

# Architektúra modellezés UML-ben

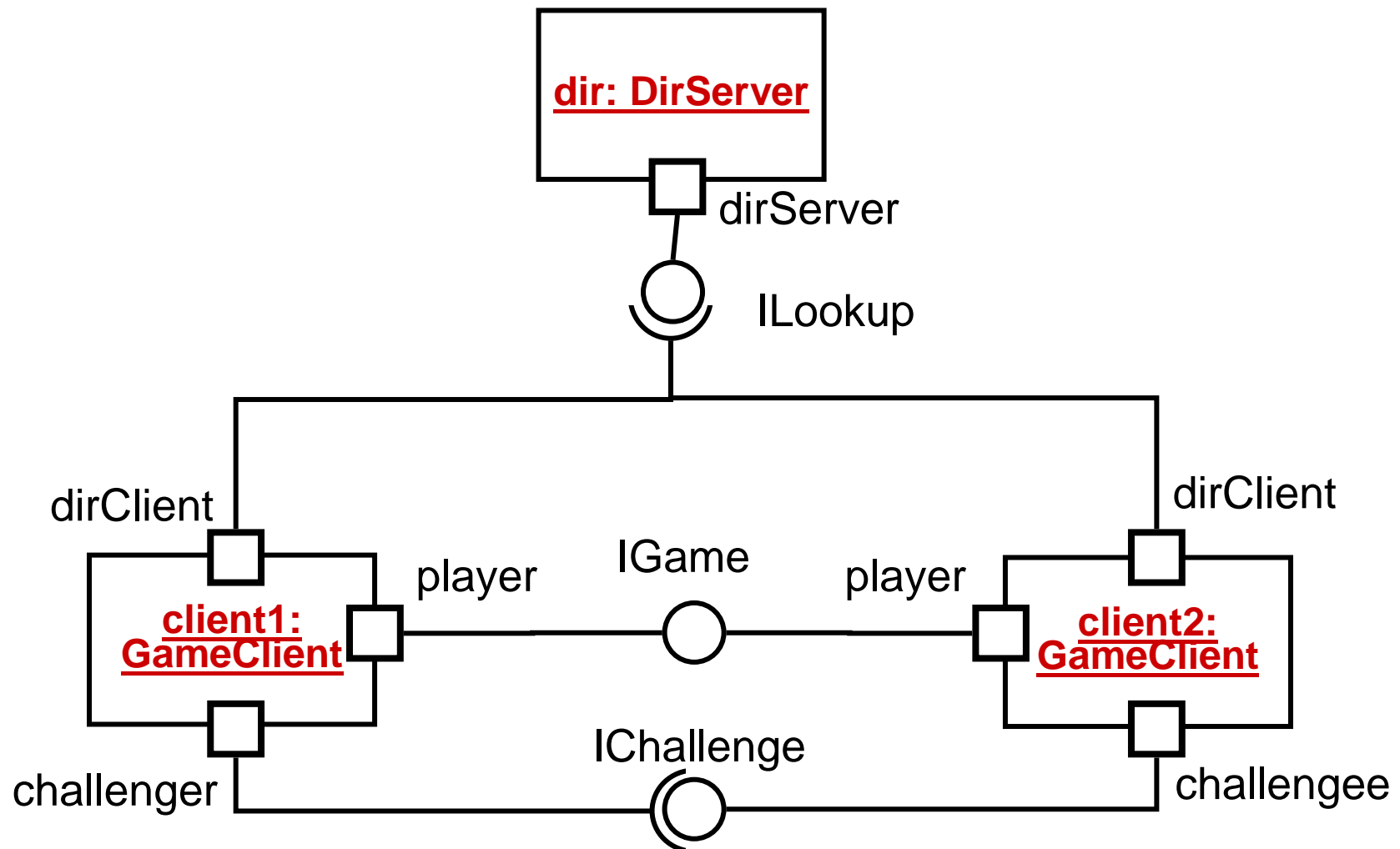
UML bázisú modellezés és analízis

# Table of Contents

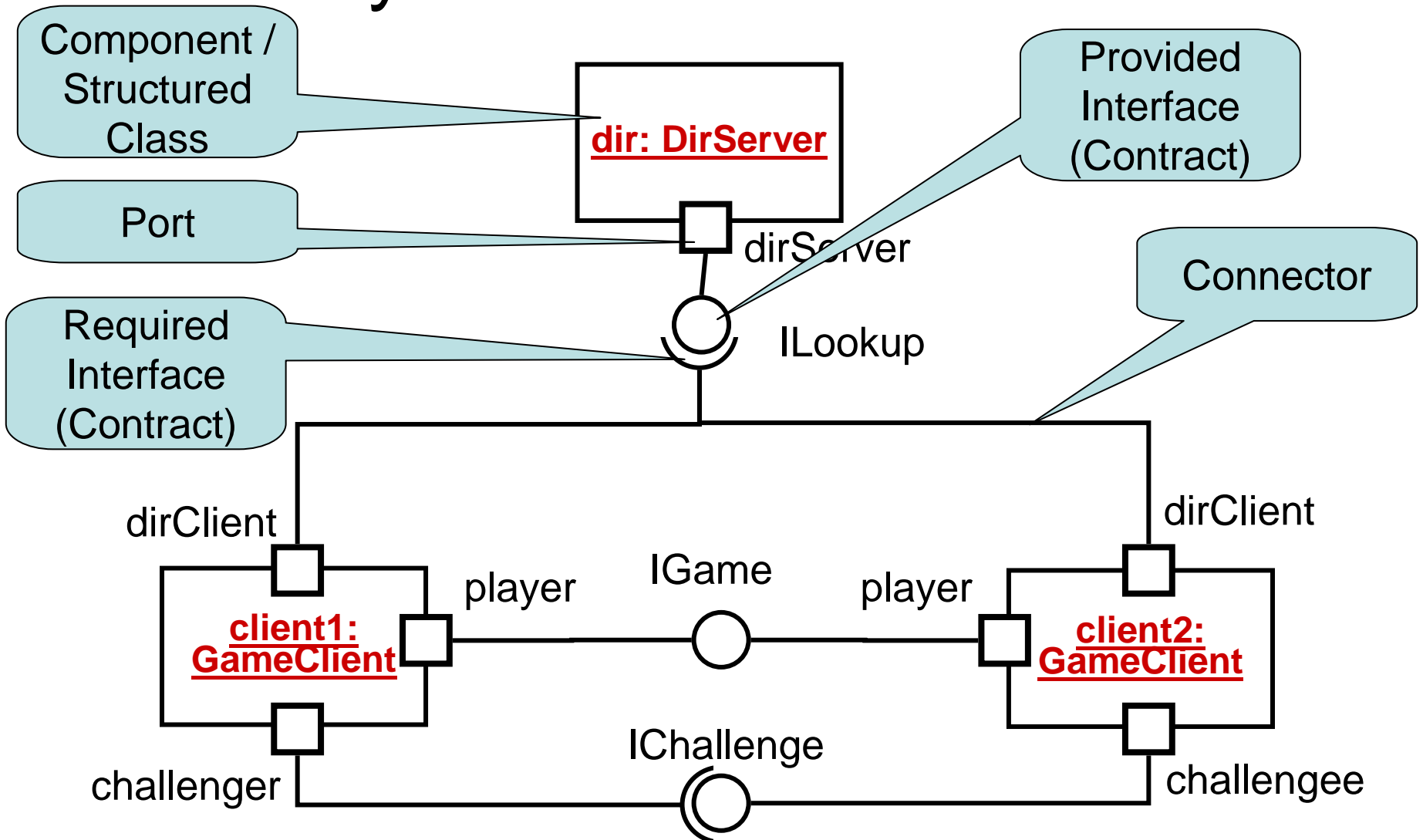
- Architectural styles of Web applications
- Components and interfaces  
(UML 2.0 composite structure diagrams)

# Architecture Modeling in Champs Manager

# Legacy Architecture: Directory Server and Game Clients



# Legacy Architecture: Directory Server and Game Clients



# Interface Details (Contracts)

<<interface>>  
*ILookup*

lookupClient(String userName): Client

## **Protocol statecharts:**

- detailed specification of interfaces
- specify the allowed call sequences on a class or interface

