Winter Games University 2011

blog.andreasstiegler.com/wgu2011

Artificial Intelligence in Games

Andreas Stiegler
Stuttgart Media University
stiegler@hdm-stuttgart.de

Artificial Intelligence

What are we talking about?

AI as part of Game Mechanics / Gameplay

Pathing, group behavior, automated features, advisors, storytelling, analysis, ...

Situational AI

Racing games, simple shooter bots, MMOG creatures, RTS units, ...

Tactical AI

Shooter bots, strategy games, crowds (zombie shooters!), ...

Strategic AI

Not much yet. Perhaps strategy games or dynamic RPG content in the future?

Player-level AI

Artificial Intelligence

And why are we talking about that?

Cause here is a market for it!

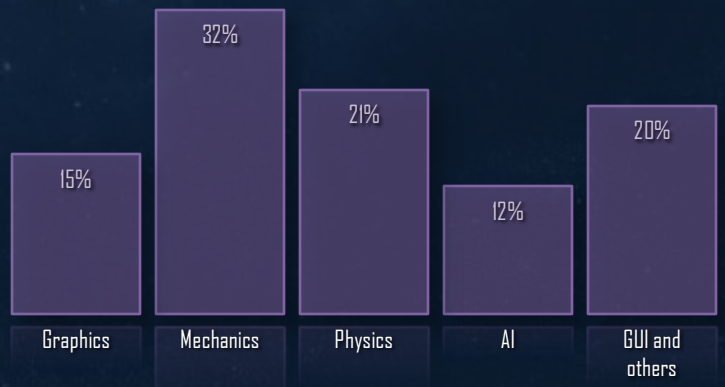
... and because it offers unique possibilities!

Artificial Intelligence

What are the challenges?

Performance!

Graphics, Physics, Game Mechanics, Sound, AI and whatnot all in real-time?!



Artificial Intelligence

What are the challenges?

Real-Time!

Yea, as said: Thinking shouldn't take hours. "soft real-time"

However, we don't have to calculate goals and make decisions (in every decision domain) at real-time. Lucky, the most complex decisions (strategic decisions) can have far greater reeval intervals.

Artificial Intelligence

What are the challenges?

Consistency!

Most games will require to continuously work towards a goal without changing ones mind every few seconds.

Especially vital for strategy games (where AI is very important either way!)

image source: Fleet Operations, www.fleetops.net

Artificial Intelligence

What are the challenges?

Believable!

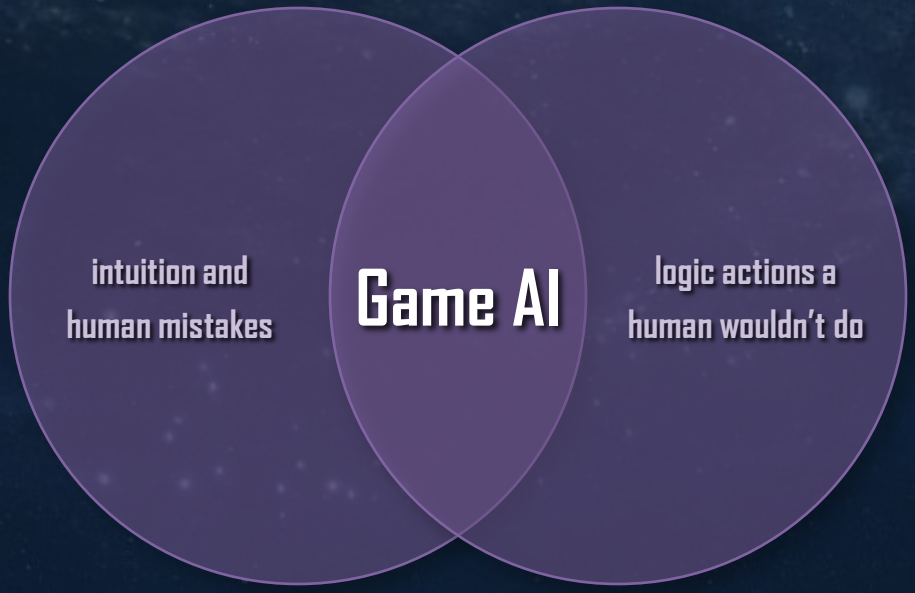
If a humans interact with the AI (friendly or hostile), the AI should produce a "believable" reaction pattern.

