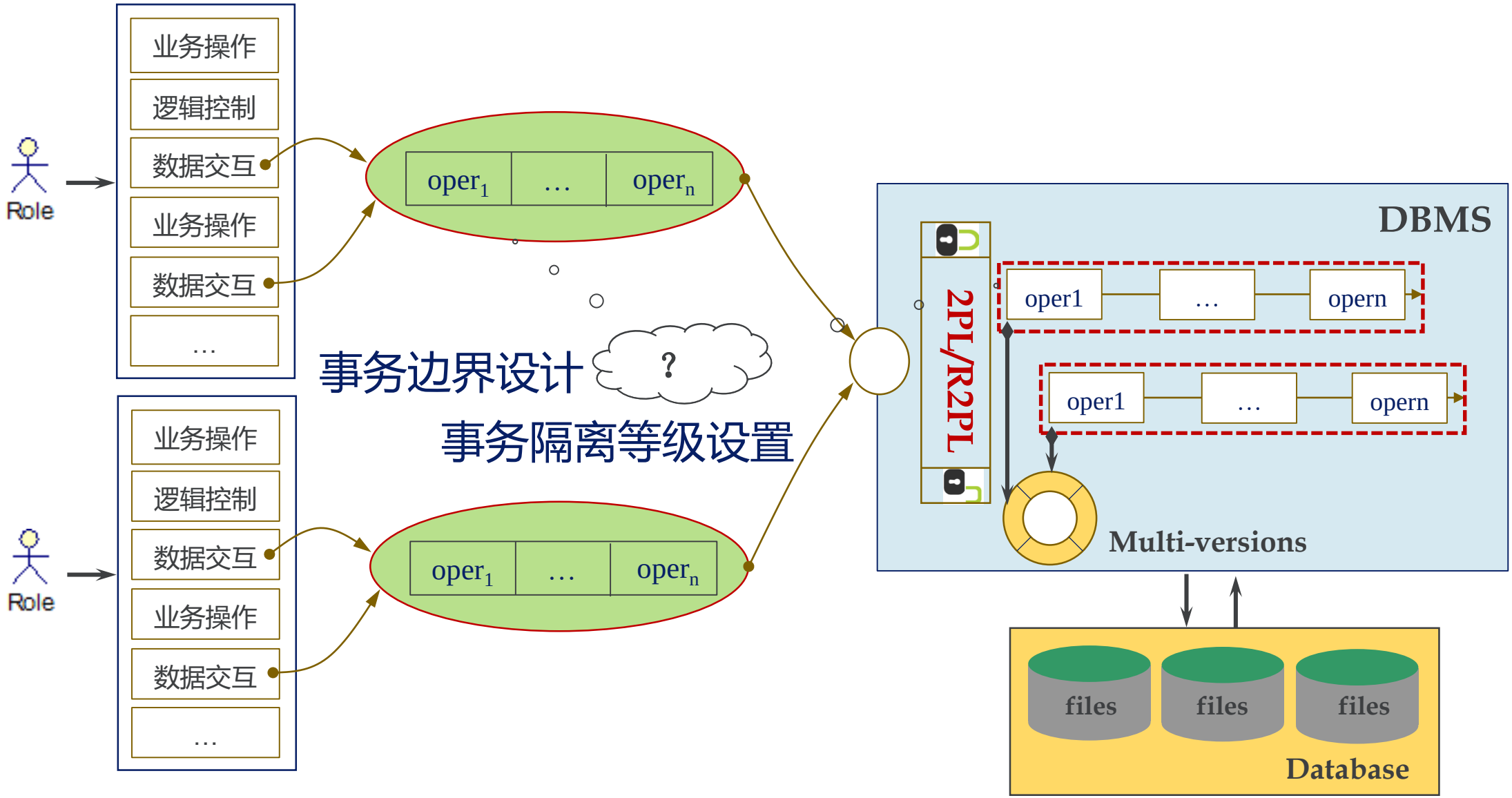
事务设计

讲解人：陆伟 教授

事务设计需要思考的问题

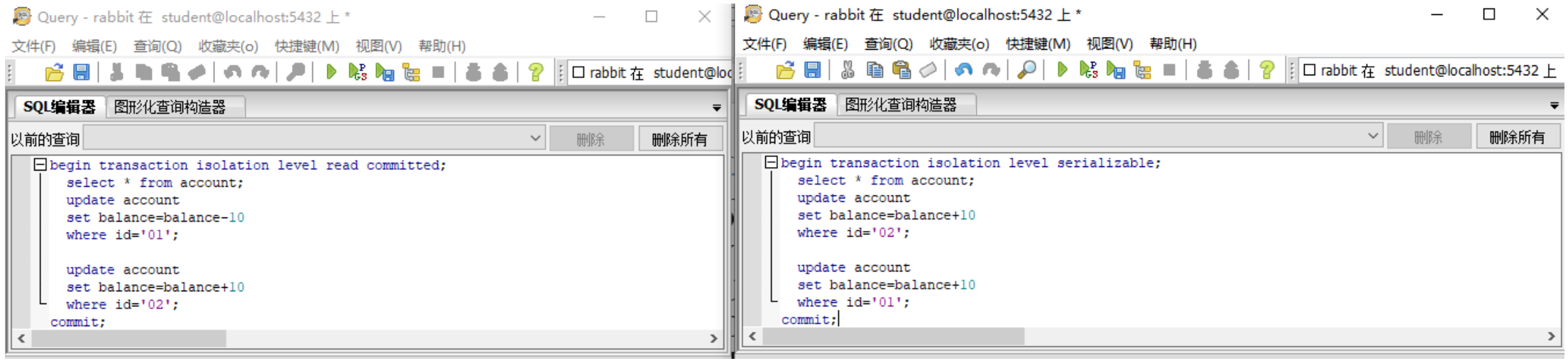


ANSI SQL2 isolation levels

```
SET TRANSACTION  READ ONLY  |  READ WRITE
  [ISOLATION LEVEL
                READ UNCOMMITTED |
                READ COMMIT |
                REPEATABLE READ |
                SERIALIZABLE]
```

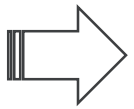
课后实验验证两段锁协议以及不同隔离等级效果

讨论：事务隔离等级设置应遵循什么样的原则?



事务边界设计

1.User logs into the system.
2.DBMS starts a transaction.
3.The system waits for user's choice of action.
4.User chooses to withdraw money from checking.
5.DBMS locks checking account.
6.The system waits for user to enter withdrawal amount.
7.User enters amount.
8.The system issues command to DBMS to update the account balance.
9.The system waits for user to choose new action or quit.
10.User chooses to quit.
11.DBMS commits transaction and unlocks the checking account.



1.User logs into the system.
2.The system waits for user's choice of action.
3.User chooses to withdraw from checking.
4.The system waits for user to enter withdrawal amount.
5.User enters amount.
6.DBMS starts transaction and locks checking account.
7.The system issues command to DBMS to update the account balance.
8.DBMS ends transaction and unlocks the checking account.
9.ATM waits for user to choose new action or quit.
10.User chooses to quit.

讨论：事务边界设计应遵循什么样的原则?

应用程序与事务交互

Consistency of Program Variables

amount = 0
Begin Transaction
For Each Bill
    1)Transfer money from payer to payee (transferring money will deduct from the payer's account balance and increment the payee's account balance).
    2)Increment amount transferred from the payer to the payee.
Commit if success, roll back otherwise
Display the amount deducted from payer's checking account.

关于本讲内容

祝各位学习愉快！

感谢观看！

讲解人：陆伟 教授