

GDC

March 21-25, 2022
San Francisco, CA

Figuring Out How Things Work in an Unfamiliar Codebase

#GDC22



About Molly Jameson



Overview

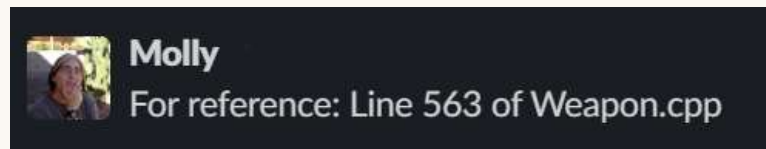
1. Assumptions and motivations
2. Tips on exploring codebases on your own
3. When and how to approach asking for help
4. Forming a big picture mental model

Assumptions

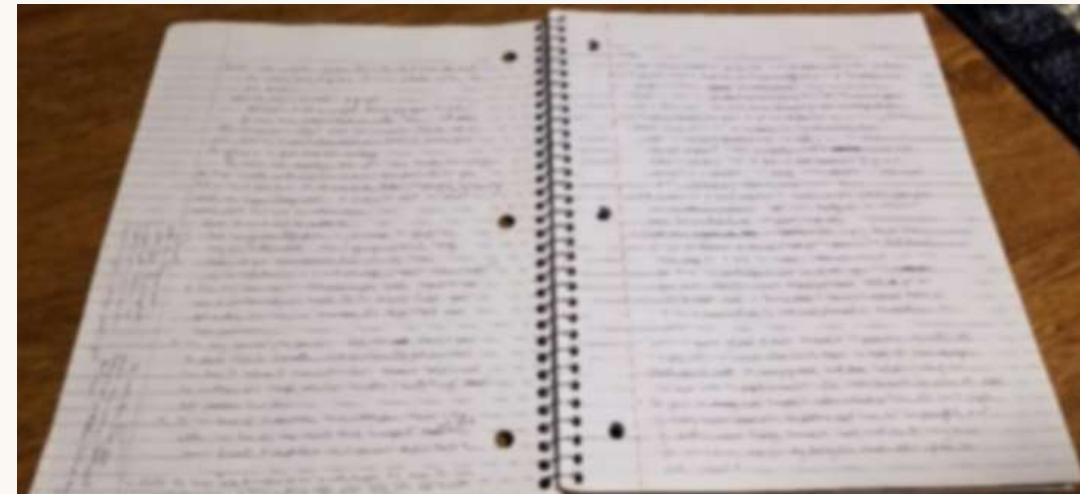
```
12 // · [TODO] · Write · docs ·  
13 // · ticket: · JIRA-002
```

Story time

What I got out of this interaction:



What the intern got out of this interaction:



Most important rule

- Don't be an arrogant a**hole
- When you see something “strange” in a new codebase, don't assume the code is just bad
- Do keep an open mind

Frequent but irrelevant advice

- “If you don’t know something, just ask.”

Exploring a Problem



Step 0

- Play with the editor and game
- Focus on a small task first

Good first tasks

- Mostly additive
- Bite sized
- “Nice to haves”

