



## Learning play finder

Conception and Implementation of a learning play finder for the robot soccer team Tigers Mannheim





# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders
- 3) Analyses
- 4) New play finder
- 5) Summary
- 6) Live demo





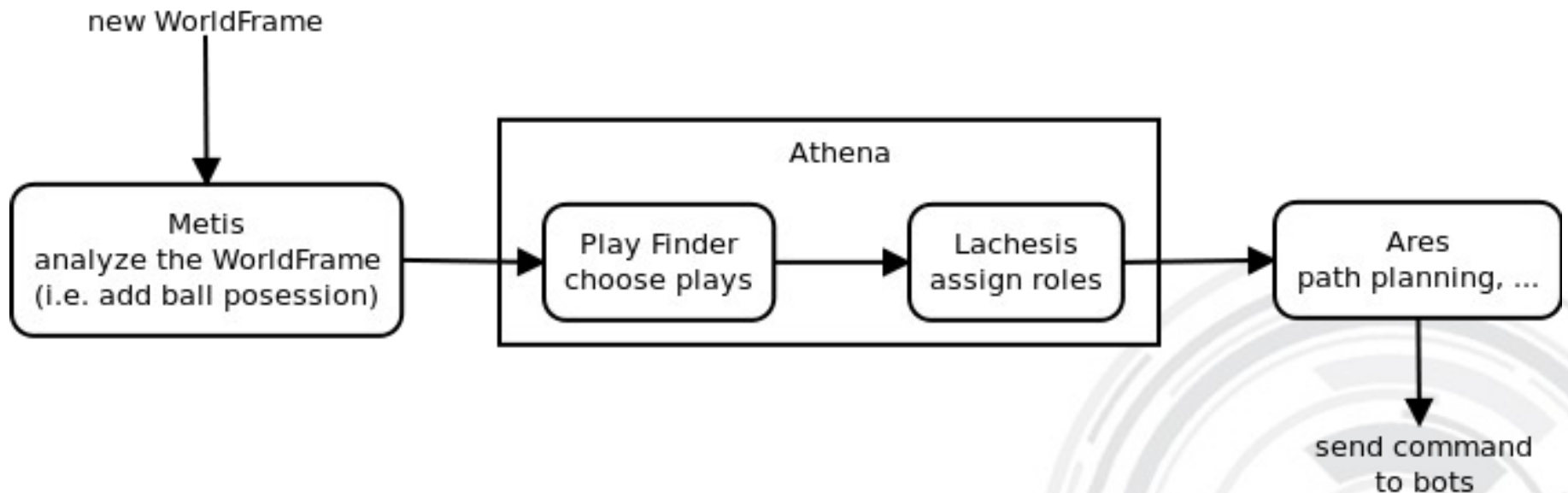
# Agenda

- 1) **Introduction**
- 2) Refactoring to allow several play finders
- 3) Analyses
- 4) New play finder
- 5) Summary
- 6) Live demo





# Component play finder





# Tasks of the play finder

- Main task: Choose plays
- Input:
  - Current field
  - Current plays
  - Referee commands
- Output:
  - Plays which should be used





# Former play finders

- Basic play finder:
  - Reacts only on referee commands
  - If there is no referee command, get the ball
  - If there is no referee command and we have the ball, shoot
- Point system:
  - Every play has the method `getPlayableScore`
  - Choose the plays with the highest score





# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders**
- 3) Analyses
- 4) New play finder
- 5) Summary
- 6) Live demo