note the stupid word believable! :-(

Basics

Basics

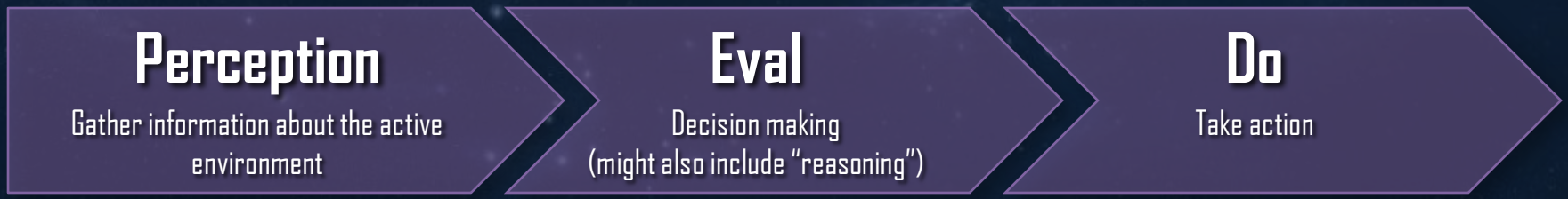
Before we start.. Artificial Intelligence in the context of gaming is a small subset of the vast field of scientific Artificial Intelligence!



Basics

Today, we will focus on – probably the most important – processing chain for real-time artificial intelligence

The **Perception-Eval-Do** chain



Basics

Where to do all the processing?

Precompiled

Runtime

Induced

World Extraction

World Extraction

In order to interact with the world, an AI has to gather information about the game world

Tends to be a very complex problem!

Static World

Architecture, Map, Location-based scripts

Dynamic World

Player Characters, NPCs, dynamic scripts

World Extraction: Static

Waypoints are a common technique to precompile information about the topology of a game world into a fast-accessible data structure.

Nodes of the waypoint grind may also store and accumulate location-based information to map memories:

Threat Map

image source: World of Warcraft, Activision Blizzard

World Extraction: Static

Similar to Way Points:
Regions of Interest

Often used in strategy games to store strategic information for the AI.

Demo: **Supreme Commander Editor**

World Extraction: Dynamic

Dynamic entities – like players – tend to have multiple states. An AI will have to extract and filter the required information

