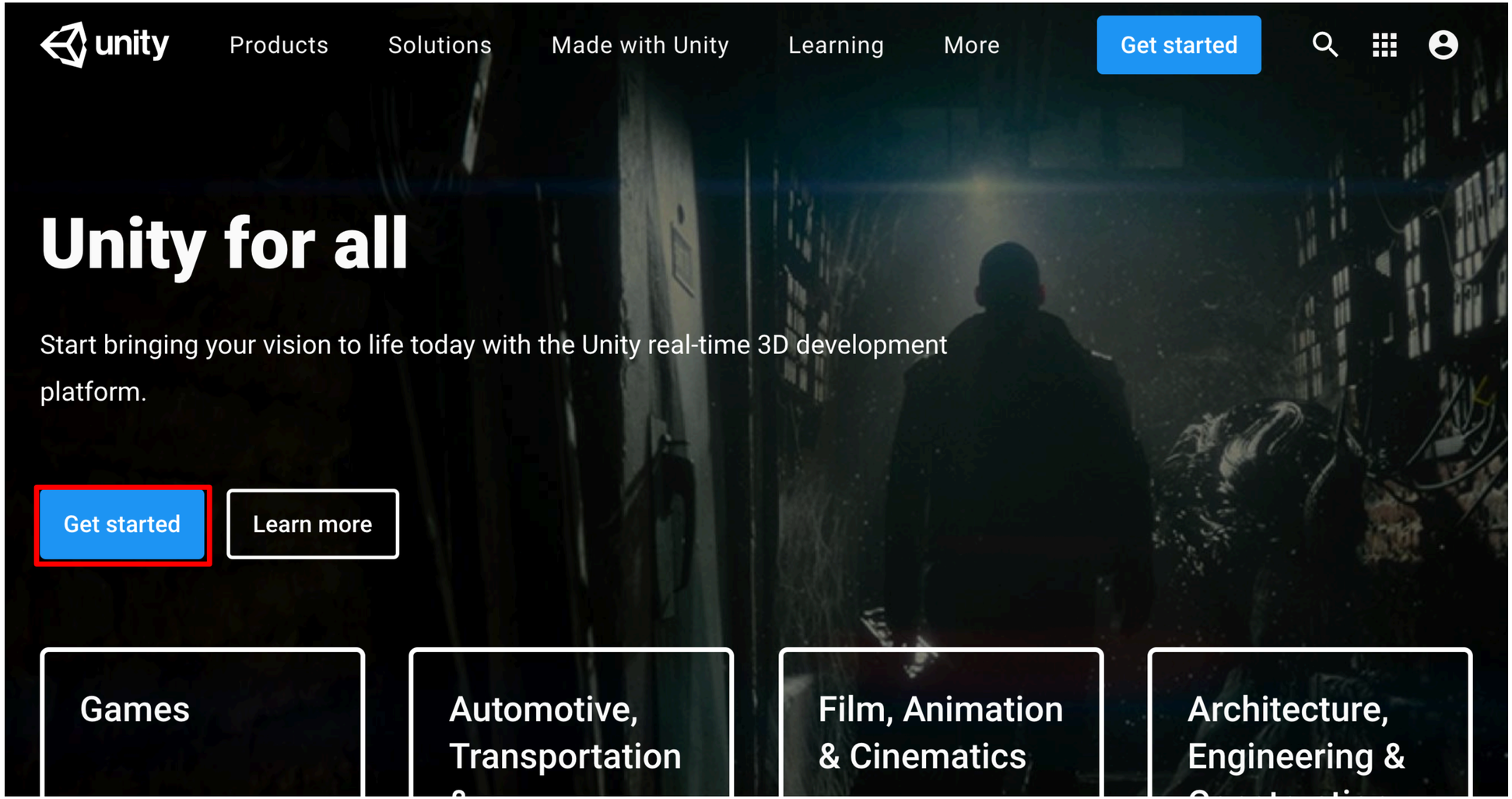
Enter a number for SOS execution times

https://bit.ly/2lq0PXy

Arduino - LED Control

Enter instructions to trigger different LED patterns.