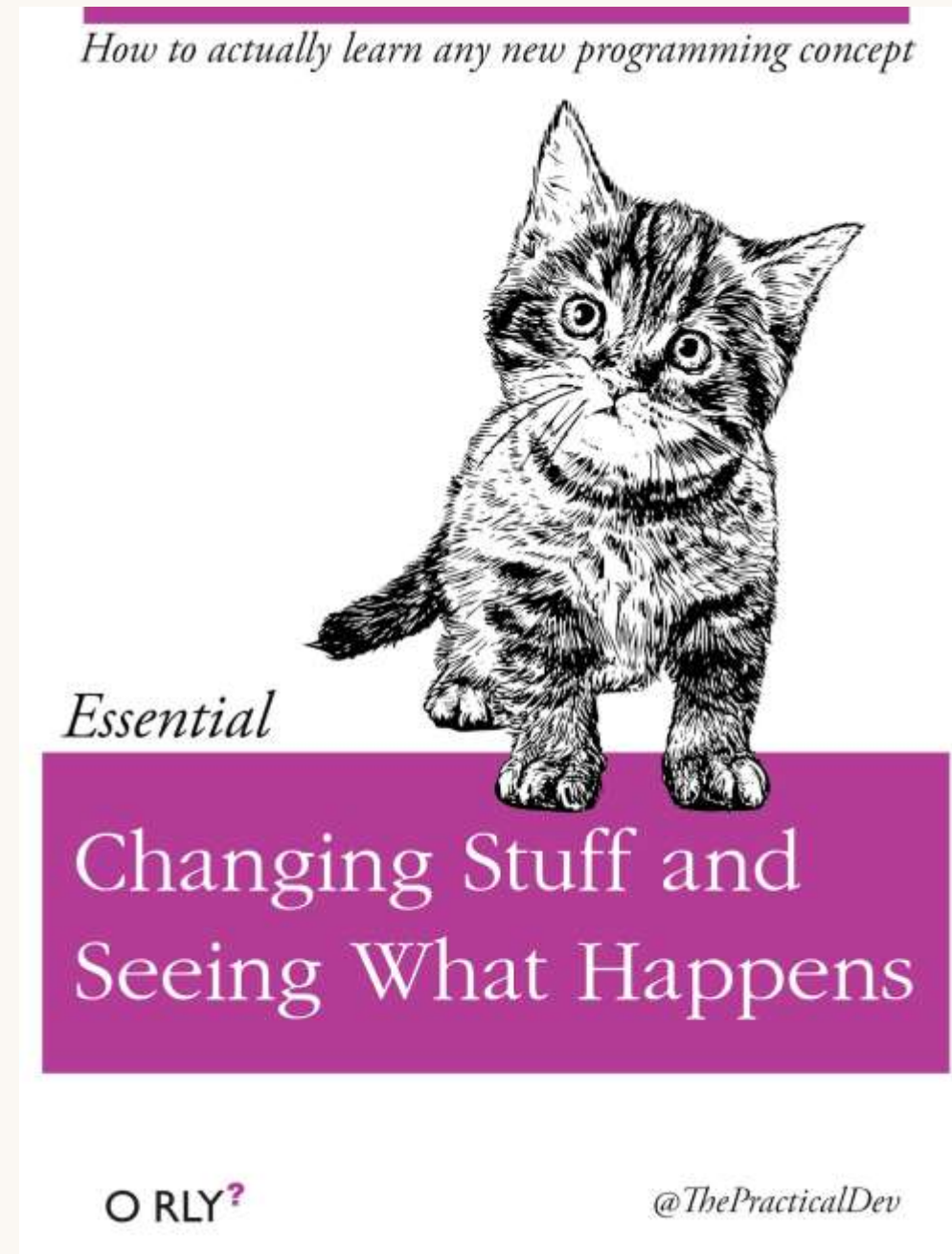


Example Starter Task



```
// TODO: This should be configurable per harvestable, not hardcoded.  
constexpr float RECHARGE_TIME_DAYS = 1.f;
```

Start with Data



Look at Logs

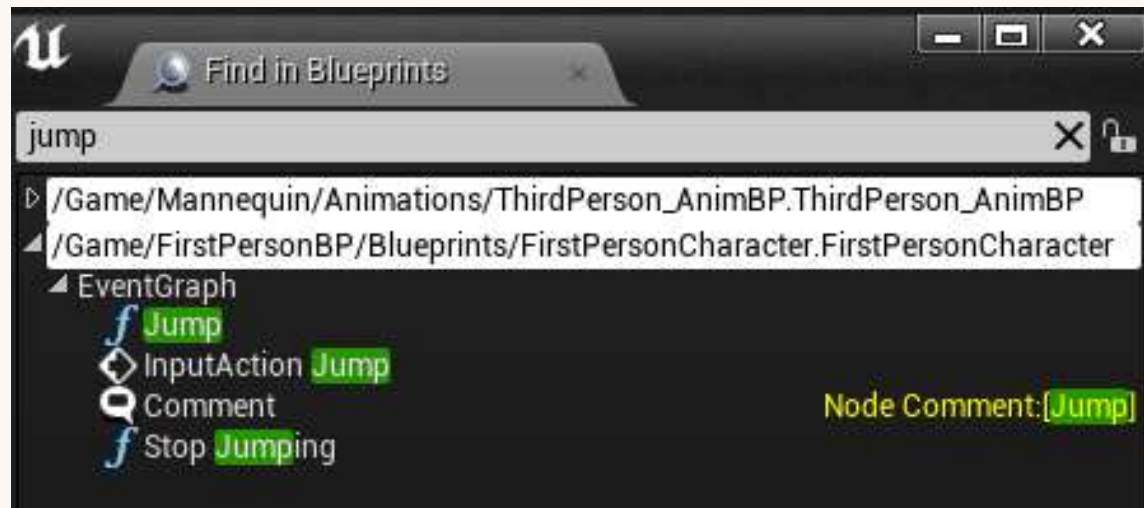
- Scan for relevant text
- Modify verbosity and channels

Use Grep/Find in Files

```
ubuntu@MollyPC:~$ grep -Hn HarvestableAmount -r ./*.cfg  
./ContrivedGDCExample.cfg:4:HarvestableAmount: 2
```

```
34 void Harvestable::Load(const HarvestableConfig* Config) {  
35     ResourceQuantity = Config->GetHarvestableAmount();
```

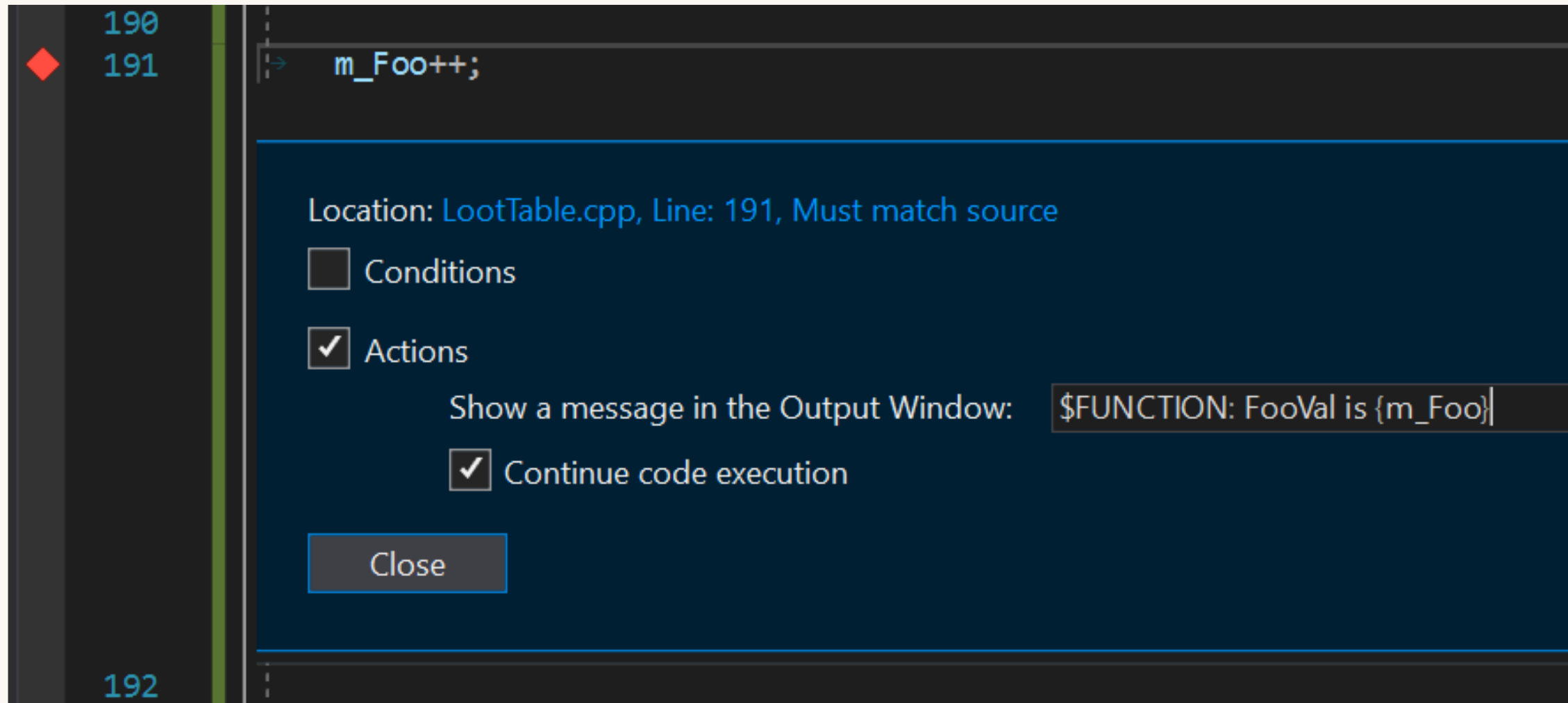
```
53 void Harvestable::OnInteract() {  
54     GiveLoot(name:ResourceID, amount:ResourceQuantity);
```



