



Creating a game with REXX – Part 2

Mike Beer

2026/05/04



Agenda

- Context
- Scripting
- Tools
- Outlook



Rexx & Games

- AI
 - Translation of texts
 - Voice generation
- OORexx
- → Till Winkler
- Story scripting
 - Which scenes
 - Scene definition
 - Sequence of scenes
- Regina
- → Mike Beer





Embark on an adventure of Hope to find Galene

Serious Life Coaching Game

Michael Beer, CEO
mike@mindcoa.ch

The Story

PROBLEM STATEMENT

Rising demand for *mental health* support far exceeds the availability of specialists, while current digital solutions feel *boring* and make users lose *motivation* after a couple of weeks.

OUR SOLUTION



Serious Life
Coaching Game



Empowers users to *build resilience* and skills for *wellbeing*



Motivates people through gameplay and narrative



Grounded in *research* and continuously evaluated *effectiveness*

WHY NOW?

- ✦ Gamification drives higher engagement & adherence than traditional methods*

<https://www.businessresearchinsights.com/market-reports/self-improvement-products-and-services-market-108694>

<https://www.researchandmarkets.com/report/healthcare-gamification>

<https://www.betterhealthfacts.com/2025/06/gamifying-health-future.html>

GAME FEATURES

Engaging Story Line

Immersion through first person perspective



Keep users engaged with narrative driven progress



Fantastical world and Magical creatures



Problem Based Exercises

Practical challenges that build real-life skills.