Unity Installation


unity
Products
Solutions
Made with Unity
Learning
More
Get started
Unity for all
Start bringing your vision to life today with the Unity real-time 3D development platform.
Get started
Learn more
Games
Automotive, Transportation
Film, Animation & Cinematics
Architecture, Engineering &

unity Store

RT

Plans and pricing

We offer a range of plans for all industries and levels of expertise.
All plans are royalty-free.

Individual | Business

Plus

More functionality and resources to

Pro MOST POPULAR

A complete solution for professionals

Enterprise

Custom solutions at scale,

Individual | Business

Student

Learn the tools and workflows professionals use on the job

Free

Sign up

Eligibility:
Students enrolled in an accredited educational institution of legal age to consent to the collection and processing of their personal information, e.g., age 13 in the US, 16 in the EU. Must join the GitHub Student Developer Pack to be verified.

Personal

Start creating with the free version of Unity

Free

Get started | Learn more

Eligibility:
Revenue or funding less than $100K in the last 12 months

Learn Premium

Master Unity with expert-led live sessions and on-demand learning

Try for free | Learn more

Included with Plus, Pro and Enterprise plans

Start Creating with Unity

First-time Users

Download Unity with this fun, **guided learning path** to create your first game today!

Start here

Returning Users

Already have experience creating with Unity and want to download the **standard Unity installer** instead?

Go here

Download Unity

Welcome! You're here because you want to download Unity, the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences.

Before you download choose the version of Unity that's right for you.

Choose your Unity + download | Download Unity Hub

Learn more about the new Unity Hub here.

Download Unity Beta

Get early access to our latest features, and help us improve quality by providing valuable feedback.

Download Beta

Already a customer?

System requirements

OS: Windows 7 SP1+, 8, 10, 64-bit versions only; Mac OS X 10.12+; Ubuntu 16.04, 18.04, and CentOS 7.

GPU: Graphics card with DX10 (shader model 4.0) capabilities.

Learn more

Resources

- LTS Releases
- Older versions of Unity
- Unity 2019.3 upgrade guide
- Patch releases
- Release notes
- Engine features
- Unity User Manual
- FAQ

We use cookies to ensure that we give you the best experience on our website. Visit our cookie policy page for more information.

Got it

Unity Hub 安裝

授權協議

在安裝 Unity Hub 之前，請檢閱授權條款。

檢閱協議的其餘部分，按 [PgDn] 往下捲動頁面。

Unity Terms of Service

Last updated: May 24, 2018

Unity Technologies ApS ("**Unity**", "**our**" or "**we**") provides game-development and related software (the "**Software**"), development-related services (like Unity Teams ("**Developer Services**")), and various Unity communities (like Unity Answers and Unity Connect ("**Communities**")), provided through or in connection with our website accessible at unity3d.com or unity.com