Set breakpoints and inspect

Call Stack			
	Name	Lang	
👉	Void GameOverState:FinishRun ()+0x1 at D:\SideProjects\UnityRunnerExample\UnityRunnerExample\A...	C#	
	Void GameOverState:Exit (AState)+0x14 at D:\SideProjects\UnityRunnerExample\UnityRunnerExample\...	C#	
	Void GameManager:SwitchState (String)+0x41 at D:\SideProjects\UnityRunnerExample\UnityRunnerEx...	C#	
	Void GameOverState:RunAgain ()+0x19 at D:\SideProjects\UnityRunnerExample\UnityRunnerExample\...	C#	
	Void UnityEngine.Events.InvokableCall:Invoke ()+0x16 at :-1	C#	
	Void UnityEngine.Events.UnityEvent:Invoke ()+0x23 at :-1	C#	
	Void UnityEngine.UI.Button:Press ()+0x2d at C:\Program Files\Unity\Hub\Editor\2019.4.17f1\Editor\Dat...	C#	
	Void UnityEngine.UI.Button:OnPointerClick (PointerEventData)+0x11 at C:\Program Files\Unity\Hub\E...	C#	
	Void UnityEngine.EventSystems.ExecuteEvents:Execute (IPointerClickHandler, BaseEventData)+0x8 at C...	C#	
	Boolean UnityEngine.EventSystems.ExecuteEvents:Execute (GameObject, BaseEventData, EventFunctio...	C#	
	Void UnityEngine.EventSystems.StandaloneInputModule:ReleaseMouse (PointerEventData, GameObjec...	C#	

