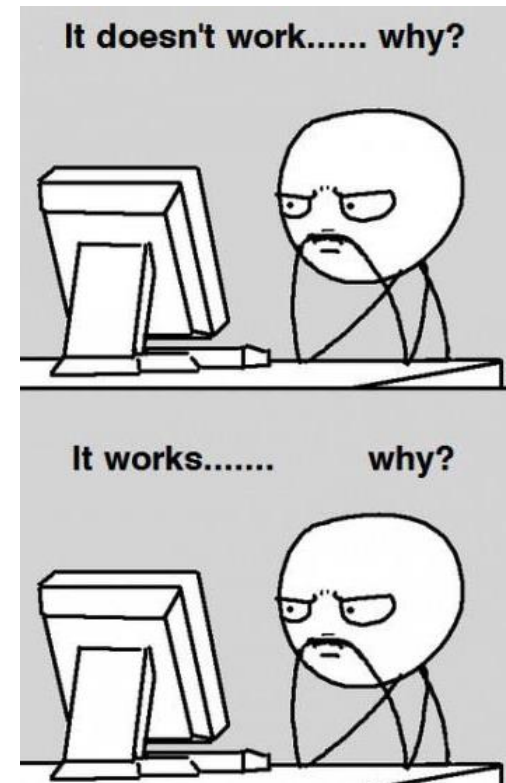


Final Project (Phase 3)

Kyu-haeng Lee
Fall, 2018



Phase 3. 요약

■ 요구사항

- 1) customer와 manager 프로세스를 분리하고 (서로 다른 두 프로그램)
- 2) 각각이 제공하는 함수를 FIFO를 통해 호출하도록 변경하시오.
- * 반드시 카페에서 실제로 주문하는 상황을 생각하면서 구현하기 바람
 - ✓ 차별화 기능은 개인(1개), 팀(2개)로 충분하니, IPC없이 다소 비현실적으로 호출된 함수들은 과감히 삭제바람. (필수기능은 삭제 x)
- * P2에서 customer와 manager 역할에 따라 함수 분리가 잘 되었다면... FIFO 지원만 추가하면 돼서 그리 어렵지 않을거라 생각합니다.

■ 제출

- Due Date: 15주차 (분반 없이 모여서 할 예정 - 드디어 종강!)
- 제출물 (Phase 2와 동일)
 - ✓ FIFO를 위한 res 폴더 반드시 추가
- 발표 (선택) (Phase 2와 동일)

Phase 3. 상세 요구사항

■ 1. 프로세스 분리

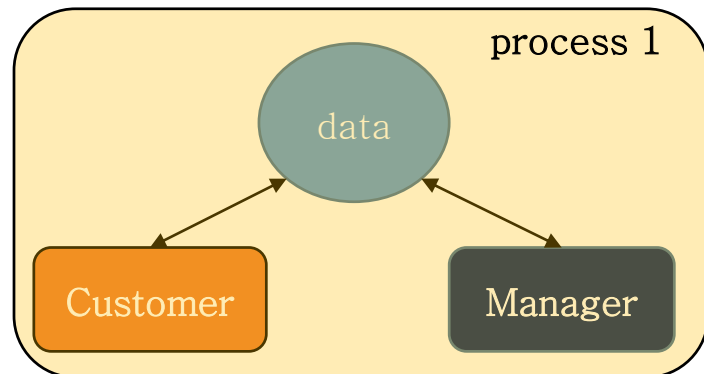
- 관리자용, 고객용 2개의 binary (실행 프로그램)을 만든다.
- 두 프로세스는 FIFO로 연결된다.
- 각 프로세스의 함수 호출에 따라 메시지가 “실시간으로” 전달된다.

● 데이터 접근 방식 차이에 유의

- ✓ 각 프로세스마다 (관리자, 고객마다) 접근할 수 있는 데이터가 다름
- ✓ 상대방 데이터를 요구하기 위해서 “통신”을 해야함.

기존 구조 (phase 2까지)

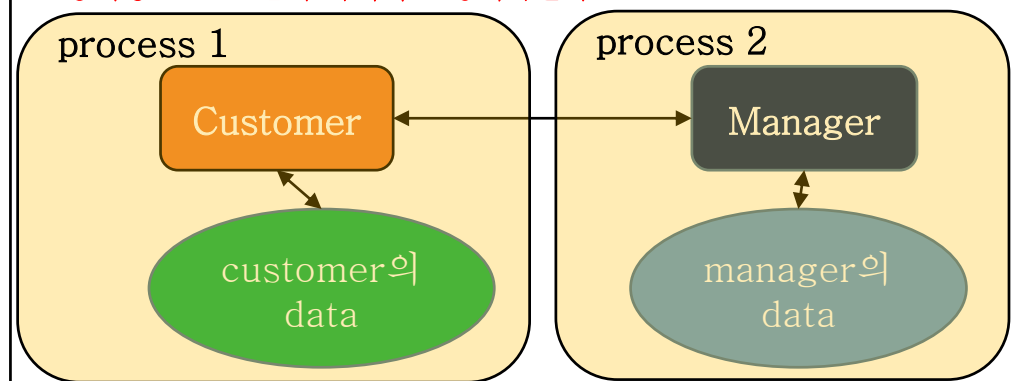
* data 저장소가 분리되어 있지 않다.