Data Extraction

Ingame Universe

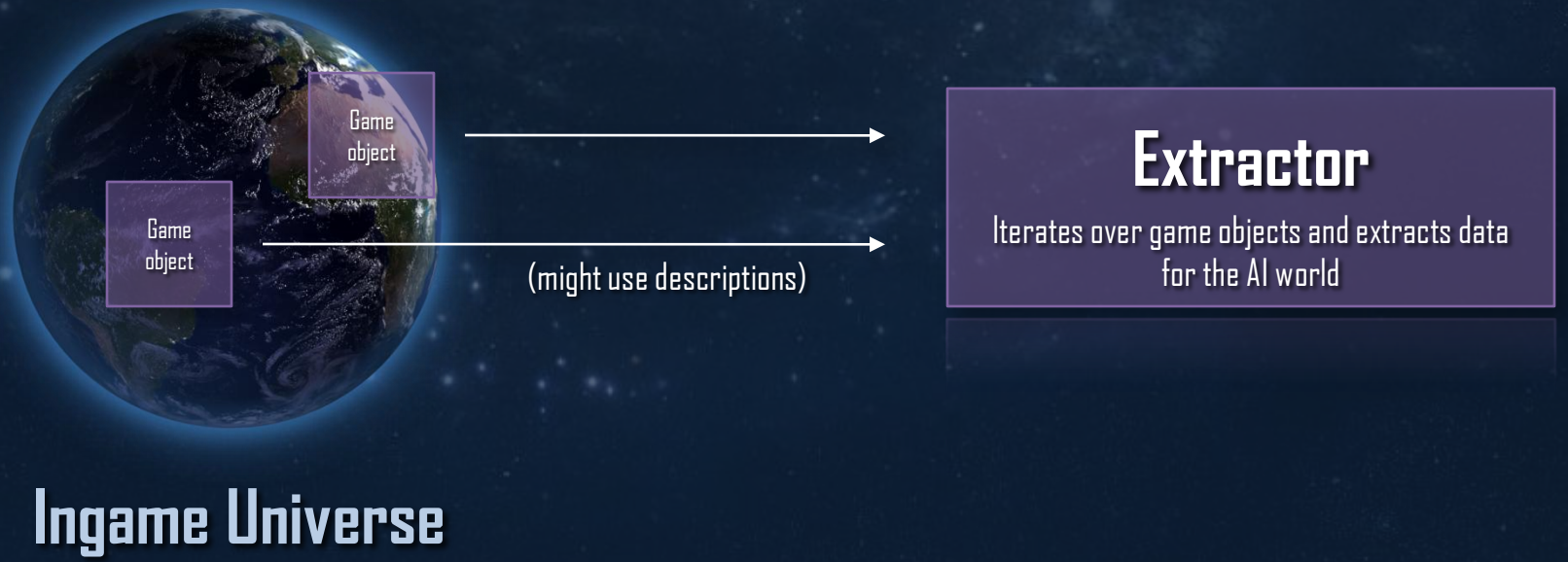
AI Universe

World Extraction: Dynamic

There are various methods to achieve that, like using description objects, extractors or direct API interfaces.

World Extraction: Dynamic

There are various methods to achieve that, like using description objects, extractors or direct API interfaces.



Situational AIs

Situational AIs

- Decisions based on scripts and parameterized rules
- No memories at all – well okay except states
- Good performance, precompiled