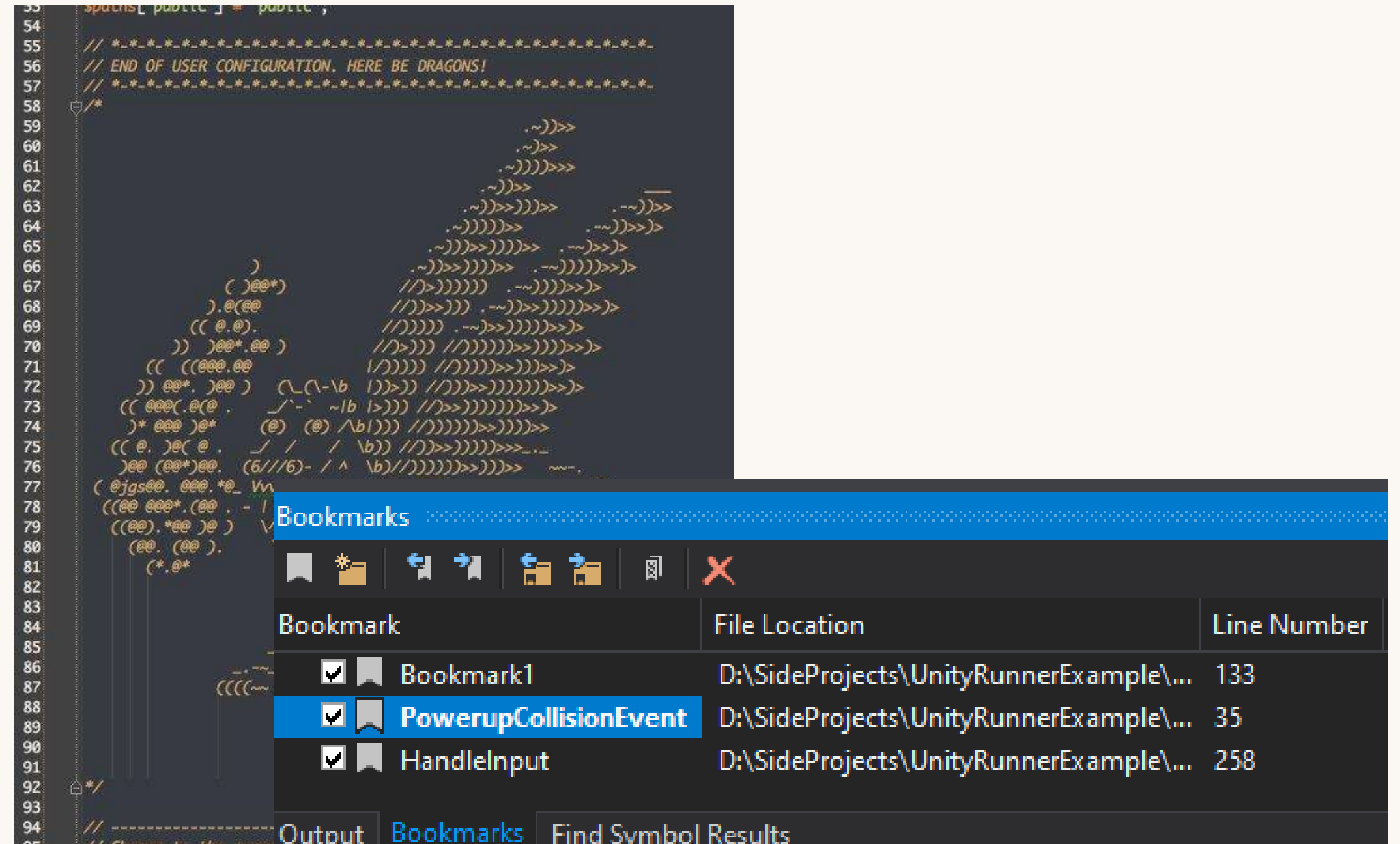
Tracepoints and Conditional Breakpoints



Thread pulling adventure



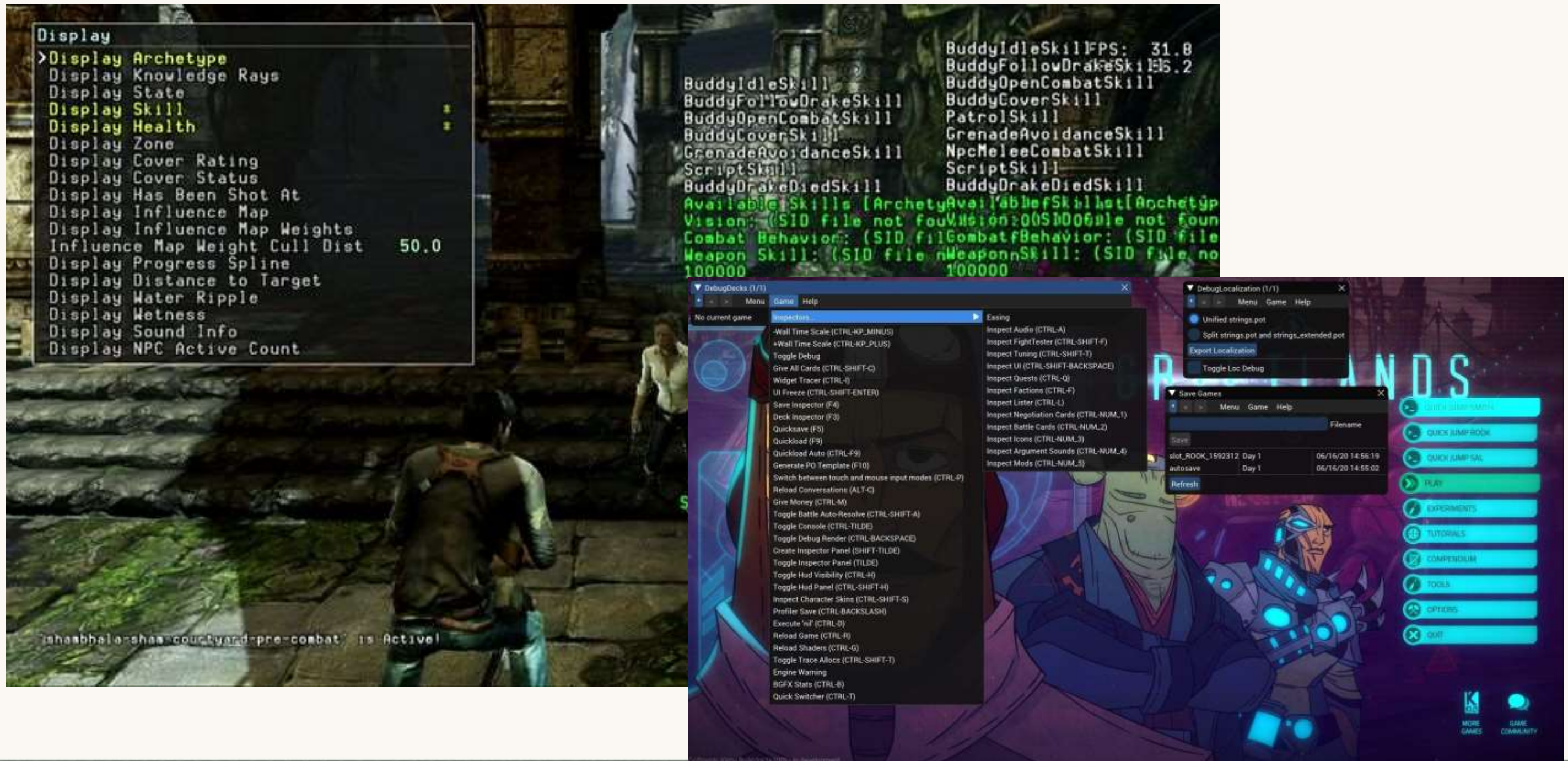
Back out of rabbit holes you've fallen into



Take notes as you step through



Explore the debug menu



Asking For Help



Goals for asking

1. Not take up too much time of the person you're going to ask
2. Not waste an excessive amount of your own time unnecessarily