Blended into the narrative to increase motivation



Scientifically validated instruments



Deep Diving Questionnaires

Questions and answers redesigned in gamified way



Understand yourself and your surroundings



Sparks curiosity and helps self reflect



WoL - Wholistic Approach

Understand your WoL and improve it

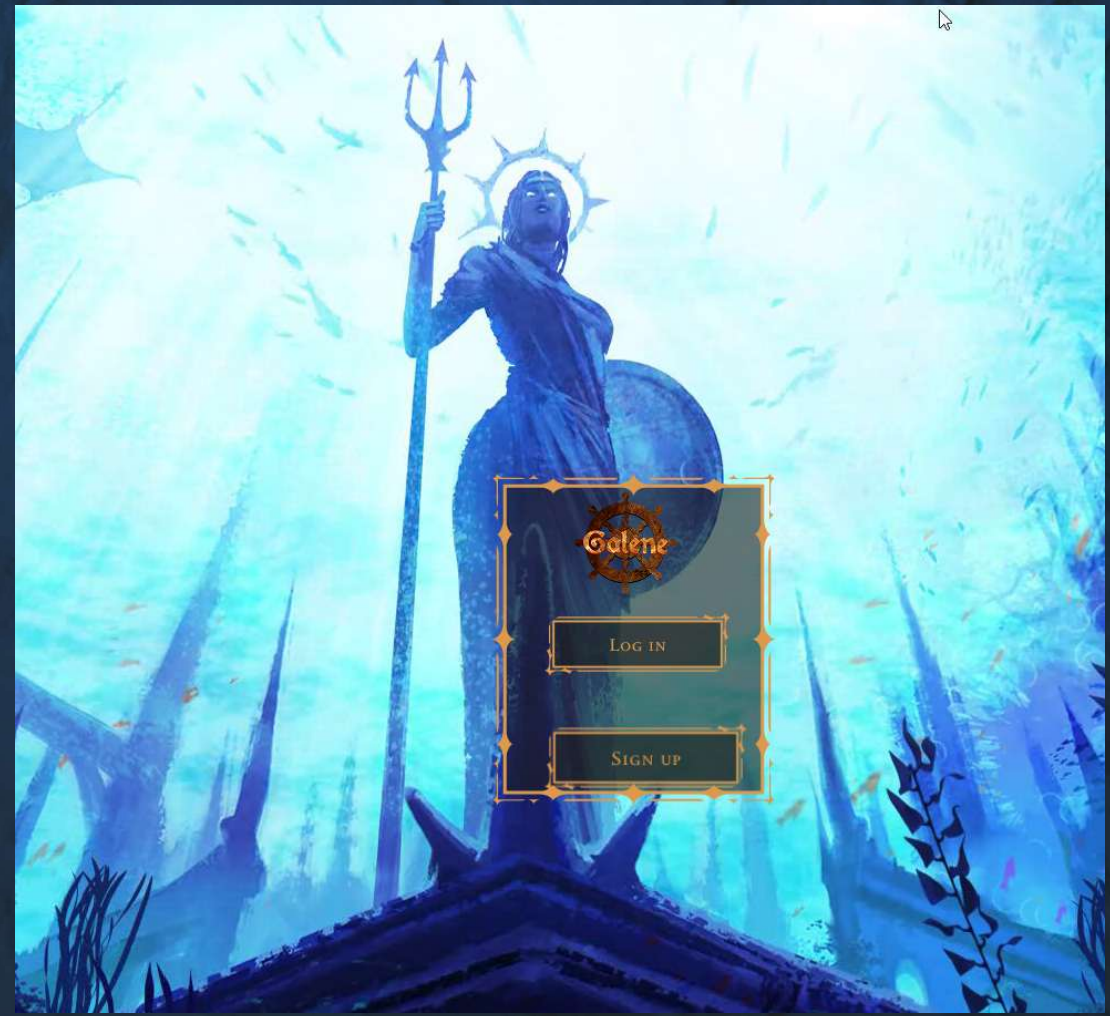
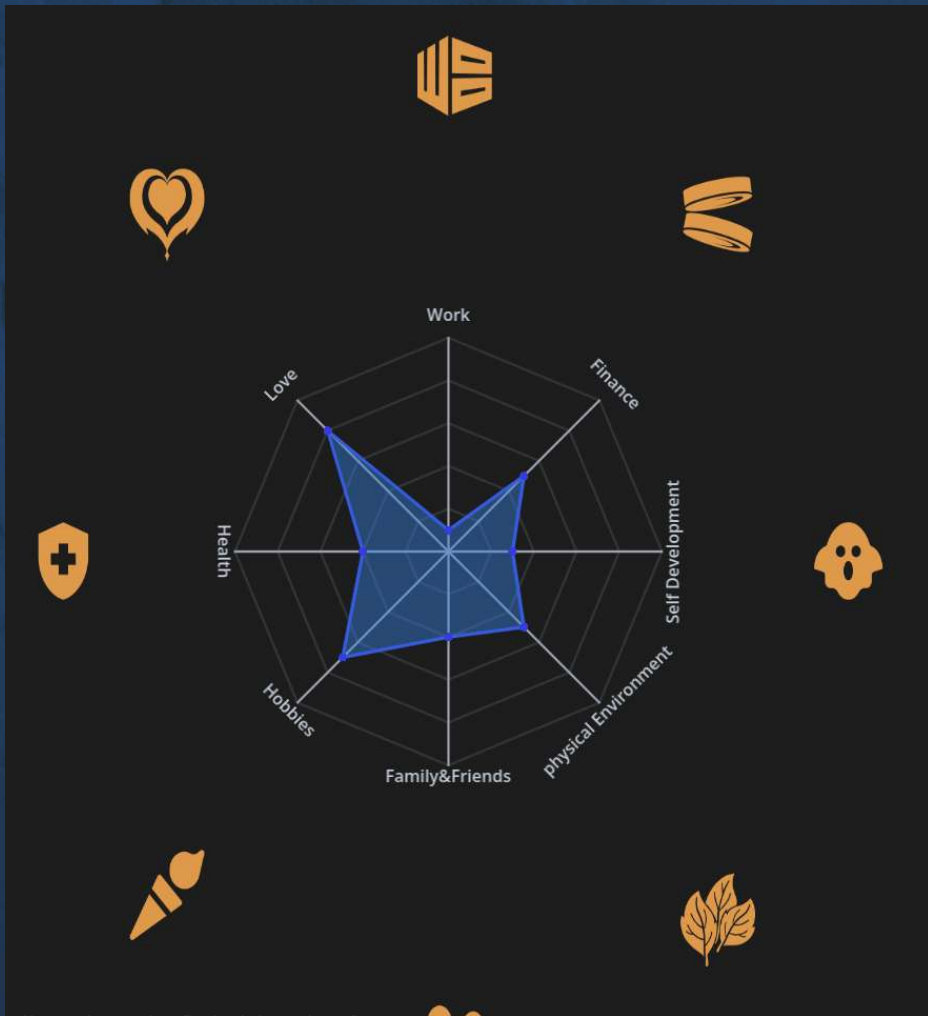