# Several play finders

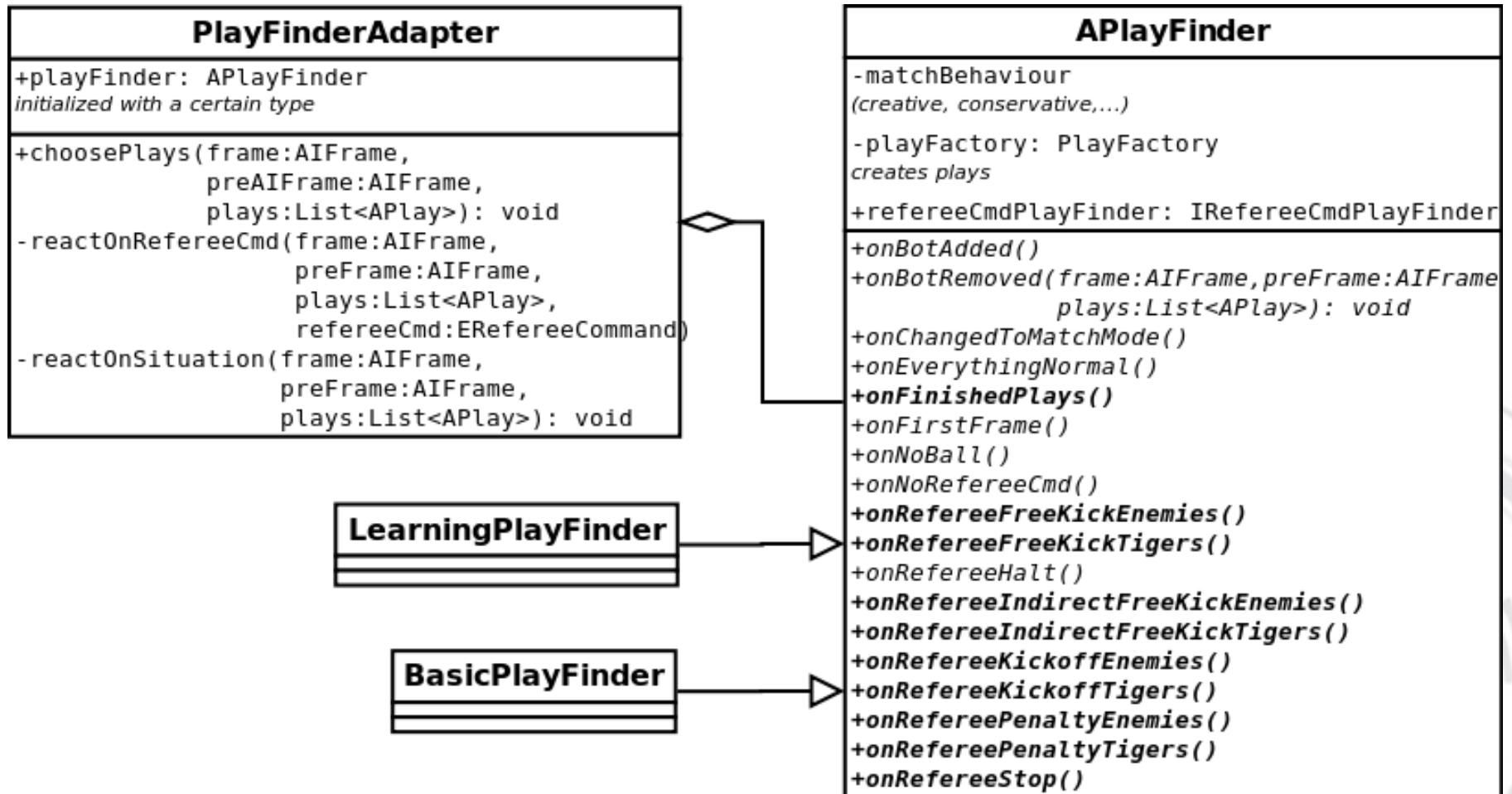
- Tigers Mannheim = group of students → high fluctuation
- Many new ideas → every component and every part of a component should be exchangeable
- Interface for a play finder
- Reuse and optimize existing Concepts
  - Playable score
  - Criteria







# Refactored structure of the play finder





# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders
- 3) Analyses**
- 4) New play finder
- 5) Summary
- 6) Live demo





- Performance measure
  - State of plays: success or fail
  - Goals
  - Referee interception
- Environment
  - Field and areas
  - Ball
  - Bots: own and enemies
  - Observable
  - non deterministic
  - multi agent based
  - discrete