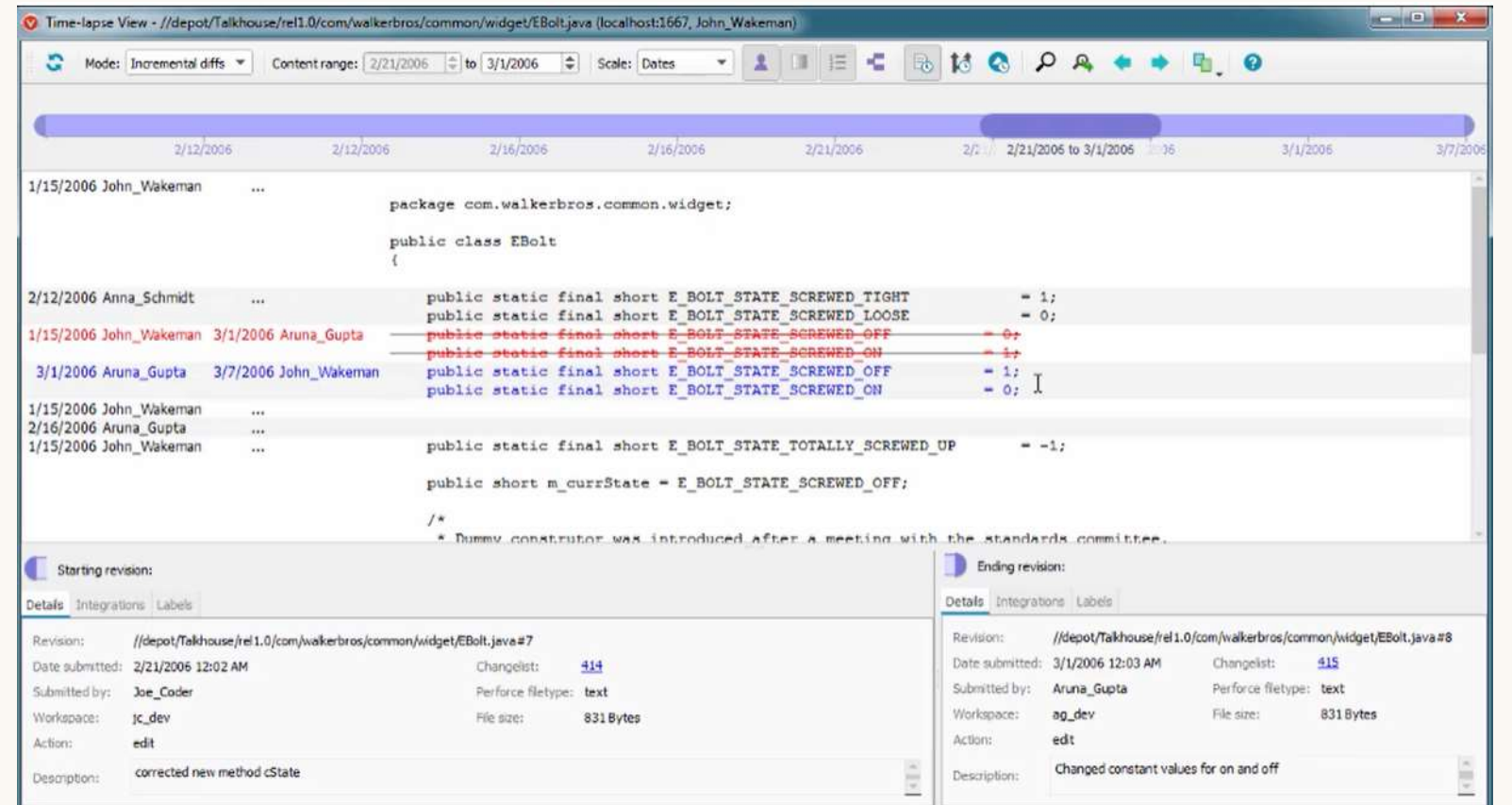
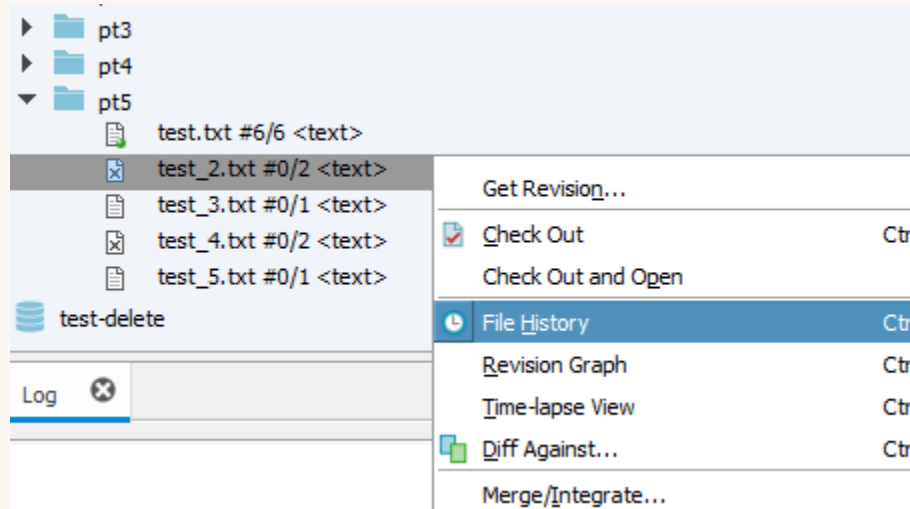


Before asking

1. Clearly state what you are trying to do
2. Establish a good list of what to you already tried
3. Timebox how long to spend before asking
4. Prefer questions that will give you the tools to debug problem yourself