Track progress and grow as you play

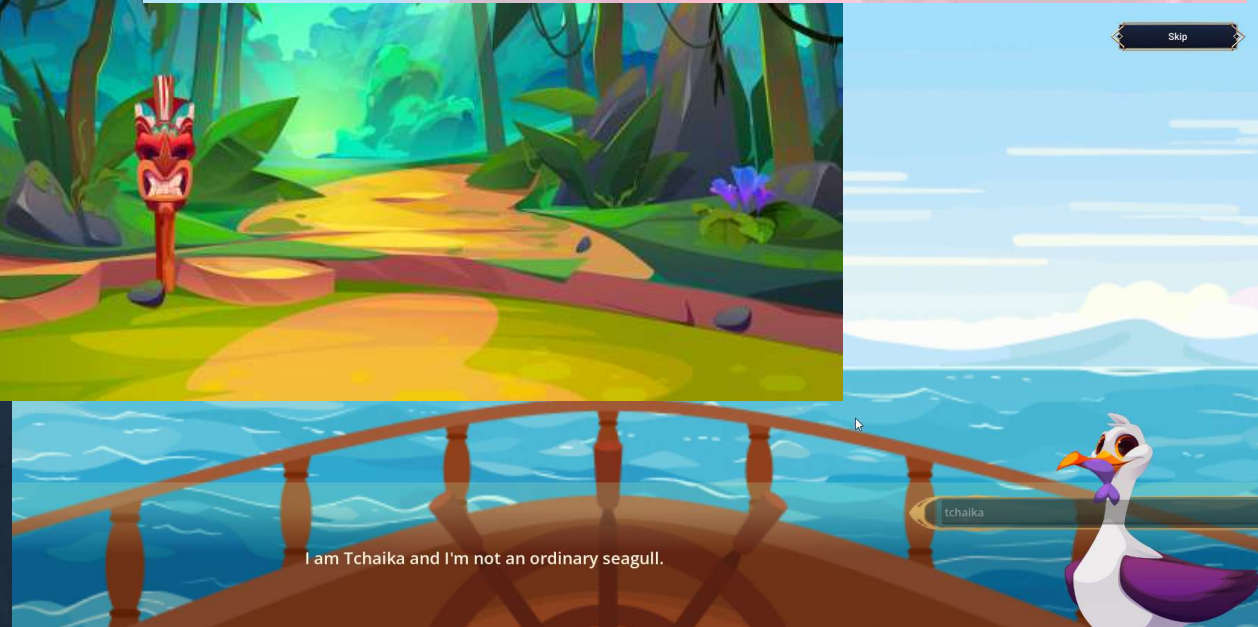
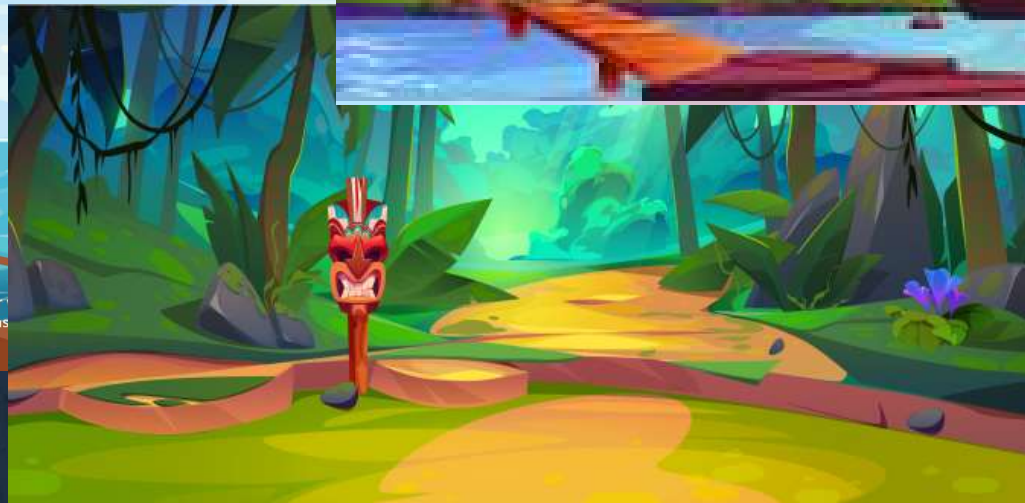
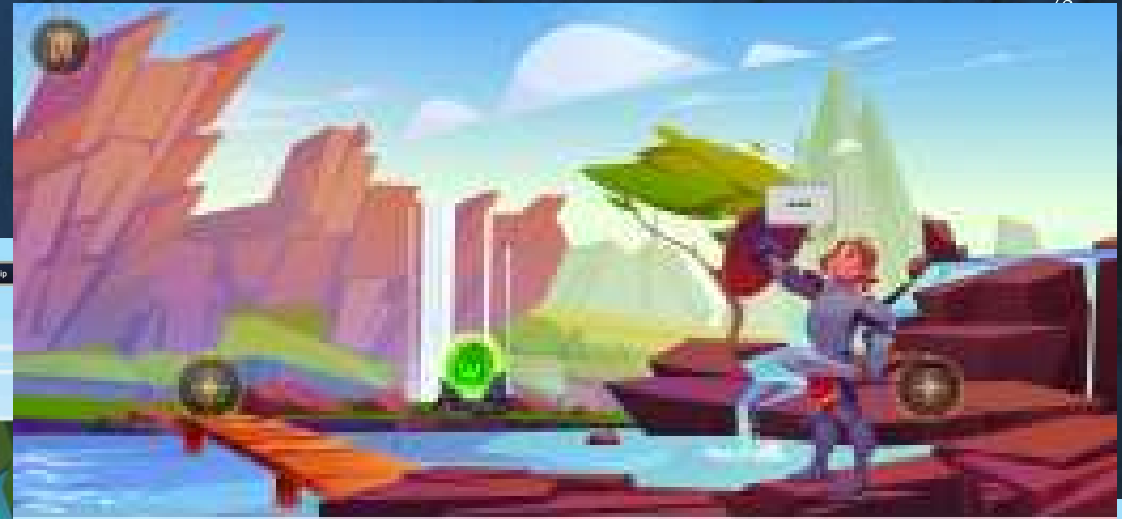