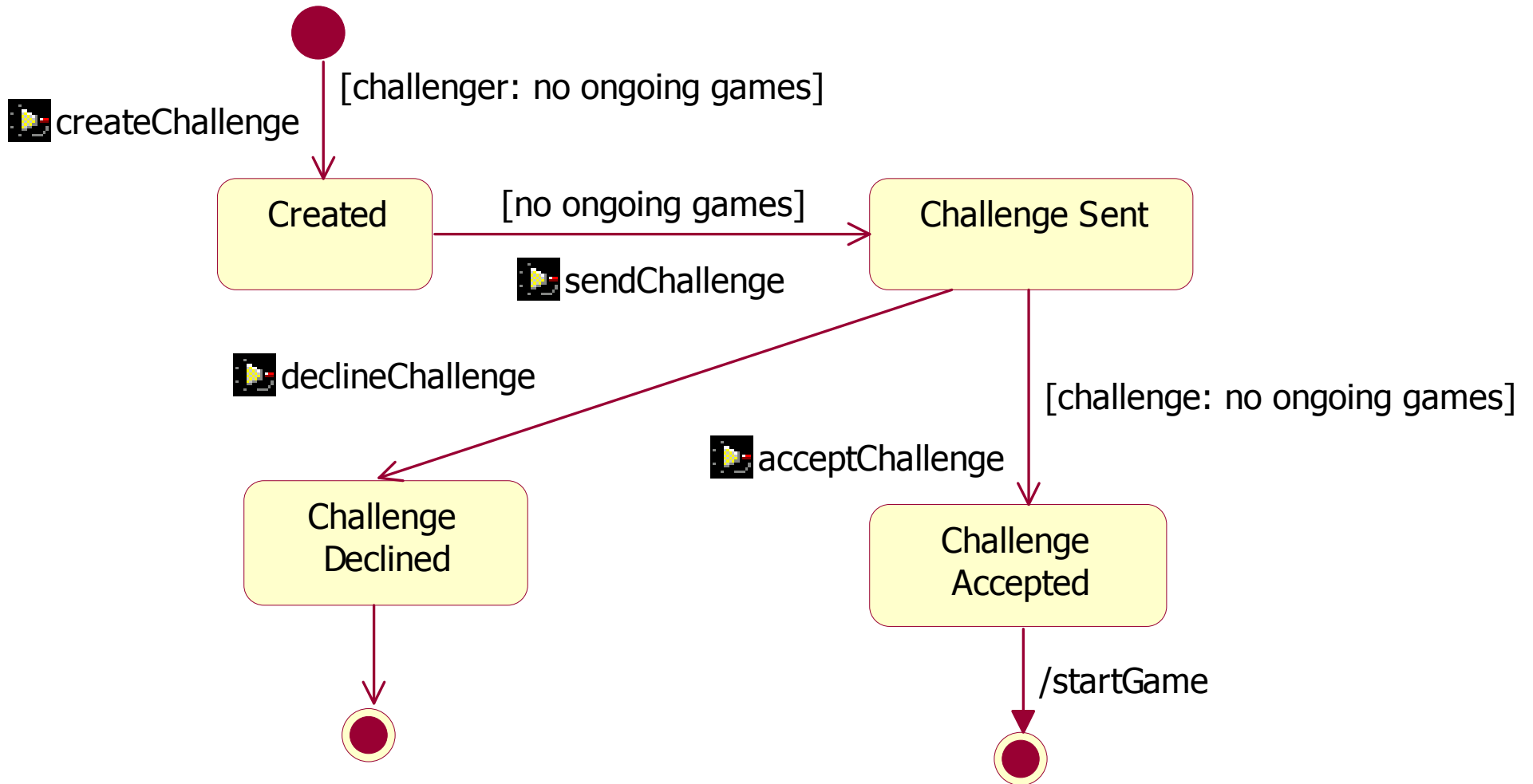
<<interface>>  
*IGame*

startGame(Client whi, Client bla): Game  
sendMove(Game gm, Move mv)  
endGame(Game gm)  
resignGame(Game gm)  
offerDraw(Game gm)  
acceptDraw(Game gm)  
rejectDraw(Game gm)