Who to ask

Check source control “blame” for suspect files and lines



While asking



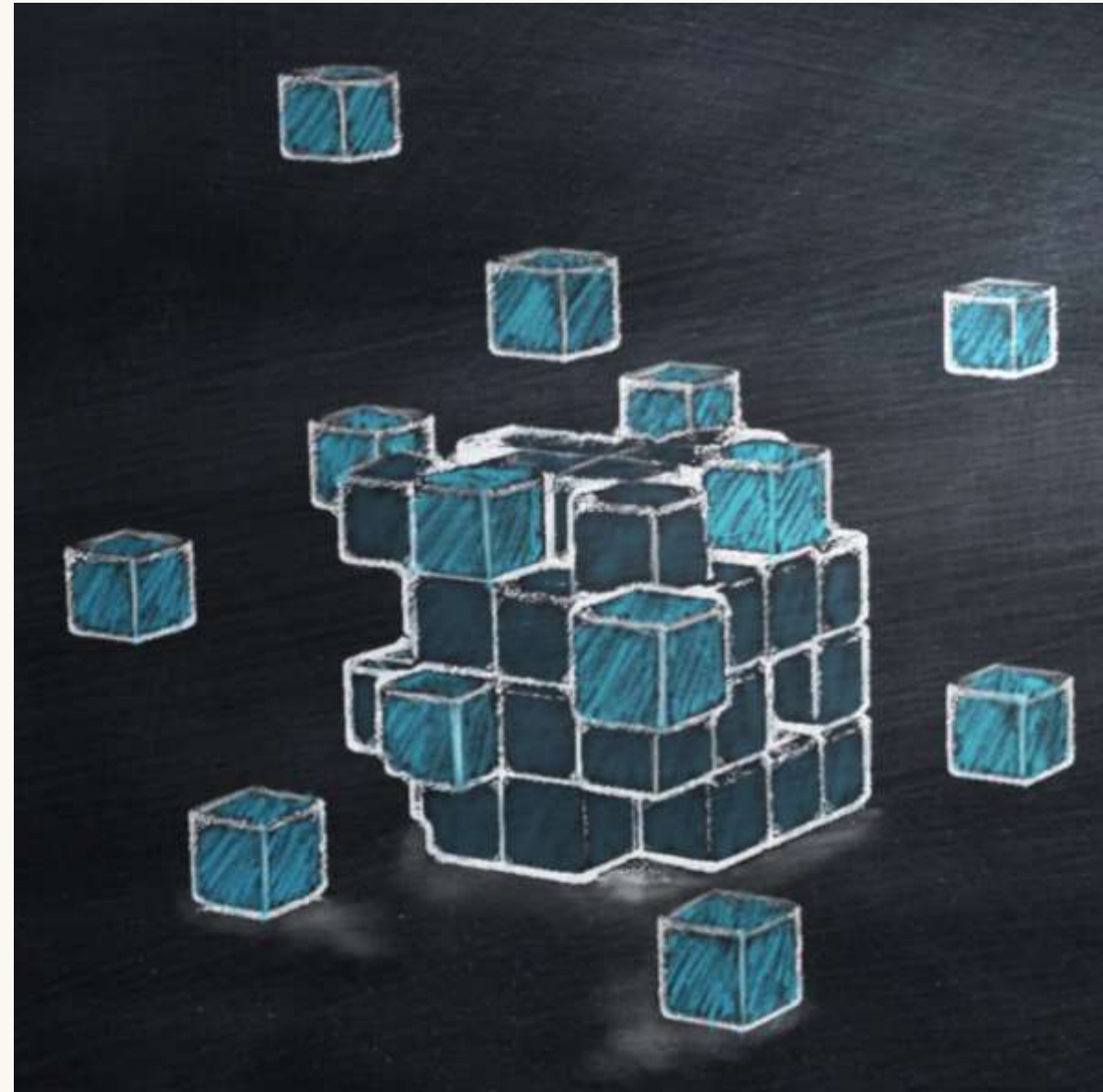
After asking

- Take notes on your recent TIL (Today, I learned...)
- “TIL... Animals are implemented by the class named FurnitureItem”
- “TIL... clicking F2 makes the avatar fly and testing way faster”