MMOGs

lots of creatures, lots of players

Sport Games

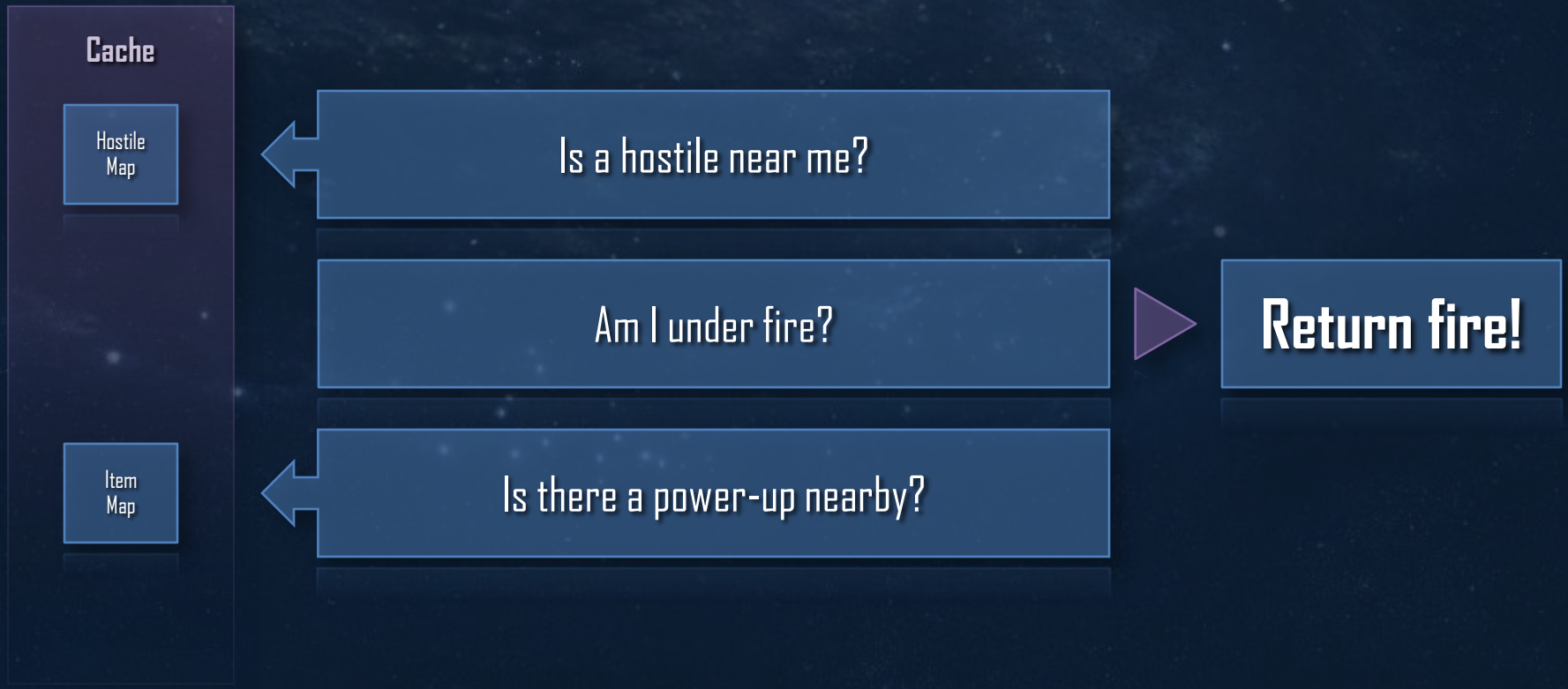
Well defined rule set

Shooters

Tactical aspects are part of the game play

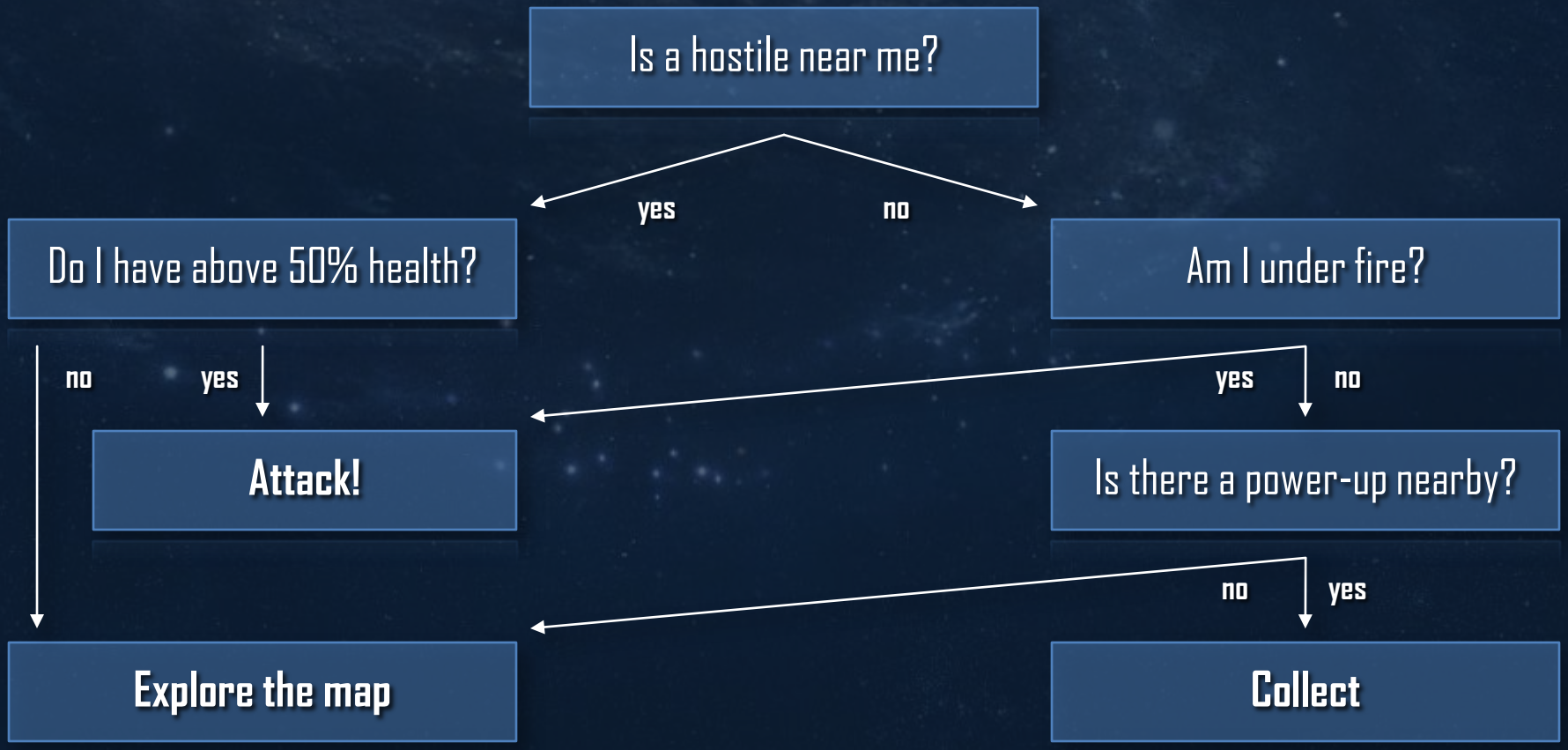
Situational AIs

There are various implementations for rules. The most straight forward is probably a plain list with a caching notepad.