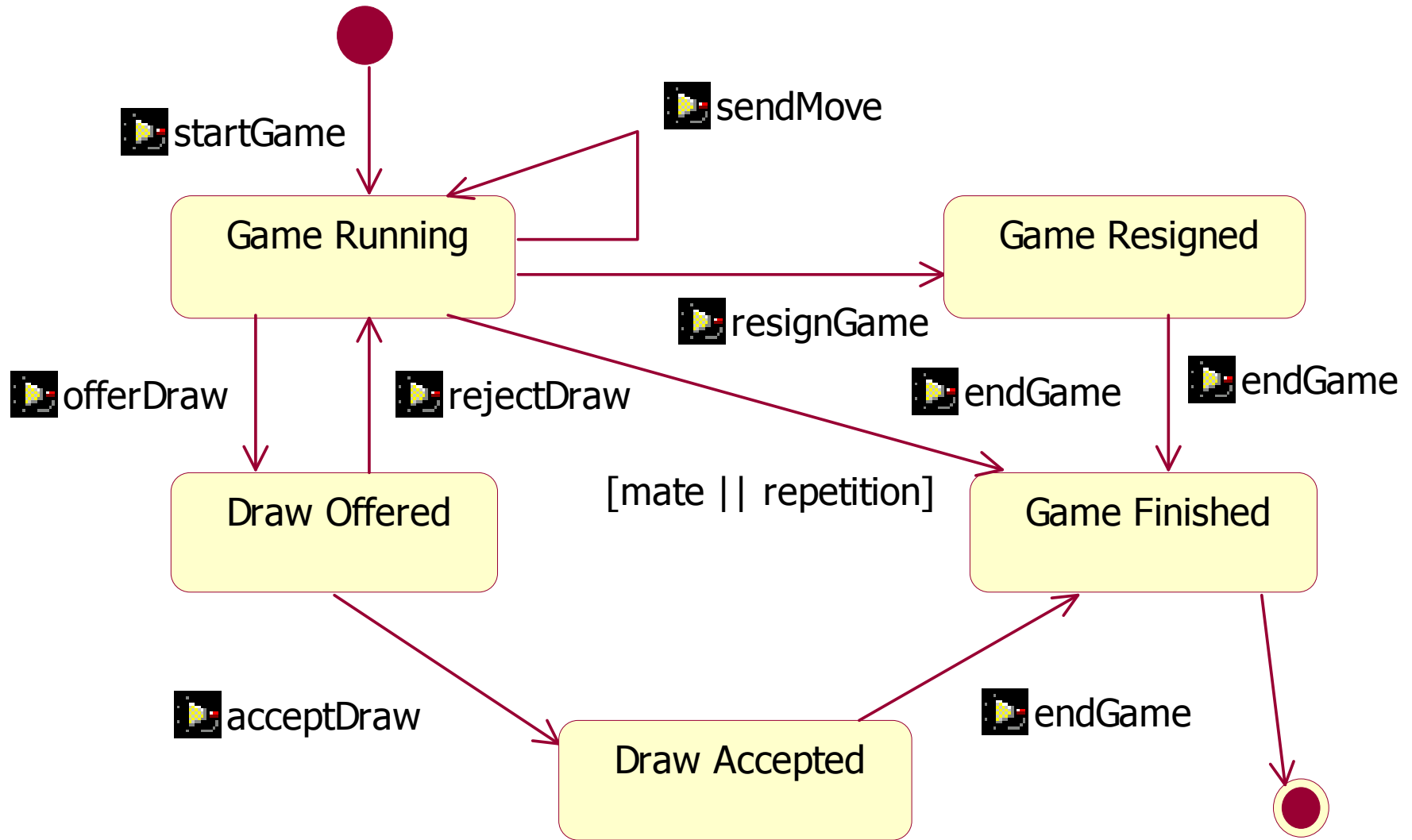
<<interface>>  
*IChallenge*

sendChallenge(Client snd, Challenge chal)  
acceptChallenge(Challenge c)  
rejectChallenge(Challenge c)