Forming A Big Mental Model



Modular Black Boxes



Code reviews

- Spy on other people's reviews
- Ask for code reviews

Connect familiar terms

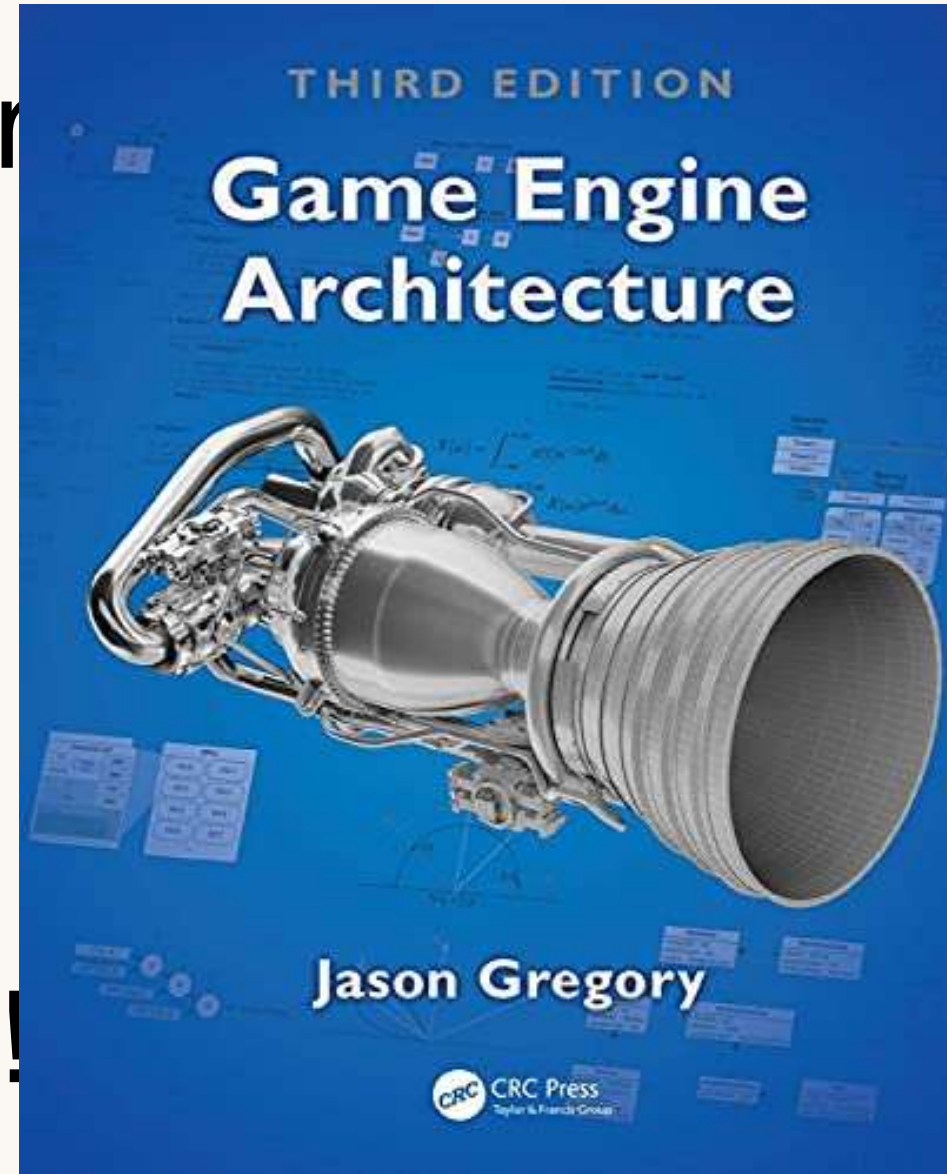
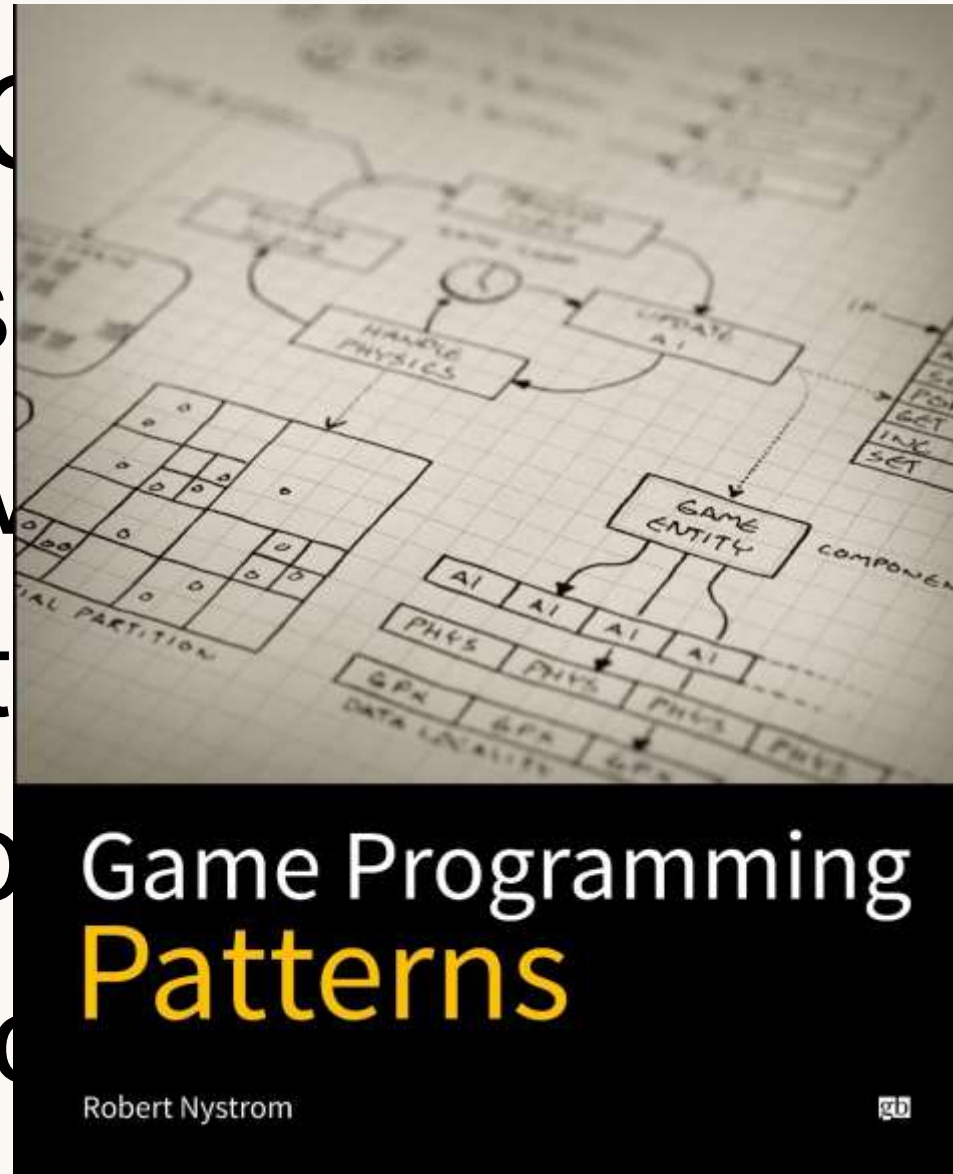
Quick Glossary

The following section contains common Unity terms on the left and their UE4 equivalents (or rough equivalent) on the right. UE4 keywords link directly to more in-depth information inside Unreal Engine's Online Documentation.

