

게임 서버 프로그래밍 교과서

GAME SERVER
PROGRAMMING
TEXTBOOK

배현직 지음

〈몬스터 길들이기〉, 〈스트리트파이터5〉 등에서 사용하는
서버&네트워크 엔진 프라우드넷 개발자가 알려주는
게임 서버 개발의 핵심 원리와 노하우!

알고리즘부터 아키텍처까지!

길벗

게임 서버 프로그래밍 교과서

Game Server Programming Textbook

초판 발행 · 2019년 4월 8일

지은이 · 배현직

발행인 · 이종원

발행처 · (주)도서출판 길벗

출판사 등록일 · 1990년 12월 24일

주소 · 서울시 마포구 월드컵로 10길 56(서교동)

대표 전화 · 02)332-0931 | 팩스 · 02)323-0586

홈페이지 · www.gilbut.co.kr | 이메일 · gilbut@gilbut.co.kr

기획 및 책임편집 · 김윤지(yunjikim@gilbut.co.kr) | 디자인 · 장기춘 | 제작 · 이준호, 손일순, 이진혁

영업마케팅 · 임태호, 전선하, 지운집, 박성용 | 영업관리 · 김명자 | 독자지원 · 송혜란, 정은주

교정교열 · 김윤지 | 전산편집 · 도설아 | 출력 · 인쇄 · 제본 · (주)벽호

- ▶ 잘못된 책은 구입한 서점에서 바꿔 드립니다.
- ▶ 이 책에 실린 모든 내용, 디자인, 이미지, 편집 구성의 저작권은 (주)도서출판 길벗과 지은이에게 있습니다. 허락 없이 복제하거나 다른 매체에 옮겨 실을 수 없습니다.

ISBN 979-11-6050-747-8 93000

(길벗 도서번호 006884)

정가 35,000원

독자의 1초를 아껴주는 정성 길벗출판사

(주)도서출판 길벗 | www.gilbut.co.kr

페이스북 · www.facebook.com/gbitbook

1장 멀티스레딩 …… 017

- 1.1 프로그램과 프로세스 018**
- 1.2 스레드 020**
- 1.3 멀티스레드 프로그래밍은 언제 해야 할까? 027**
 - 1.3.1 오래 걸리는 일 하나와 빨리 끝나는 일 여럿을 같이 해야 할 때 028
 - 1.3.2 어떤 긴 처리를 진행하는 동안 다른 짧은 일을 처리해야 할 때 030
 - 1.3.3 기기에 있는 CPU를 모두 활용해야 할 때 031
- 1.4 스레드 정체 038**
- 1.5 스레드를 다룰 때 주의 사항 042**
- 1.6 임계 영역과 뮤텍스 045**
- 1.7 교착 상태 054**
- 1.8 잠금 순서의 규칙 062**
- 1.9 병렬성과 시리얼 병목 065**
- 1.10 싱글스레드 게임 서버 071**
- 1.11 멀티스레드 게임 서버 072**
- 1.12 스레드 풀링 074**
- 1.13 이벤트 077**
- 1.14 세마포어 080**
 - 1.14.1 세마포어의 또 다른 용도 081
- 1.15 원자 조작 085**
- 1.16 멀티스레드 프로그래밍의 흔한 실수들 087**
 - 1.16.1 읽기와 쓰기 모두에 잠금하지 않기 087
 - 1.16.2 잠금 순서 꼬임 088
 - 1.16.3 너무 좁은 잠금 범위 089
 - 1.16.4 디바이스 타임이 섞인 잠금 090
 - 1.16.5 잠금의 전염성으로 발생한 실수 091
 - 1.16.6 잠금된 뮤텍스나 임계 영역 삭제 092
 - 1.16.7 일관성 규칙 깨기 093
- 1.17 심화 내용 및 더 읽을거리 095**

2장 컴퓨터 네트워크 …… 097

2.1 컴퓨터 네트워크를 구성하는 기기 098

2.1.1 OSI 모델 100

2.1.2 OSI 모델의 계층 2 101

2.1.3 OSI 모델의 계층 3 104

2.2 인터넷 106

2.3 컴퓨터 네트워크 데이터 107

2.3.1 스트림 형식 107

2.3.2 메시지 형식 109

2.4 컴퓨터 네트워크 식별자 112

2.5 컴퓨터 네트워크의 품질과 특성 116

2.5.1 네트워크의 품질을 저해하는 것들 116

2.5.2 전송 속도와 전송 지연 시간 119

2.5.3 네트워크 품질 기준 세 가지 122

2.5.4 무선 네트워크의 품질 123

2.6 컴퓨터 네트워크에서 데이터 보내기와 받기 124

2.6.1 UDP 네트워킹 125

2.6.2 TCP 네트워킹 128

2.7 패킷 유실 시 UDP와 TCP에서 현상 133

2.8 주로 사용하는 메시지 형식 135

2.9 네트워크 주소 변환 137

2.10 요약 140

2.11 더 읽을거리 141

3장 소켓 프로그래밍 145

- 3.1 블로킹 소켓 147
- 3.2 네트워크 연결 및 송신 148
- 3.3 블로킹과 소켓 버퍼 149
- 3.4 네트워크 연결받기 및 수신 154
- 3.5 수신 버퍼가 가득 차면 발생하는 현상 156
- 3.6 논블록 소켓 157
- 3.7 Overlapped I/O 혹은 비동기 I/O 167
- 3.8 epoll 175
- 3.9 IOCP 179
- 3.10 더 읽을거리 185