如果接受協議的條款，按 [我同意(A)] 繼續安裝。必須要接受協議才能安裝 Unity Hub。

Unity Hub 2.3.0

我同意(A) 取消

Unity Hub 安裝

選取安裝位置

選取 Unity Hub 要安裝的資料夾。

安裝程式會將 Unity Hub 安裝在以下資料夾。要安裝到不同的資料夾，按「瀏覽(B)...」並選擇其他資料夾。 按「安裝(I)」開始安裝。

目標資料夾

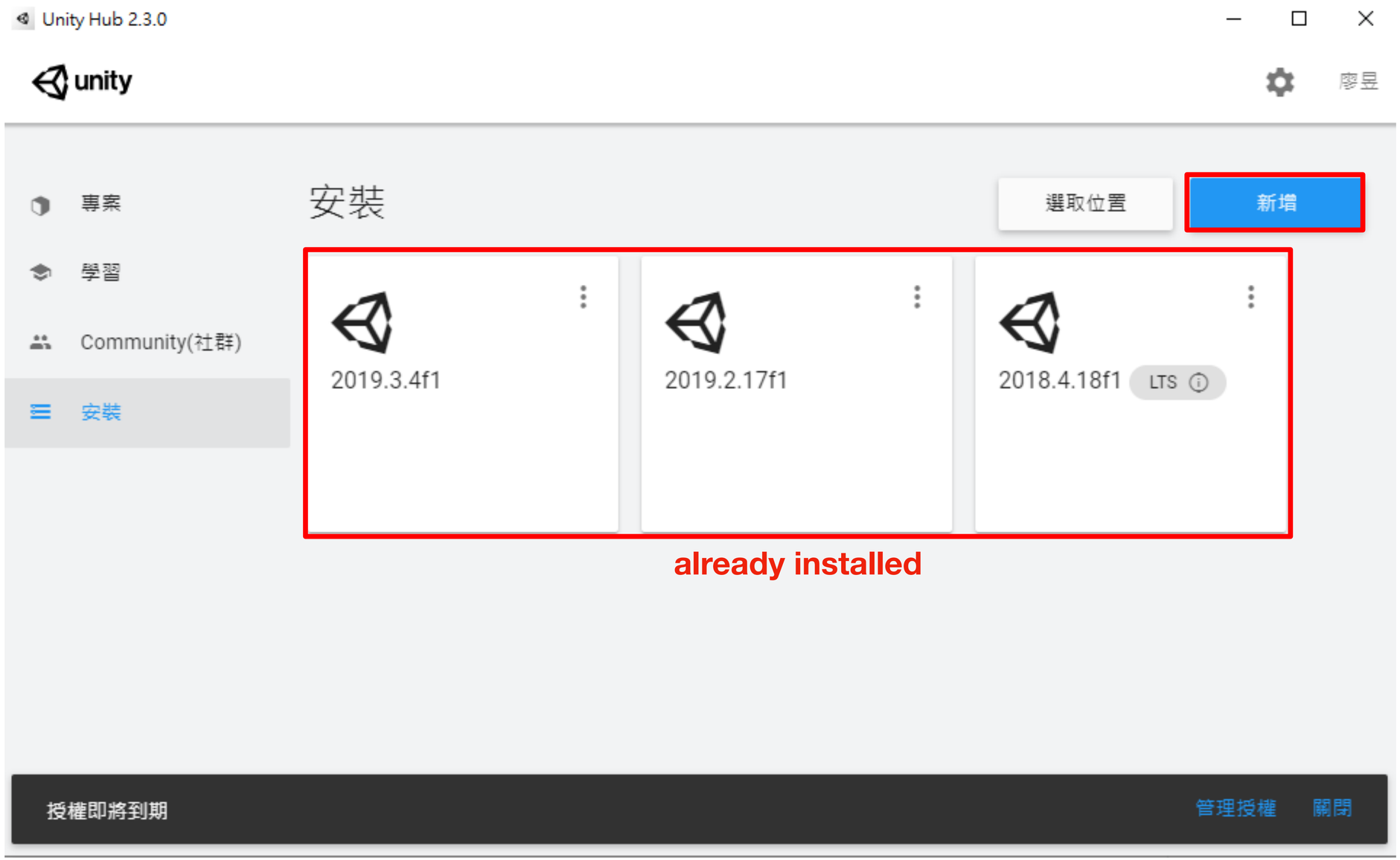
C:\Program Files\Unity Hub

瀏覽(R)...