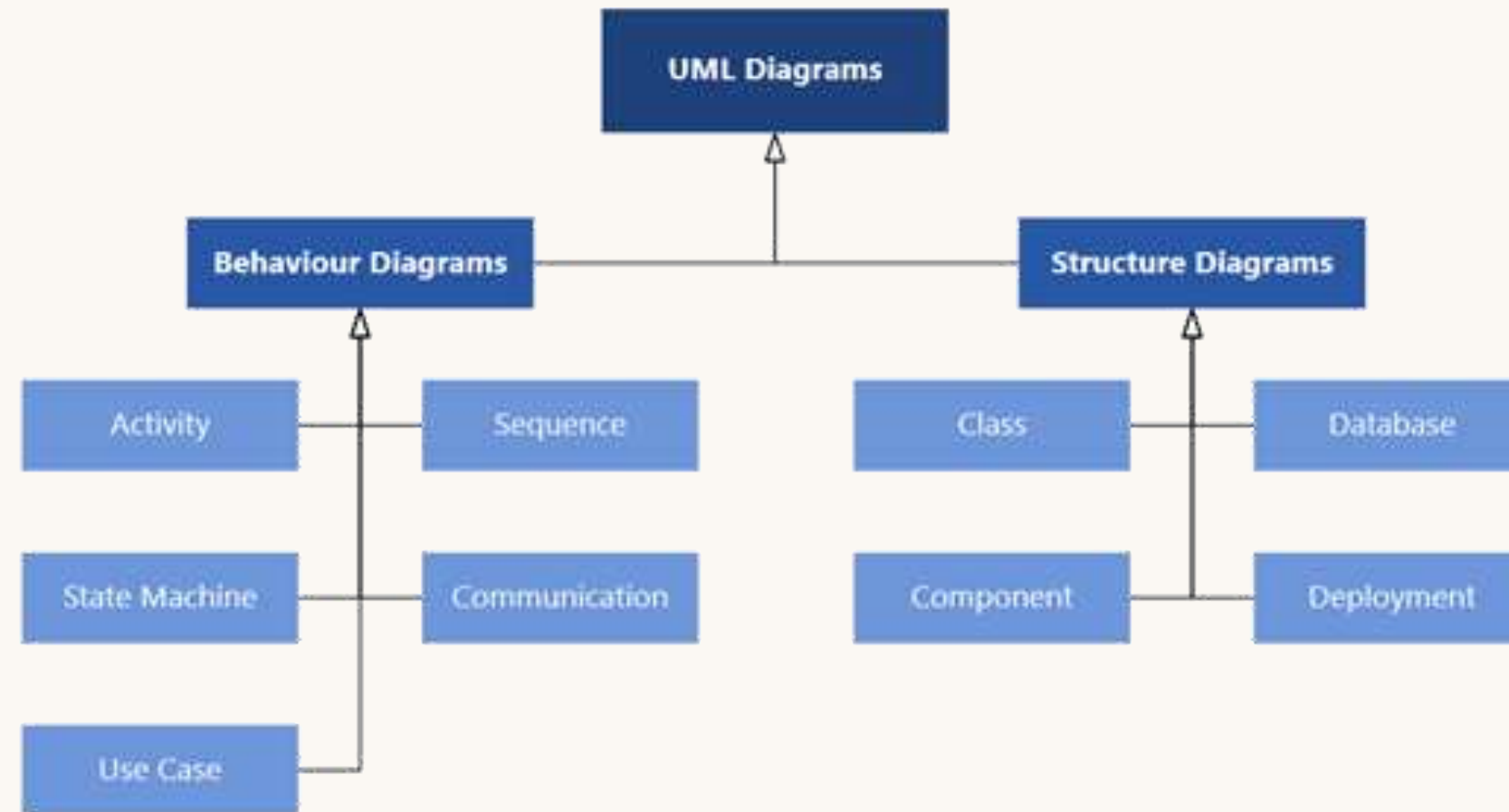
Category	Unity	UE4
Gameplay Types	Component	Component
	GameObject	Actor, Pawn
	Prefab	Blueprint Class
Editor UI	Hierarchy Panel	World Outliner
	Inspector	Details Panel
	Project Browser	Content Browser

Common Game Programming Paradigms

- Entity Component Paradigm
- Events
- State Machine
- Singleton
- Adaptor
- And More

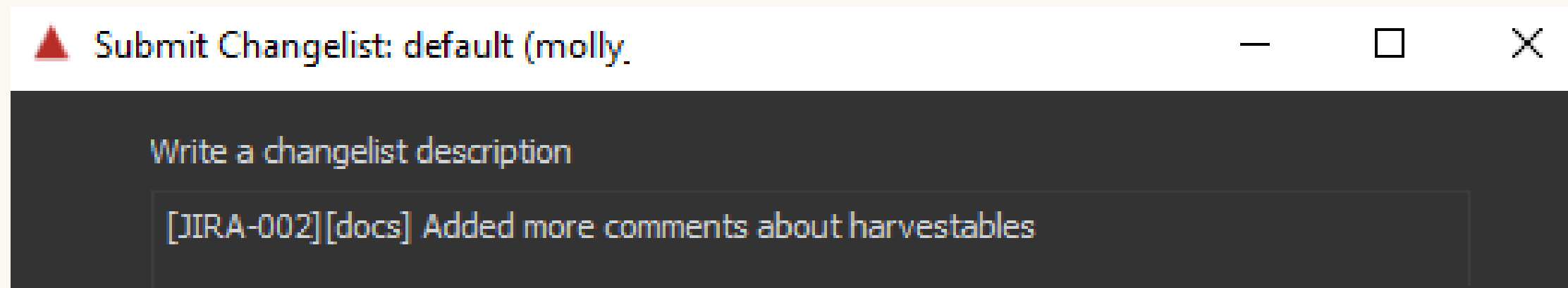


Formal or Informal Diagrams



Write docs for the next person

- Reinforce what you've learned and help the organization



Takeaways

- Keep an open mind
- Start with data, start small
- Ask targeted questions
- Don't worry about learning everything at once

Contact: @UltraRat, www.MollyJameson.com