4장 게임 서버와 클라이언트 187

- 4.1 패키지 게임에서 게임 서버 188
- 4.2 온라인 게임에서 게임 서버 190
- 4.3 서버의 역할 191
- 4.4 게임 클라이언트와 서버의 상호 작용 192
- 4.5 게임 서버가 하는 일 194
- 4.6 게임 서버의 품질 197
 - 4.6.1 안정성 197
 - 4.6.2 확장성 201
 - 4.6.3 성능 203
 - 4.6.4 관리 편의성 210
- 4.7 플레이어 정보의 저장 212
- 4.8 서버 구동 환경 216
- 4.9 서버 개발 지침 221
- 4.10 더 읽을거리 222

5장 게임 네트워킹 223

- 5.1 UML 224
 - 5.1.1 UML 시퀀스 다이어그램 225
- 5.2 게임 플레이 네트워킹 227
 - 5.2.1 모든 역할을 서버에서 하기 227
 - 5.2.2 렌더링은 클라이언트에서 하기 229
 - 5.2.3 추측항법 234
- 5.3 레이턴시 마스킹 237
- 5.4 넓은 월드, 많은 캐릭터 처리 241
- 5.5 실시간 전략 시뮬레이션 게임에서 네트워크 동기화 243
- 5.6 실제 레이턴시 줄이기 246
- 5.7 게임 플레이 이외의 네트워킹 247
- 5.8 해킹과 보안 252
 - 5.8.1 네트워크 해킹 253
 - 5.8.2 클라이언트 컴퓨터 해킹 256
 - 5.8.3 서버 컴퓨터 해킹 256
 - 5.8.4 게임 치트 257
- 5.9 요약 258

6장 게임 네트워크 엔진 프라우드넷 259

- 6.1 게임 서버, 네트워크 엔진 260
- 6.2 개발 환경과 기본 모듈 261
- 6.3 게임 클라이언트-서버 간 통신 261
- 6.4 메시지 주고받기 267
- 6.5 와이파이 셀룰러 연결 핸드오버 기능 269
- 6.6 원격 메서드 호출 271

- 6.7 클라이언트끼리 P2P 통신 279
- 6.8 예시: 채팅 처리 284
- 6.9 스레드 모델 289
- 6.10 더 읽을거리 291

7장 데이터베이스 기초 293

- 7.1 플레이어의 정보 저장 294
- 7.2 데이터베이스 사용 297
- 7.3 데이터베이스의 데이터 구성 299
- 7.4 데이터베이스 시작 300
- 7.5 SQL 질의 구문 303
- 7.6 인덱스와 키 304
- 7.7 플레이어 정보를 데이터베이스에 저장하는 방법 1 307
- 7.8 플레이어 정보를 데이터베이스에 저장하는 방법 2 310
- 7.9 질의 구문 실행 314
 - 7.9.1 트랜잭션 317
- 7.10 게임 서버에서 질의 구문 실행 323
- 7.11 보안을 위한 주의 사항 326
- 7.12 더 읽을거리 328

8장 NoSQL 기초 331

- 8.1 관계형 데이터베이스와 NoSQL 332
- 8.2 관계형 데이터베이스에서 확장성 334
- 8.3 관계형 데이터베이스에서 고가용성 338

- 8.4 MongoDB를 위한 JSON 이해 342
- 8.5 MongoDB 시작 344
- 8.6 MongoDB에 데이터 액세스 348
 - 8.6.1 생성(create) 349
 - 8.6.2 읽기(read) 349
 - 8.6.3 업데이트(update) 354
 - 8.6.4 지우기(delete) 355
- 8.7 성능 분석 기능 356
- 8.8 MongoDB 수평 확장 357
- 8.9 게임 서버에서 MongoDB 명령 실행 360
- 8.10 요약 및 더 알아보기 361

9장 분산 서버 구조 363

- 9.1 수직 확장과 수평 확장 364
- 9.2 서버 분산이 없다면? 365
- 9.3 고전적인 서버 분산 방법 369
- 9.4 논리적 단일 서버 분산 372
- 9.5 데이터 분산 vs 기능적 분산 375
- 9.6 로직 처리의 분산 방식들 377
 - 9.6.1 동기 분산 처리 378
 - 9.6.2 비동기 분산 처리 383
 - 9.6.3 데이터 복제에 기반을 둔 로컬 처리 386
- 9.7 데이터 응집도 388
- 9.8 기능적 분산 처리 392
- 9.9 분산 처리를 염선해야 하는 이유 394
- 9.10 분산 처리 전략 395
- 9.11 분산 서버의 또 다른 장점 395

- 9.12 고가용성 396
- 9.13 데이터베이스의 분산 401
- 9.14 요약 402

10장 분산 서버 구조 사례 405

- 10.1 로그인 처리의 분산 406
- 10.2 데이터베이스의 수평 확장 410
- 10.3 매치메이킹의 분산 처리 421
- 10.4 몬스터 NPC 처리의 분산 처리 433
- 10.5 플레이어 간 상호 작용 분산 처리 437
- 10.6 로그 및 통계 분석의 분산 처리 438
- 10.7 게임 장르별 분산 서버 형태 442
- 10.8 요약 및 결론 449

찾아보기 450