Cover all areas of Life.

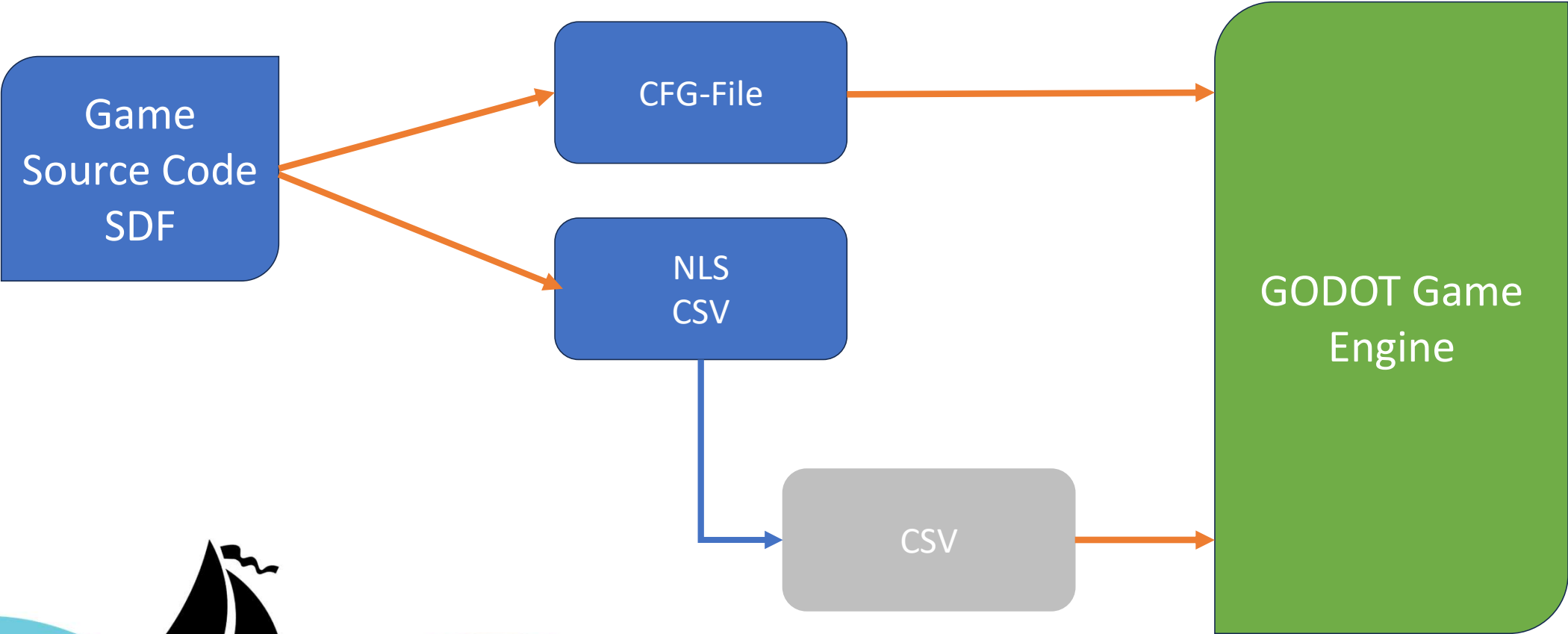




SNEAK PEAK



Structure



Scene (CFG-File)

- Media
 - Background Image
 - Background Video
 - Audio
- Flow
 - Next Scene(s)
- GODOT-Scene
 - .TSCN
- Parameters/Data



Scene Definition

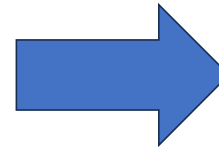
:SCENE.surprise

:BACKGROUND.black.jpg

:STAGE.dialog

:SLEEP.

:ESCENE.



Dialog Definition

:SAY pos=r npc=nymph.Traveler..Are you the chosen one?

:SAY pos=l npc=USER.I am not the chosen one. I am only a candiate.

:SAY pos=r npc=nymph.Maybe you will still be able to help me.,
I am being followed and do not have time to find someone else.

:SAY pos=r npc=nymph.A great evil has descended on my island of Gra,
and imprisoned our great guardian throwing the inhabitants into chaos.,
We need to urgently free him or panic will ensue on the lands.,
Please help us.

:SAY npc=*.With that urgent message the nymph vanished in a hurry,
as if sensing danger approaching.



Sharing Data between Modules

- COMMON AREA / LASTING GLOBALV
 - SDF: :COMMON VAR=text.Resume game where you left
 - Gdscript: put_common(), get_common()



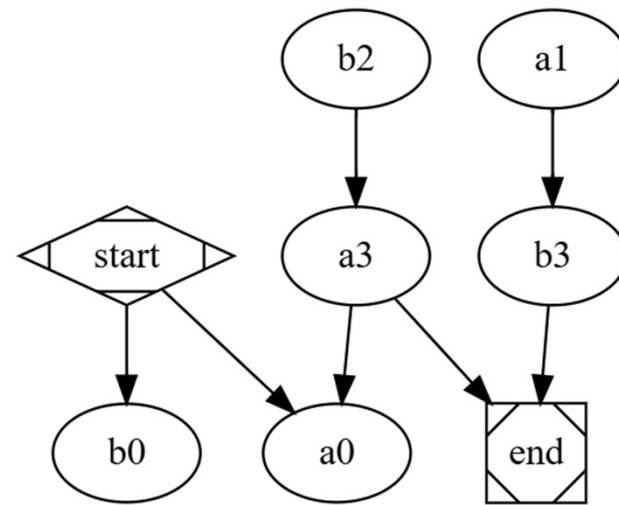
Visualize the Story: Graphviz

- Graphviz is open source graph visualization software. Graph visualization is a way of representing structural information as diagrams of abstract graphs and networks. It has important applications in networking, bioinformatics, software engineering, database and web design, machine learning, and in visual interfaces for other technical domains.
- <https://graphviz.org/>