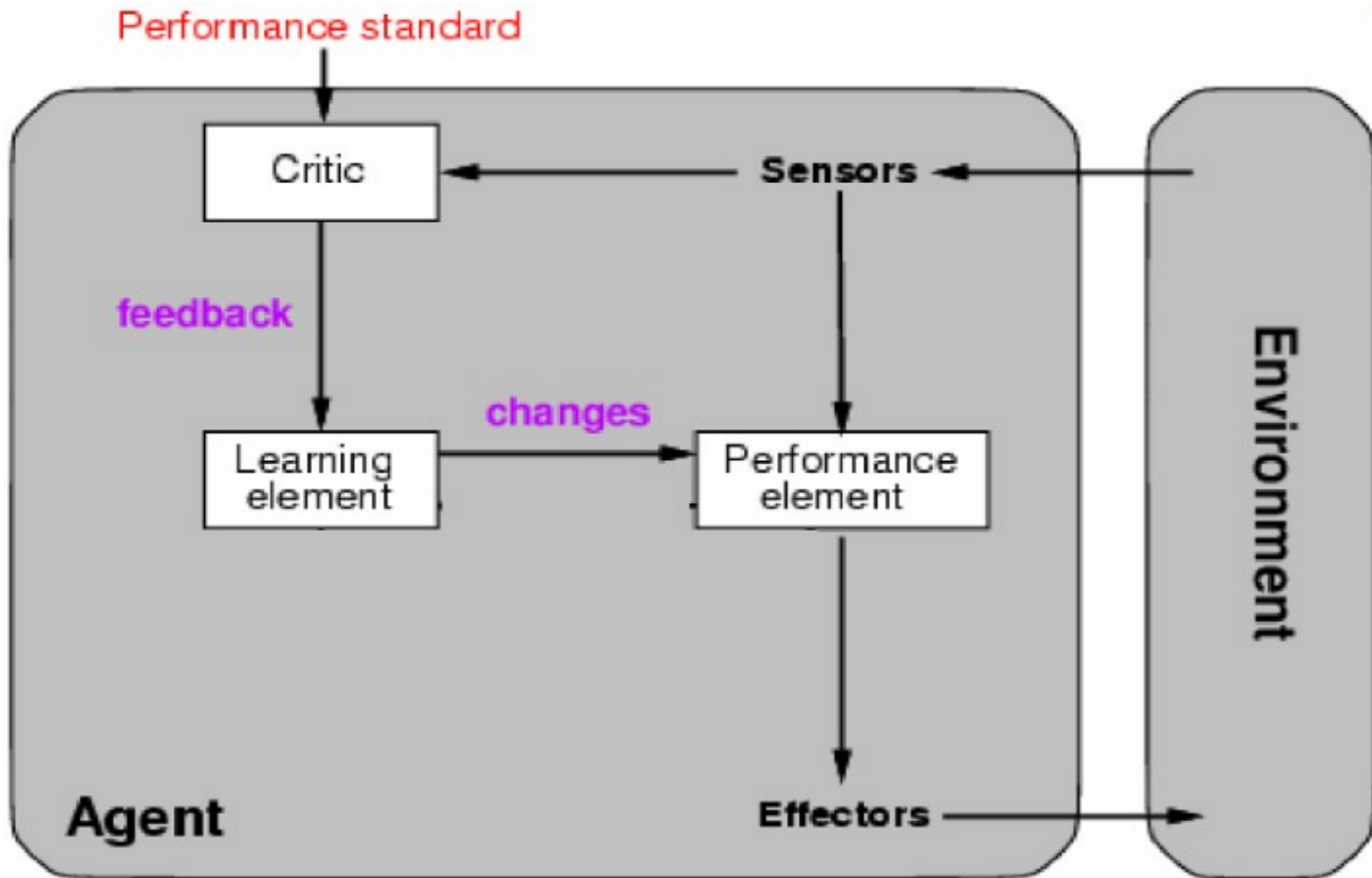


- Actuators
  - Bots
    - 4 wheels
    - Kicker
    - Chipkicker
    - Dribbling machine
- Sensors
  - Vision
  - WorldPredictor
  - Bot
  - Referee messages