# Protocol Statechart of IChallenge



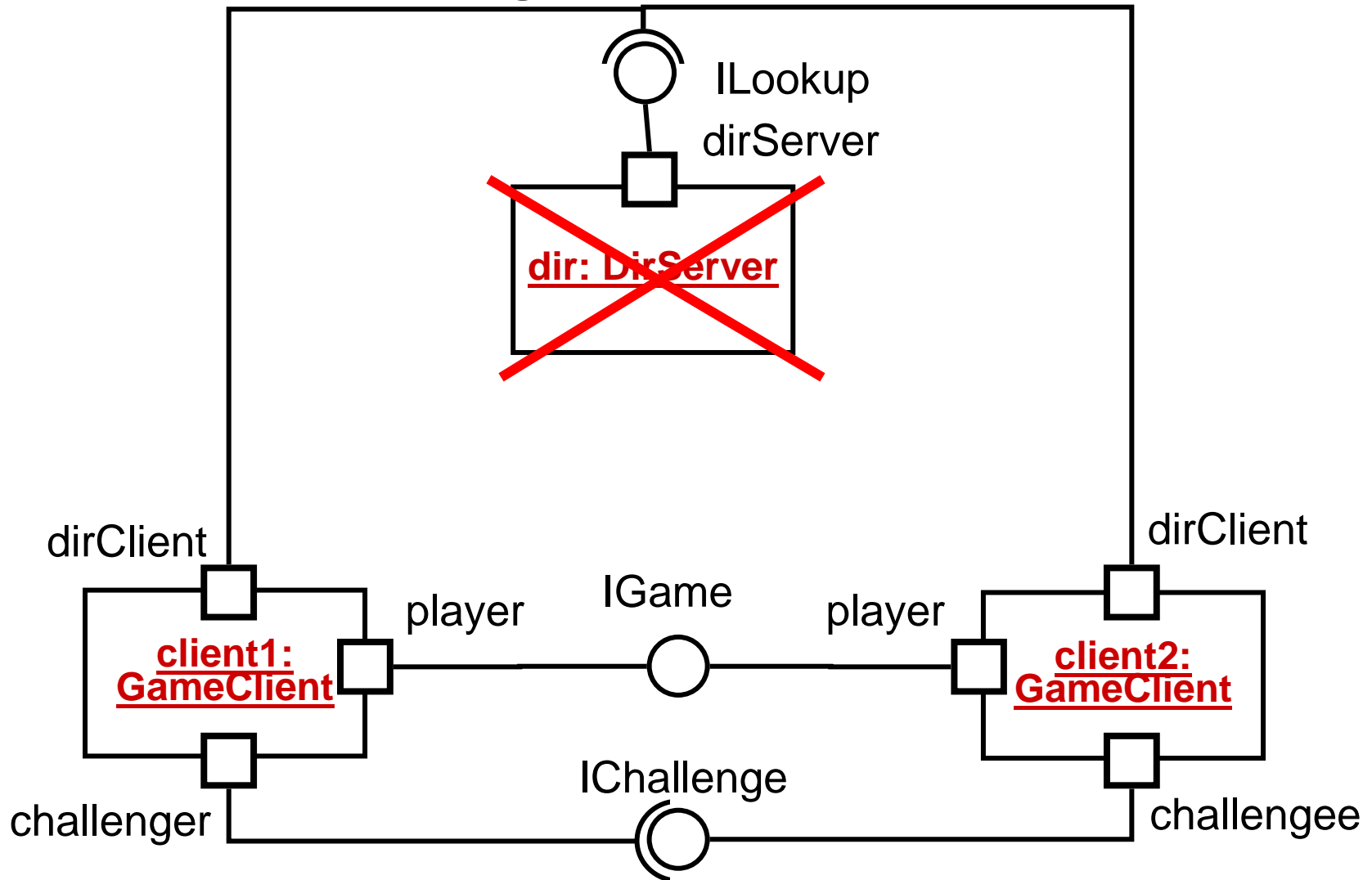
# Protocol Statechart of IGame



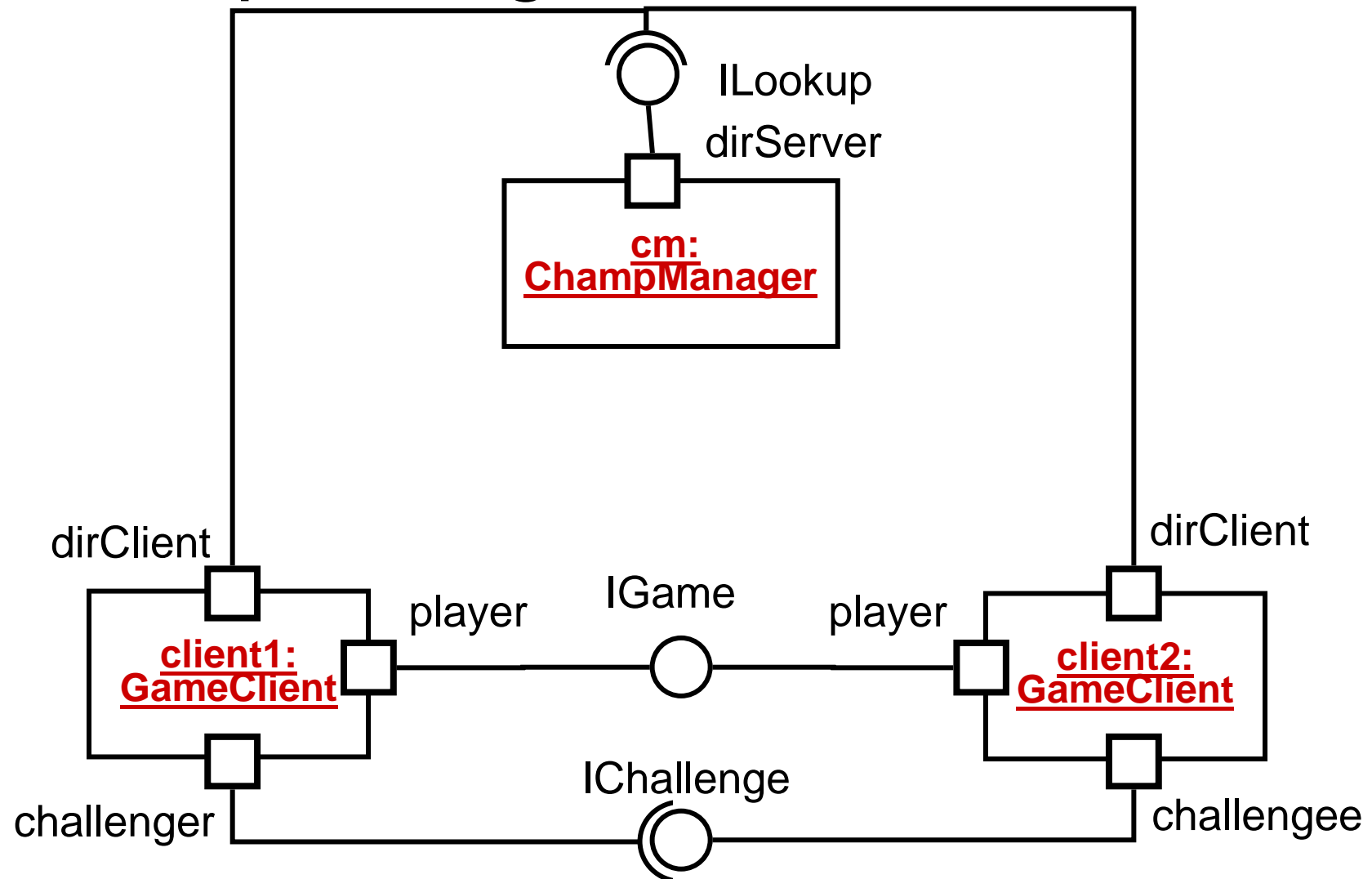


# New Architecture with Champs Manager

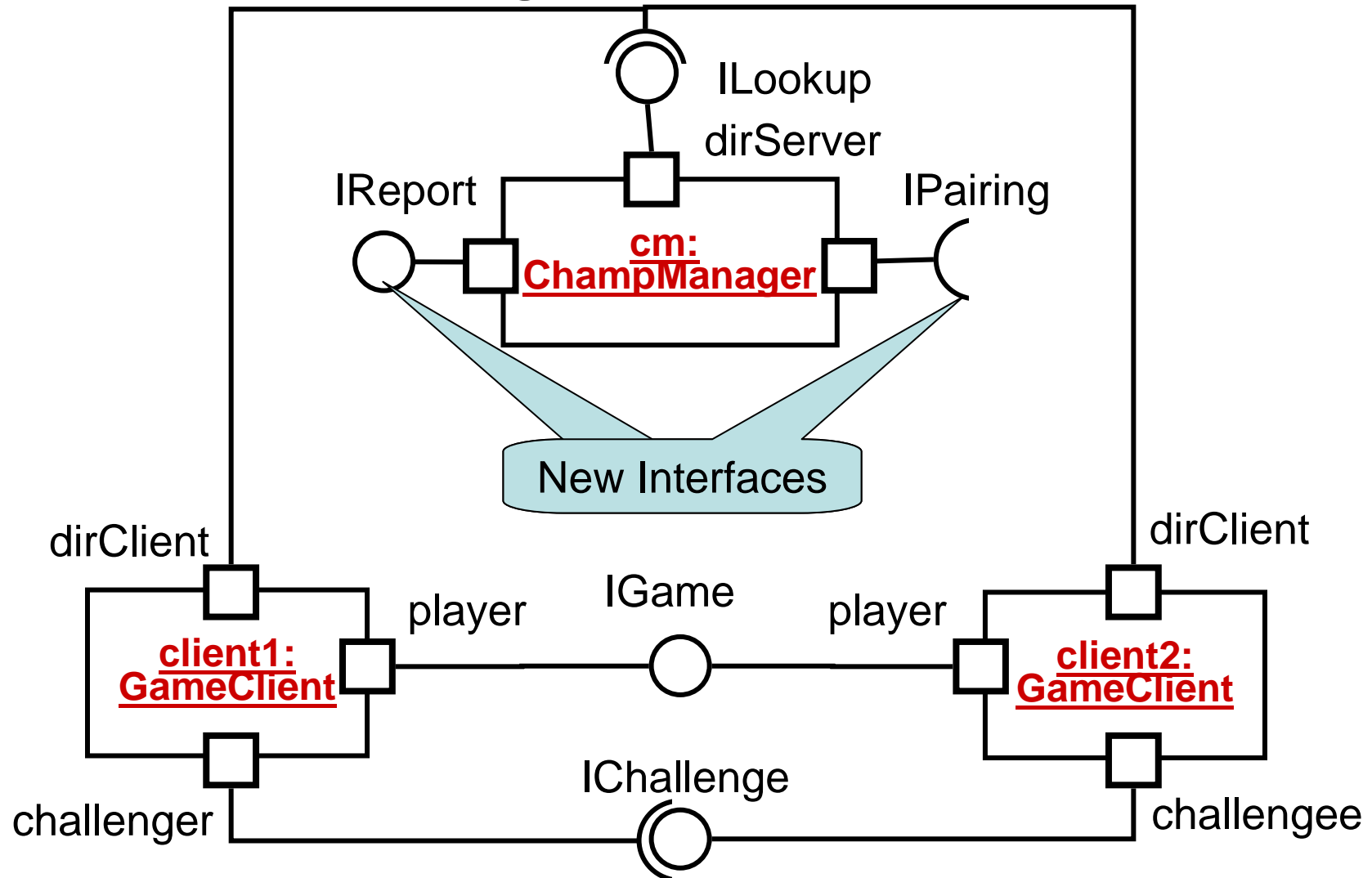
