

系统设计报告

一、需求分析

1. 需求描述

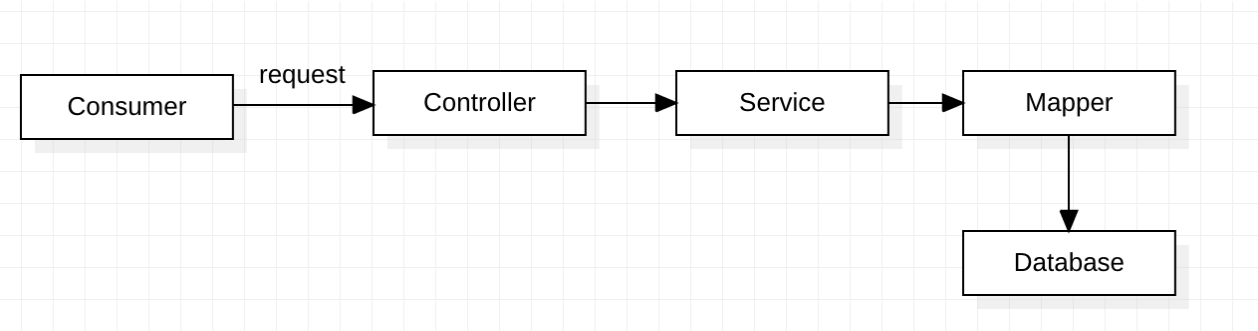
实现一个小型DOTA2游戏信息的查询系统，用户可修改或查询知名玩家，战队，比赛，地区等游戏的相关信息。需要在网页上实现数据的增删改查功能。

2. 数据流图

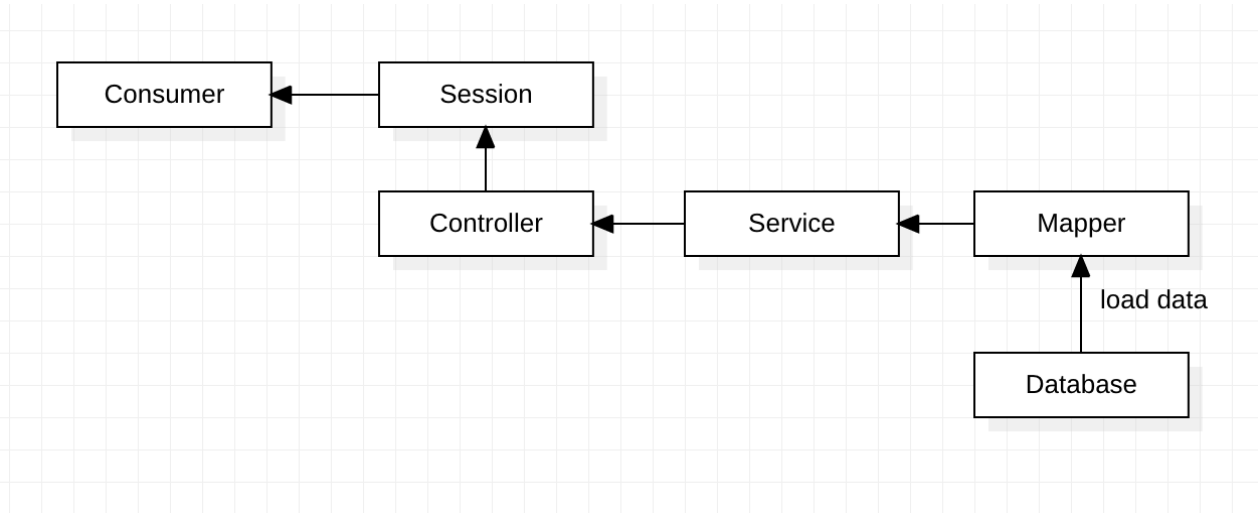
它从数据传递和加工角度，以图形方式来表达系统的逻辑功能、数据在系统内部的逻辑流向和逻辑变换过程，是结构化系统分析方法的主要表达工具及用于表示软件模型的一种图示方法。