<https://graph.flyte.org/>

```
digraph G {  
  start -> a0;  
  start -> b0;  
  a1 -> b3;  
  b2 -> a3;  
  a3 -> a0;  
  a3 -> end; b3 -> end; start  
  [shape=Mdiamond]; end [shape=Msquare];  
}
```



CFG -> GraphVIZ

```
; STORY/CHAPTER Galene/GAME_START  
; SOURCE FILE = gra.SDF
```

```
island_port -> sea_travel [label="sea"]  
island_port -> $target_port; [label="land"]
```

```
[scene]  
path="res://assets/scenes/templates/dialog.tscn"
```

```
[next]  
sea="res://assets/scenes/configs/sea_travel.cfg"  
land="res://assets/scenes/configs/$target_port;.cfg"
```



Next Steps

To Do

- Integrate consistency check into CFG-generation
- Integrate NLS/TTS processes into generation
- Integrate documentation process

- AUTOMATE as much as possible
- Game development is very iterative and agile



GODOT programming

- [GScript \(all versions\)](#)
- [C# \(.NET version\)](#)
- [C++ \(via GDExtension\)](#)
- [D](#)
- [Go](#)
- [Java/Kotlin](#)
- [Nim](#)
- [Rust](#)
- [Swift](#)
- [Odin](#)



GD Script

- Python-like

extends Node2D

```
func _ready():
```

```
    print("Hello, Godot!")
```



C#

```
using Godot;
public class HelloWorld : Node2D
{
    public override void _Ready()
    {
        GD.Print("Hello, Godot!");
    }
}
```



Issues



Rexx?



GDExtension

- **GDExtension** is a Godot-specific technology that lets the engine interact with native [shared libraries](#) at runtime. You can use it to run native code without compiling it with the engine.



The screenshot shows a webpage for GodotCon Boston 2025. The header is blue with the text 'GodotCon Boston 2025' and a 'login' link. Below the header, the event title 'Adding new Script Languages to Godot' is displayed in bold. To the right of the title are two icons: a calendar icon with '.ical' and a green speech bubble icon. Below the title, the date and time '05-06, 15:45-16:45 (EST5EDT), Deborah Sampson' are listed. The main body of text describes the talk: 'Godot has two major languages it supports for scripting: GDScript and C#. But what if you wanted to write your Godot scripts in a different language? With the new GDExtension system in Godot 4+, we now have the ability add other languages. This talk will take you through the process of binding a language to Godot, using examples from Godot Dart. It will take you through some of the inner workings of GDExtensions, the pitfalls of using GDExtension to add a Script Language, and various ways for new languages to interact with core Godot features, addons, and other extensions.'



Next Step

- Evaluate REXX-Integration into GODOT
 - C-Binding
 - Objects
 - Inheritance
- Call REXX from GODOT (gdscript)
- Enhance/Replace gdscript



What does AI say? ;-)

ChatGPT

create a wrapper for godot 4.6 to use oorexx instead of gdscript

- That's not a small “wrapper” job—you're essentially talking about embedding a full scripting language runtime into the Godot Engine and wiring it into its scripting API the way GDScript, C#, and GDExtension languages work. It's doable, but it's closer to writing a language binding/plugin than a thin wrapper.



create a language binding for godot 4.6 to use oorex instead of gdscript

- If you want to use ooRexx *as a first-class scripting language* inside Godot Engine 4.6, you're building a full **ScriptLanguage binding via GDExtension**—not just a wrapper.
- You are implementing Godot's scripting interfaces:
 - ScriptLanguage → registers "ooRexx"
 - Script → represents .rex files
 - ScriptInstance → runs code attached to nodes
 - Conversion layer → Godot Variant ↔ ooRexx objects



Hard problems you will hit

- **Lifetime management**

- Godot objects vs Rexx

- **Object binding**

- Wrapping Godot objects into Rexx objects

- **Performance**

- Script call overhead

- **Error handling**

- Mapping Rexx exceptions → Godot errors



A more practical hybrid (often better)

- Instead of full replacement:

- Keep GDScript
- Use ooRexx for:
 - AI scripting
 - modding
 - external logic

- Expose:

Variant call_rexx(String code);

- This gets 80% of value for 20% effort



So some brain food....

Thank you for your attention!