# New Architecture: ChampManager and Game Clients



# New Architecture: ChampManager and Game Clients



# New Architecture: ChampManager and Game Clients



# New Interfaces

|  |
|--|
| <b>&lt;&lt;interface&gt;&gt;</b><br><b><i>IPairing</i></b> |
| createMatch(Client whi, Client bla): Match                 |

|   |
|---|
| <b>&lt;&lt;interface&gt;&gt;</b><br><b><i>IResult</i></b> |
| reportResult(Client cli, Result res)                      |

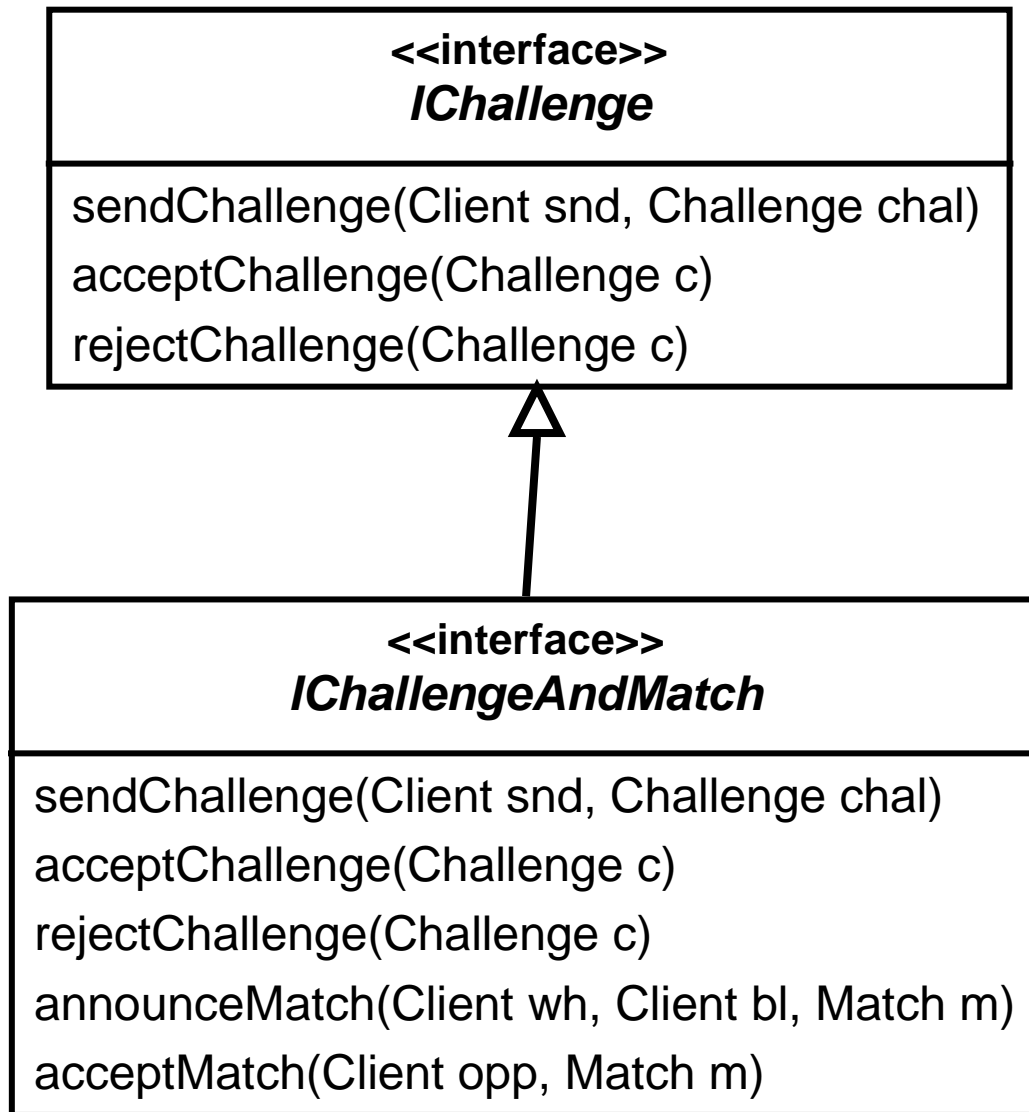
**Question:** How to connect these new interfaces to the existing system?