整个过程基本包含两个过程，即用户可以向服务器发送请求，服务器可以向用户返回信息。整个过程通过三个大的模块，即Controller, Service, Mapper，服务器。

用户向服务器发送请求

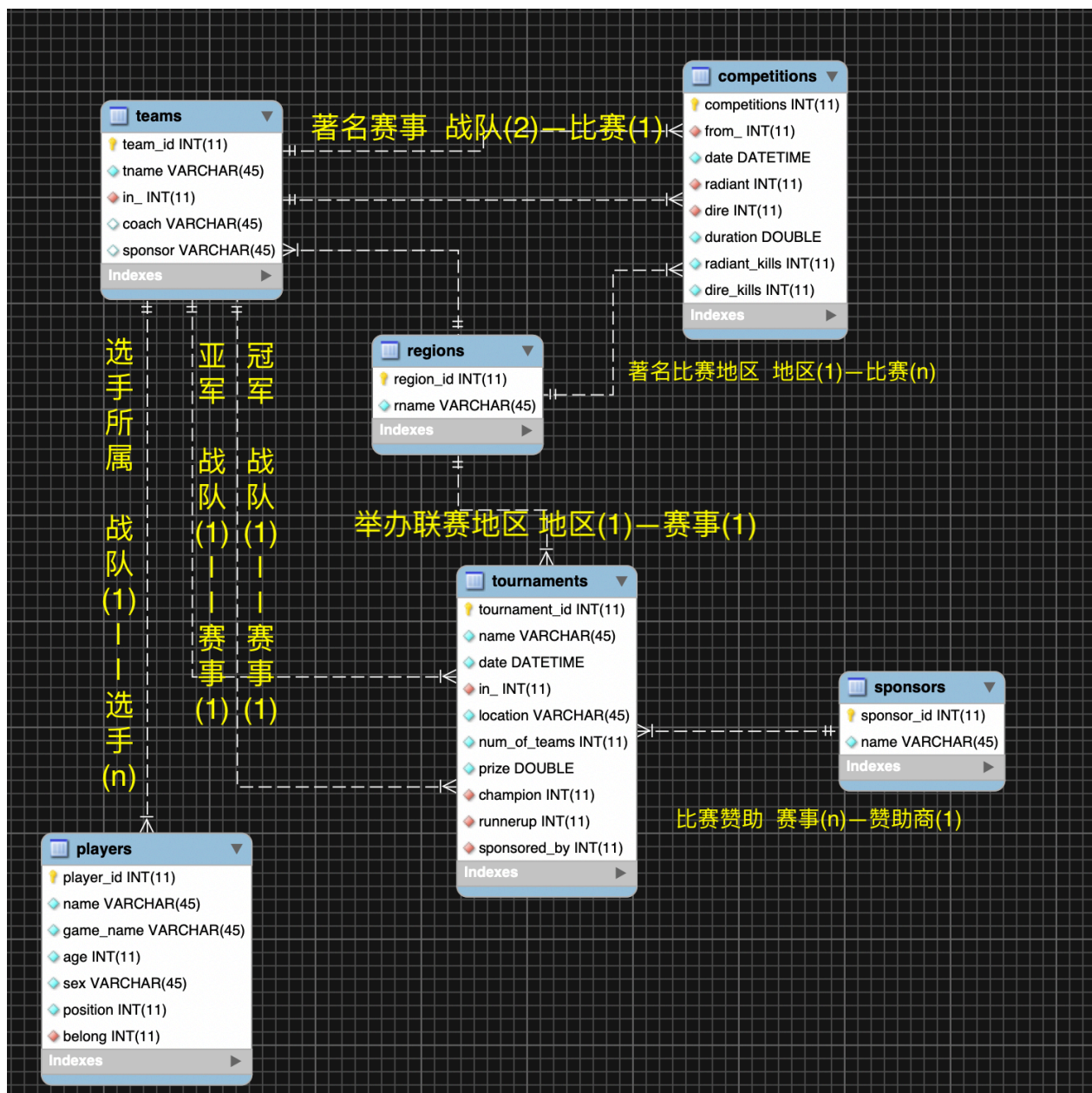


服务器向用户返回信息



二、数据库概念模式设计

E-R图



三、数据库逻辑模式设计

1. 数据库关系模式

teams

属性

teams(team_id, tname, in_, coach, sponsor)