# Agent type: Learning





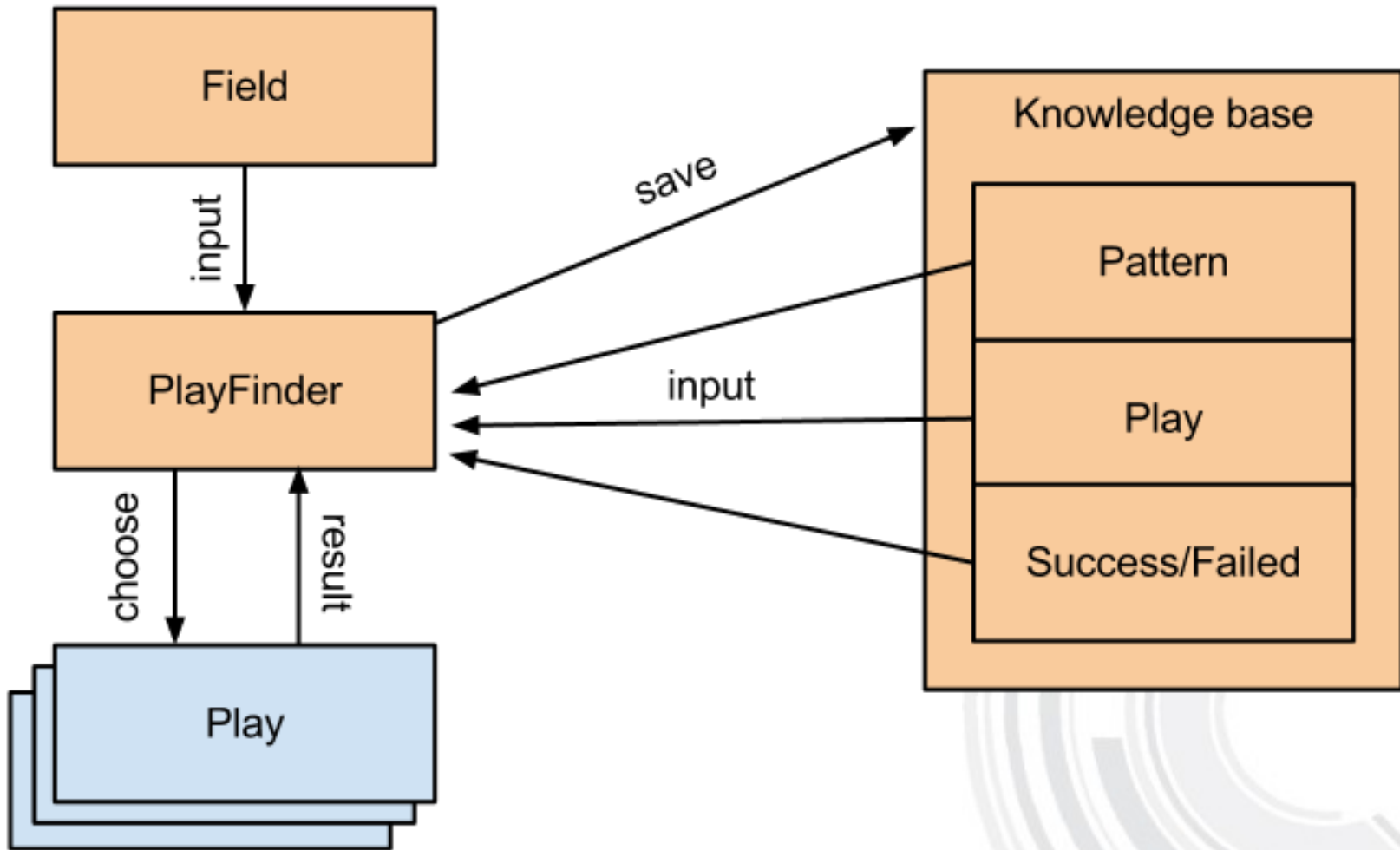
# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders
- 3) Analyses
- 4) New play finder**
- 5) Summary
- 6) Live demo