**Constraint:** Existing modules cannot be changed (e.g. source code is lost)

- **createMatch()**  
a new match is scheduled between the two clients
  - Should be accepted by both clients
- **reportResult()**  
the final result is reported by a client
  - Should be reported by both clients

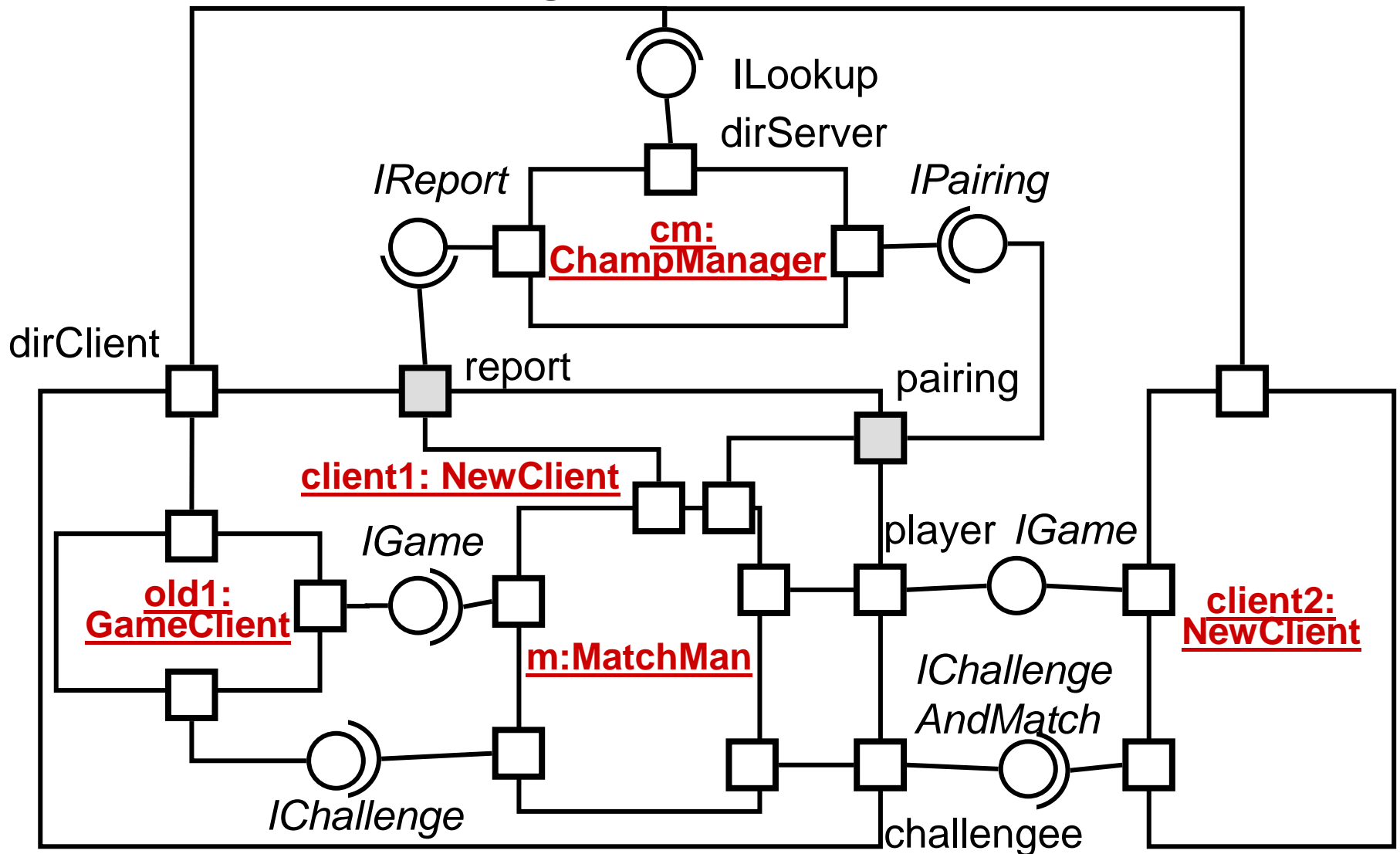
**Solution:** New client program with embedded legacy components