属性域的映像集合

dom(team_id) = number

dom(tname) = string

dom(in_) = region

$\text{dom}(\text{coach}) = \text{players}$

$\text{dom}(\text{sponsor}) = \text{sponsor}$

依赖关系集合

$F = \{\text{team_id} \rightarrow \text{tname}, \text{team_id} \rightarrow \text{in_}, \text{team_id} \rightarrow \text{coach}, \text{team_id} \rightarrow \text{sponsor}\}$

regions

属性

$\text{regions}(\text{region_id}, \text{rname})$

属性域的映像集合

$\text{dom}(\text{region_id}) = \text{number}$

$\text{dom}(\text{rname}) = \text{string}$

依赖关系集合

$F = \{\text{region_id} \rightarrow \text{rname}\}$

competitions

属性

$\text{competitions}(\text{competitions}, \text{from_}, \text{date}, \text{radiant}, \text{dire}, \text{duration}, \text{radiant_kills}, \text{dire_kills})$

属性域的映像集合

$\text{dom}(\text{competitions}) = \text{number}$

$\text{dom}(\text{from_}) = \text{region}$

$\text{dom}(\text{date}) = \text{time}$

$\text{dom}(\text{radiant}) = \text{teams}$

$\text{dom}(\text{dire}) = \text{teams}$

$\text{dom}(\text{duration}) = \text{time}$

$\text{dom}(\text{radiant_kills}) = \text{number}$

$\text{dom}(\text{dire_kills}) = \text{number}$

依赖关系集合

$F = \{\text{competitions} \rightarrow \text{from_}, \text{competitions} \rightarrow \text{date}, \text{competitions} \rightarrow \text{radiant}, \text{competitions} \rightarrow \text{dire}, \text{competitions} \rightarrow \text{duration}, \text{competitions} \rightarrow \text{radiant_kills}, \text{competitions} \rightarrow \text{dire_kills}\}$

players

属性

players(player_id, name, game_name, age, sex, position, belong)

属性域的映像集合

dom(player_id) = number

dom(name) = string

dom(game_name) = string

dom(age) = number

dom(sex) = sex

dom(position) = number

dom(belong) = teams

依赖关系集合

$F = \{ \text{player_id} \rightarrow \text{name}, \text{player_id} \rightarrow \text{game_name}, \text{player_id} \rightarrow \text{age}, \text{player_id} \rightarrow \text{sex}, \text{player_id} \rightarrow \text{position}, \text{player_id} \rightarrow \text{belong} \}$

tournaments

属性

tournaments(tournament_id, name, date, in_, location, num_of_teams, prize, champion, runnerup, sponsored_by)

属性域的映像集合

dom(tournament_id) = number

dom(name) = string

dom(date) = time

dom(in_) = region

dom(location) = string

dom(num_of_teams) = number

dom(prize) = number

dom(champion) = teams

dom(runnerup) = teams

依赖关系集合

$F = \{\text{tournament_id} \rightarrow \text{name}, \text{tournament_id} \rightarrow \text{date}, \text{tournament_id} \rightarrow \text{in_}, \text{tournament_id} \rightarrow \text{num_of_teams}, \text{tournament_id} \rightarrow \text{prize}, \text{tournament_id} \rightarrow \text{champion}, \text{tournament_id} \rightarrow \text{runnerup}, \text{tournament_id} \rightarrow \text{sponsored_by}\}$

sponsors

属性

sponsors(sponsor_id, name)

属性域的映像集合

$\text{dom}(\text{sponsor_id}) = \text{number}$

$\text{dom}(\text{name}) = \text{string}$

依赖关系集合

$F = \{\text{sponser_id} \rightarrow \text{name}\}$

2. 关系模式范式等级的判定与规范化

所有关系模式都由主键唯一决定，都符合3NF范式。

teams -- 3NF

regions -- 3NF

sponsors -- 3NF

players -- 3NF

tournaments -- 3NF

competitions -- 3NF