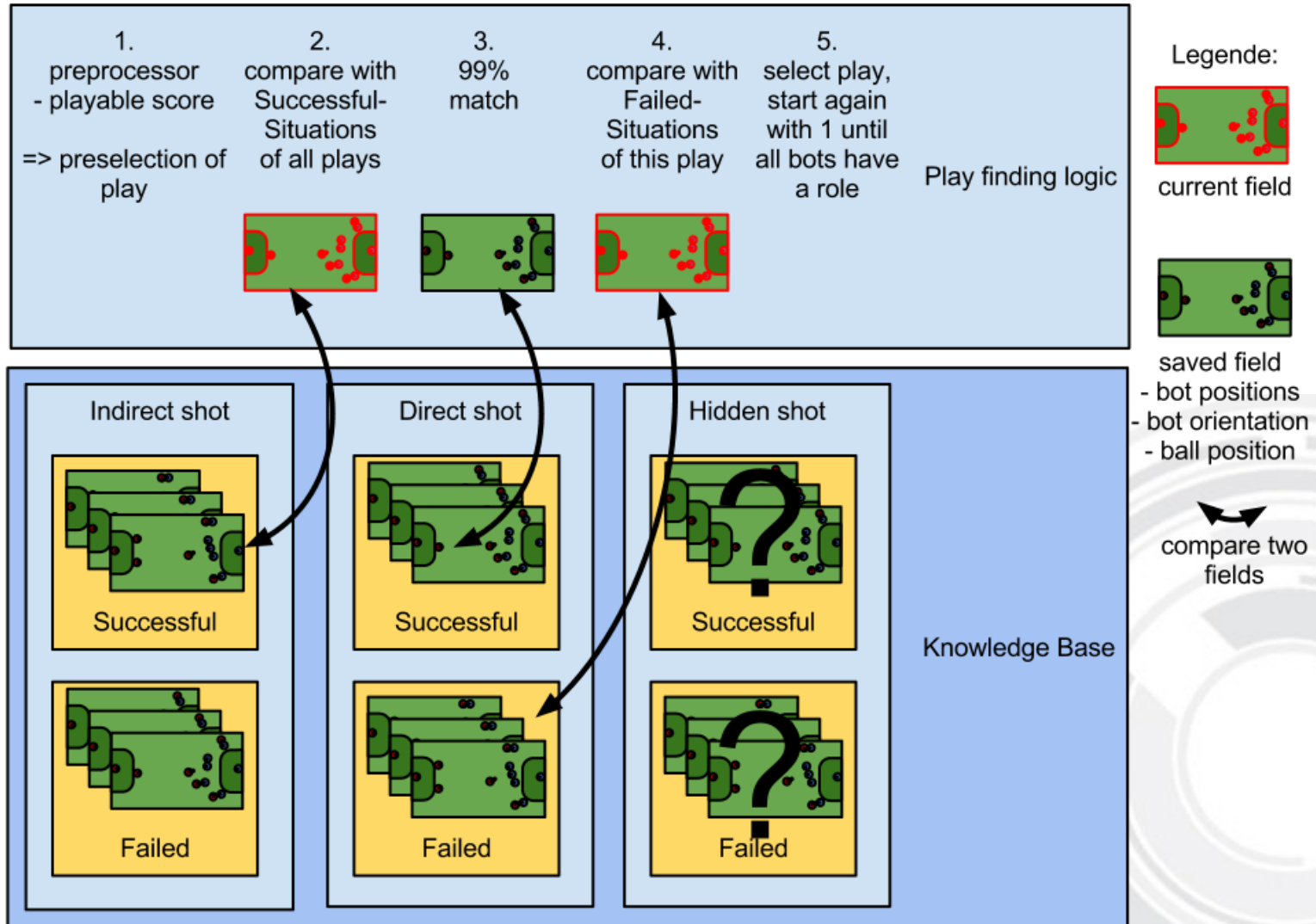


# Basic idea





# Information flow







# Compare Fields Idea

Whiteboard





# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders
- 3) Analyses
- 4) New play finder
- 5) Summary**
- 6) Live demo





# Conclusion

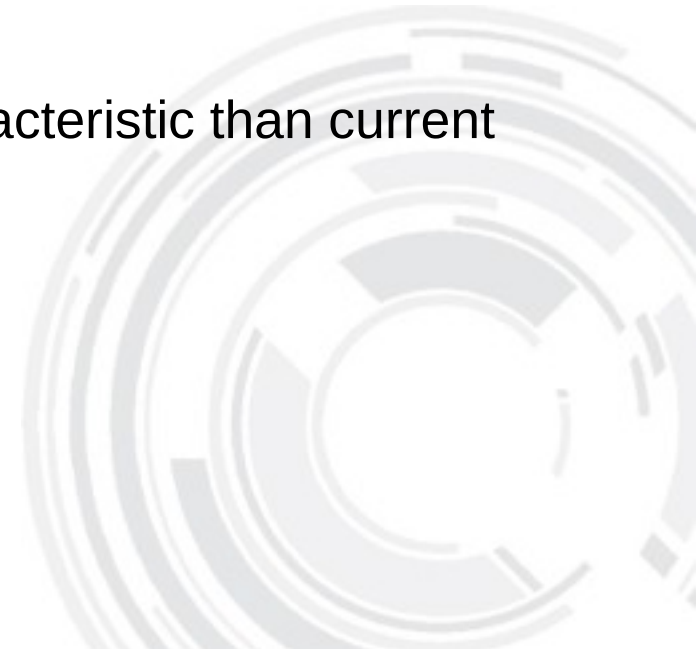
- Learning Play Finder
  - Knowledge base can grow
  - Decisions regarding the knowledge base are done
  - Integrated into Sumatra
  - 2 different field comparison algorithms implemented
- Sumatra
  - Refactoring of feedback of the plays
  - Reintegration of play finder into Sumatra





# Ideas for the future

- Replace success/failed by points, e.g.:
  - 10 pt for goal
  - 3 pt for getting a free kick
  - 1 pt for throw-in
  - -1 pt for throw-in of the enemy
- Comparison with a series of world frame
  - Increasing comparison result better characteristic than current field





# Ideas for the future

- Save knowledge base in a database
- Better comparison algorithm
- Acceptable match (at the moment: 0.8) changeable and learnable
- Different knowledge bases
  - General
  - Game





# Agenda

- 1) Introduction
- 2) Refactoring to allow several play finders
- 3) Analyses
- 4) New play finder
- 5) Summary
- 6) Live demo**