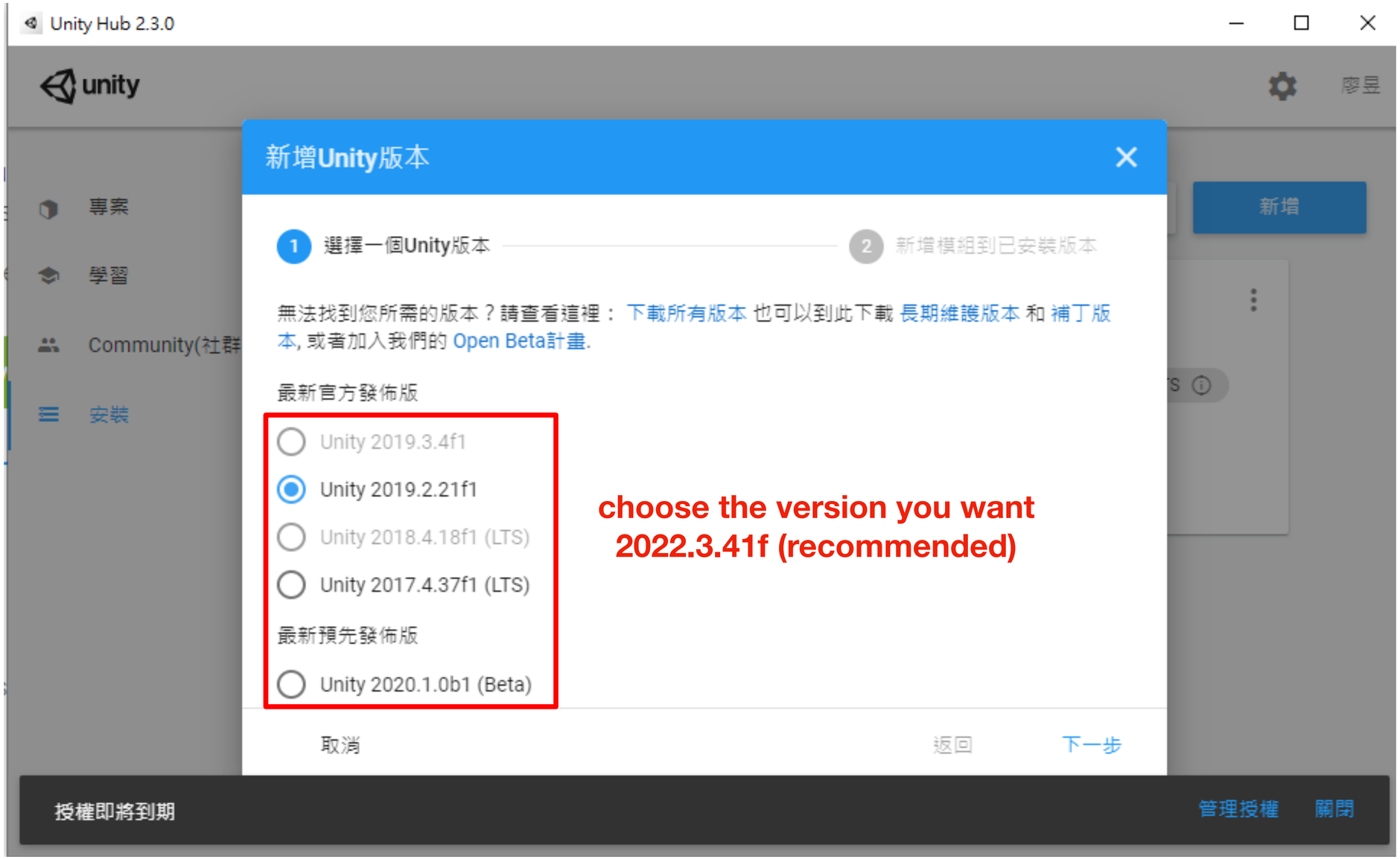
Unity Hub 2.3.0

< 上一步(B) | 安裝(I) | 取消

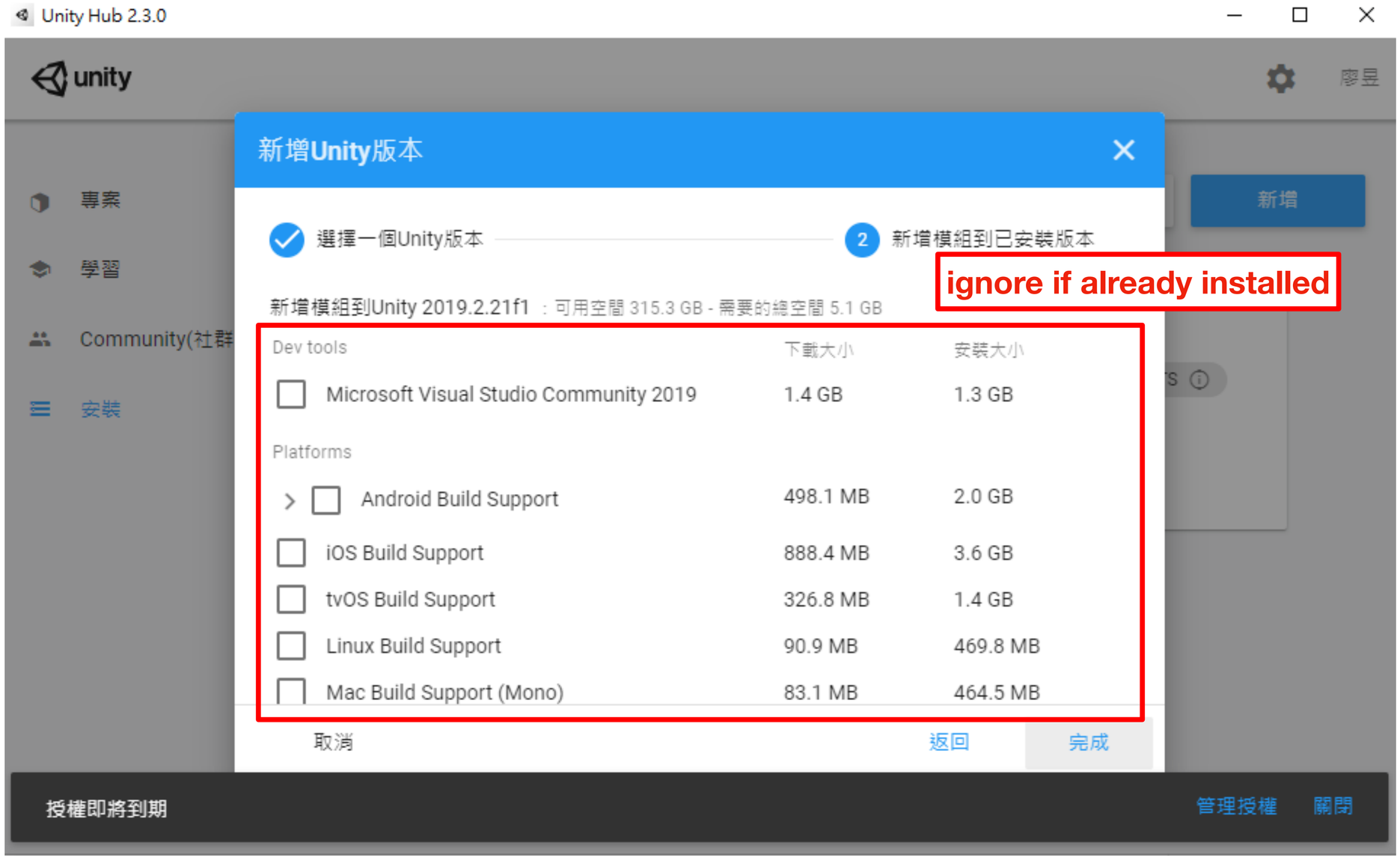
Unity Hub 安裝
即將完成安裝 Unity Hub
已在電腦安裝 Unity Hub 。
按 [完成(F)] 關閉安裝程式。
執行 Unity Hub(R)
< 上一步(B)
完成(F)
取消