변경 구조 (phase 3)

* data가 분리되어 있다.

* 상대방 data를 얻기 위해서 요청해야한다.



Phase 3. 상세 요구사항

■ 1. 프로세스 분리

- 예) customer가 menu를 확인하는 상황
- 이전 버전: 프로세스가 시작되면, 바로 메뉴를 보여준다. (x)
 - ✓ customer가 메뉴판을 항상 들고 다녀야함으로 해석됨.

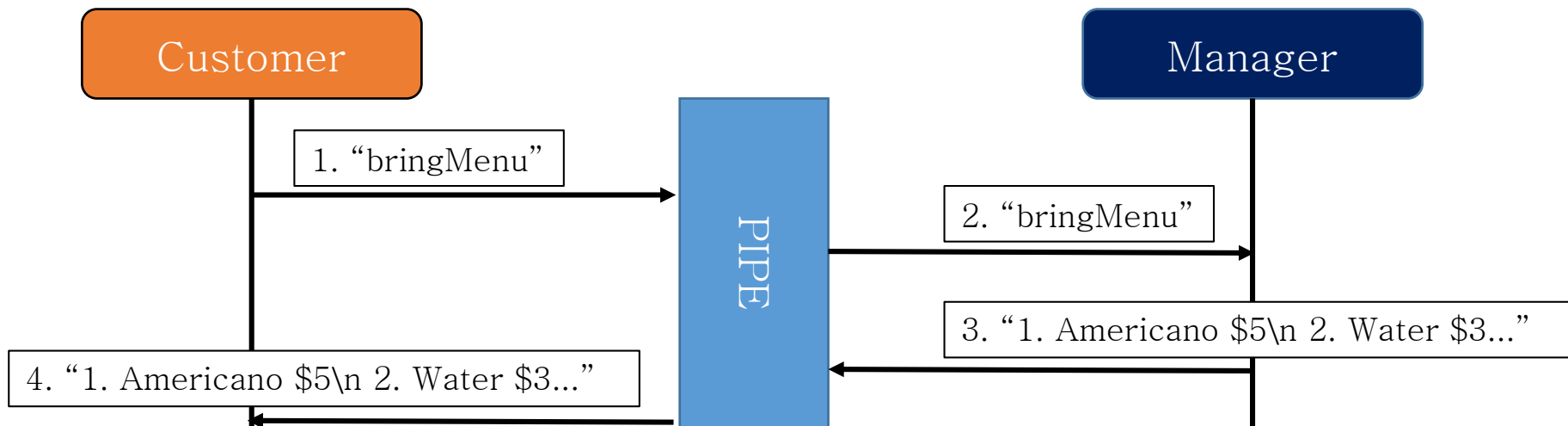
```
khlee@khlee-VirtualBox:~/Downloads/dev(1)/dev/dev$ ./a.out

*****Recommand menu is Espresso
*****menu*****
0.Espresso(2000)
1.Americano(2200/2800)
2.Cafe latte(2500/3100)
3.Cappuccino(2500/3100)
4.Vanilla latte(3000/3600)
5.Secret beverage(2500/3100)
6.exit
```

Phase 3. 상세 요구사항

■ 1. 프로세스 분리

- 예) customer가 menu를 확인하는 상황
- 새로운 버전: customer가 manager에게 메뉴를 보여달라고 요청한다.



- ✓ 1. Customer가 "bringMenu"라는 메시지를 pipe에 쓴다.
- ✓ 2. Manager는 pipe에서 데이터를 읽고, 수신된 메시지가 "bringMenu"임을 확인한다.
- ✓ 3. Manager는 pipe에 menu 데이터를 쓴다.
- ✓ 4. Customer는 pipe에서 데이터를 읽고 menu를 확인한다.

Phase 3. 상세 요구사항

■ 1. 프로세스 분리

- 예) manager가 매출을 확인하는 상황
- 새로운 버전: customer와 프로세스가 분리되어 있으므로 언제든지 확인할 수 있다.

Phase 3. 상세 요구사항

■ 2. FIFO

- FIFO는 res 폴더아래 만들도록 합시다.

■ 4. CMakeLists.txt

- 두개의 바이너리를 만들어야하므로 add_executable()이 두번 필요

■ 5. 팀 참여자 限

- commit 각자 최소 6개 이상 (총 12개 이상)
- customer 최소 2명 지원
 - ✓ 즉, 프로세스가 동시에 3개 해야함.
- 발표할 경우, git clone부터 시작
 - ✓ clone된 것으로만 시연 바람.