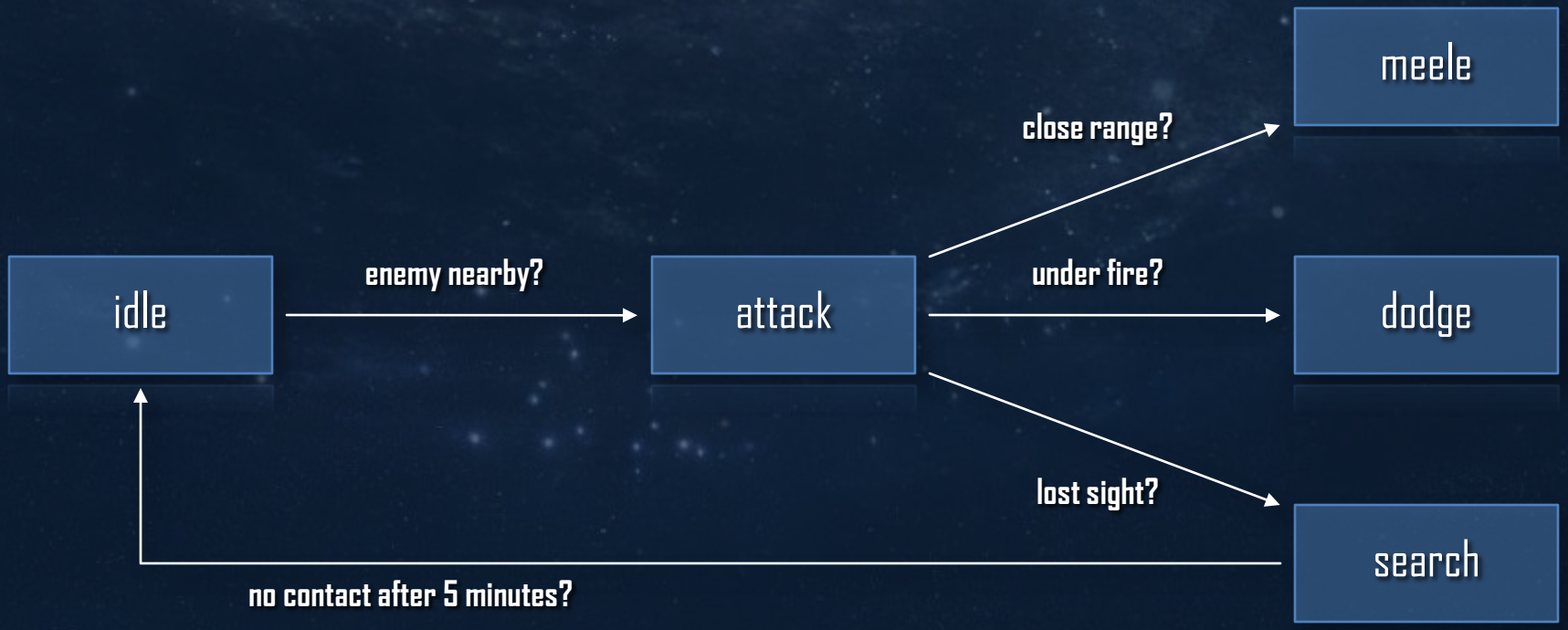
Situational AIs

There are various implementations for rules. A very common implementation is a tree-based evaluation.



Situational AIs

There are various implementations for rules. A very straight forward solution for game developers is a finite-state machine.



Situational AIs

- Very good performance
- Good caching capabilities
- Low memory footprint
- High control over AI-controlled characters
- Relatively high development speed once some rules are there

Situational AIs

- A lot of precompiled content: Will increase compile time
- The AI will always fall to the same mistakes
- Only static behavior is possible – The AI is predictable
- The AI can't adapt to new content
- High development time to get a first efficient rule-set
- No way to find trade-offs. Rules either break or don't
- CRUEL debuging