# Changes in the protocol

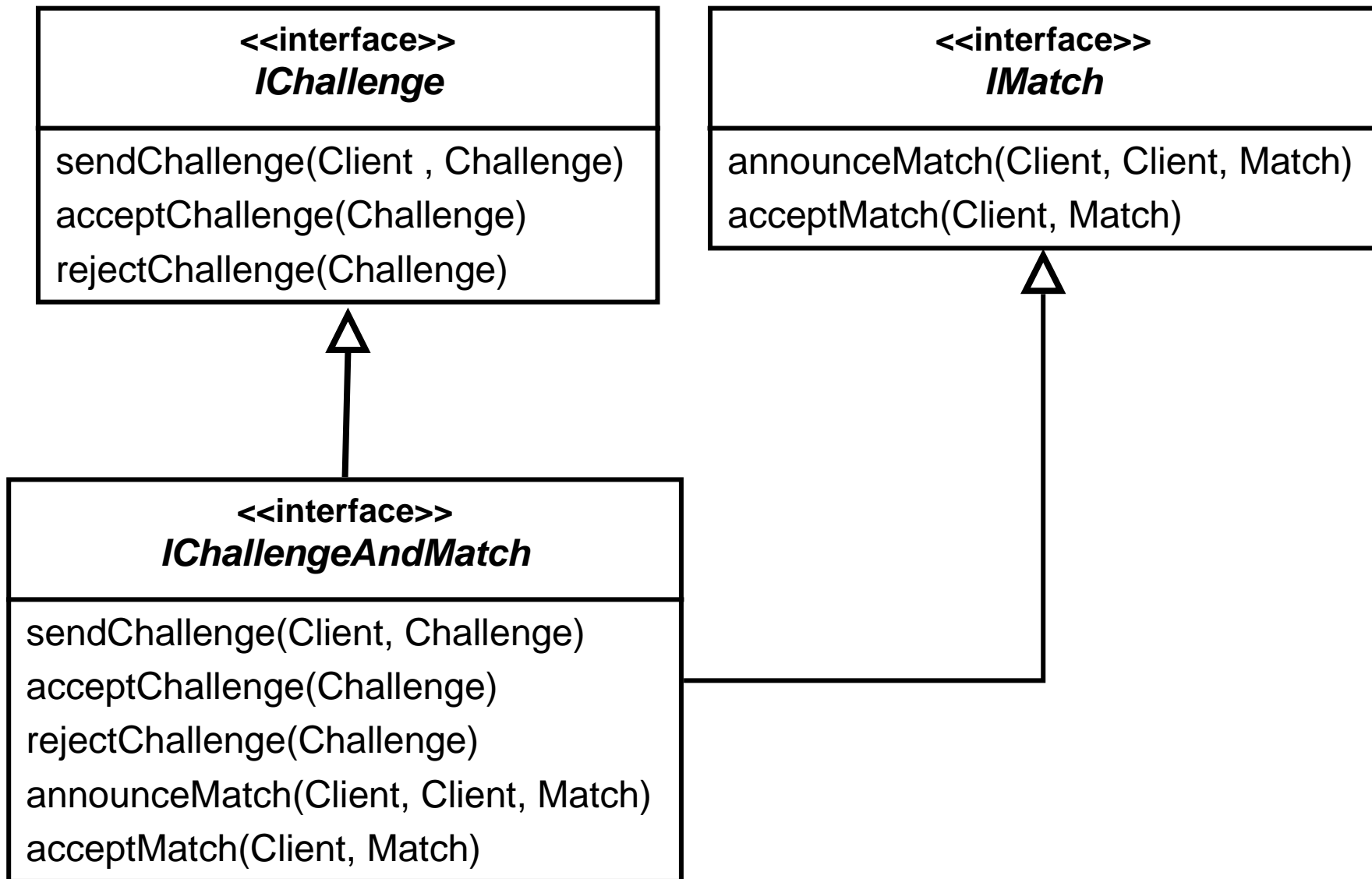


- Proposal: `createMatch()`
  - Requires new functionality for *IChallenge*
  - Derive a new interface *IChallengeAndMatch* by generalization from *IChallenge*
  - Delegate calls to proper ports
- Proposal: `reportResult()`
  - Intercept calls of `endResult()` and also call `reportResult()` inside
  - Delegate all other calls to old ports

# New Architecture: ChampManager and Game Clients



# Another Solution







# Process Structure: Activities and their Dependencies