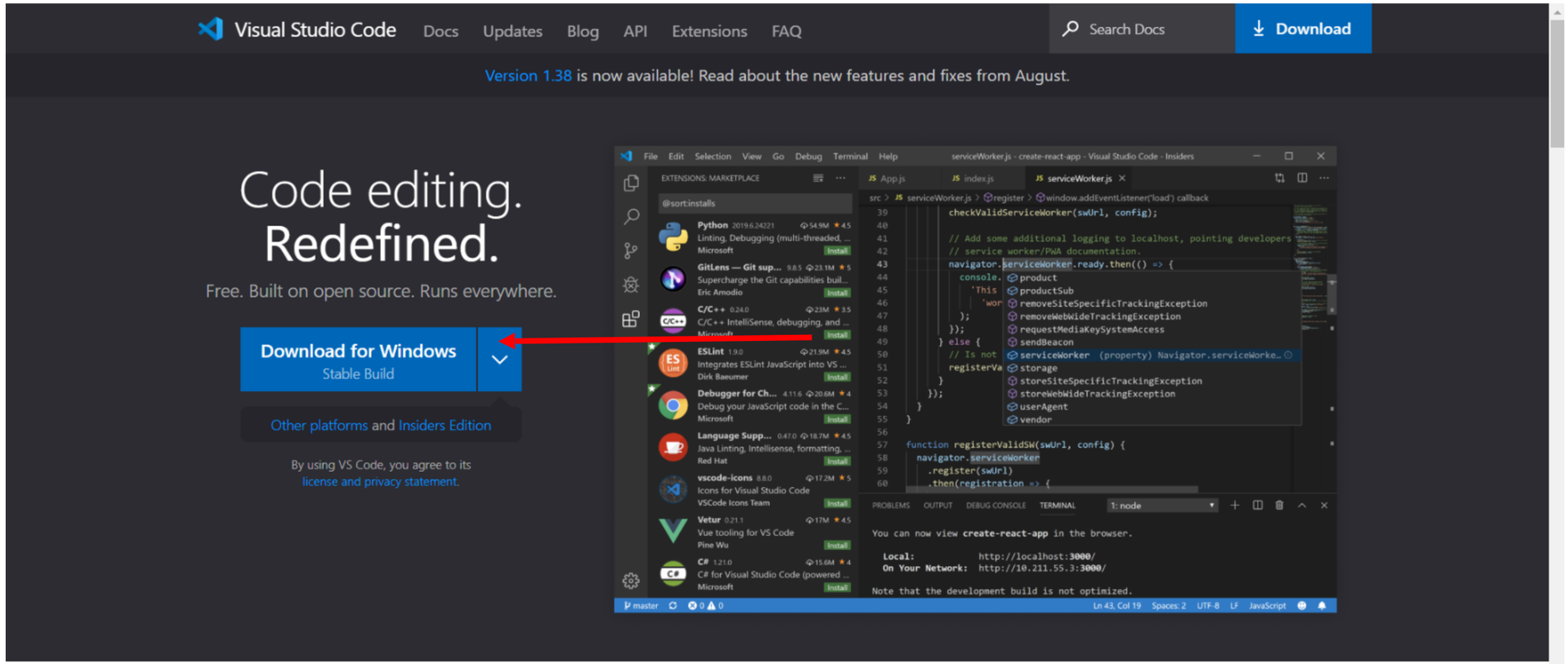
already installed



choose the version you want
2022.3.41f (recommended)


Unity Hub 2.3.0
unity
專案
學習
Community(社群
安裝
新增
新增Unity版本
選擇一個Unity版本
2 新增模組到已安裝版本
ignore if already installed
新增模組到Unity 2019.2.21f1 ：可用空間 315.3 GB - 需要的總空間 5.1 GB
Dev tools
下載大小
安裝大小
Microsoft Visual Studio Community 2019 1.4 GB 1.3 GB
Platforms
Android Build Support 498.1 MB 2.0 GB
iOS Build Support 888.4 MB 3.6 GB
tvOS Build Support 326.8 MB 1.4 GB
Linux Build Support 90.9 MB 469.8 MB
Mac Build Support (Mono) 83.1 MB 464.5 MB
取消
返回
完成
授權即將到期
管理授權
關閉

Visual Studio Code Installation (optional)



Go to: https://code.visualstudio.com/ and download