

系统设计第二课

即将开始，小憩片刻

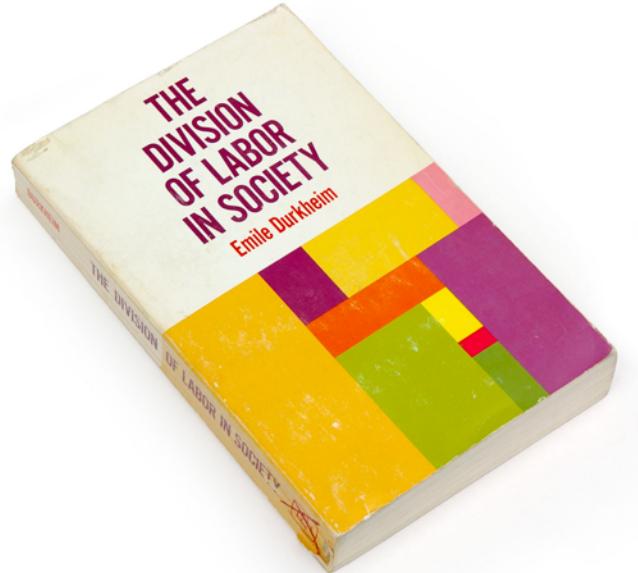


关注微信/微博，获取最新面试题及权威解答

微信: [ninechapter](#)

微博: <http://www.weibo.com/ninechapter>

官网: www.jiuzhang.com



System_Design_2

Database System

张无忌

2015-11-21

V4.00

After class_2, you can answer

- Design data with class and inheritance
 - Google, Alibaba, Facebook, Epic, ...
- Design user system
 - Netflix, Symantec
- Design payment system
 - Yelp, BigCommerce

Q1

Interviewer: design user system

Scenario



- Register/Update/Remove
- Login/Logout
- Balance/Membership



Already have an account?

Phone, email or username

Password

Remember me · [Forgot password?](#)

Log in

New to Twitter?

Sign up now to get your own
personalized timeline!

Sign up

Necessary - Register

- Ask
 - Total users: 100,000,000
 - Daily active users: 1,000,000
- Predict
 - Daily active users in three months = $1,000,000 * 2 = 2,000,000$
 - Register percentage = 1%
 - Daily new register users = $2,000,000 * 1\% = 20,000$

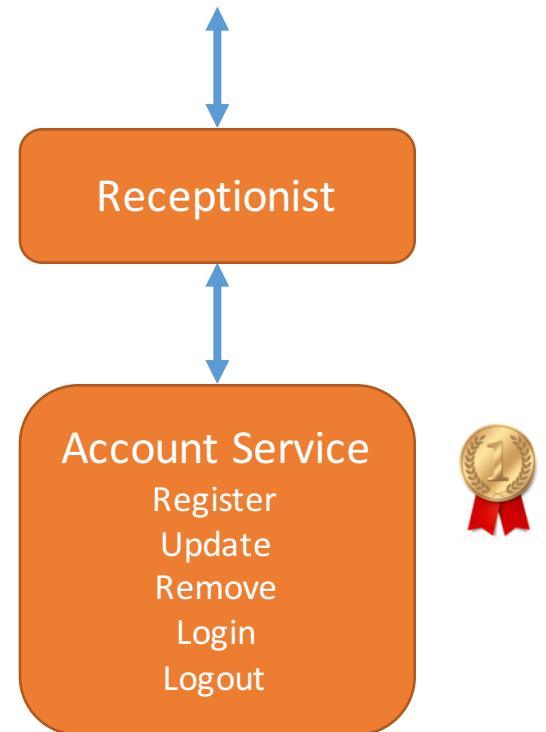


Necessary - Login frequency

- Predict
 - Login percentage = 15%
 - Average login time = 1.2
 - Daily login time = $2,000,000 * 15\% * 1.2 = 360,000$
 - Average login frequency = $360,000 / 24 \text{hour} = 360,000 / 24 / 60 / 60 = 4.2/\text{s}$
 - Normal login frequency = $4.2 * 2 = 8.4/\text{s}$
 - Peak login frequency = $8.4 * 5 = 42/\text{s}$



Application



Data – User (v1)



```
class User{  
    private:  
        int userID; //Primary key  
        string name;  
        string password;  
};
```



[Read More](#)

Novice, <http://url.cn/ZtCS09>

Data – User Table (v1)

```
class UserTable{  
private:  
    vector<User> table;  
public:  
    ... Insert(...);  
    ... Delete(...);  
    ... Update(...);  
    ... Select(...);  
};
```

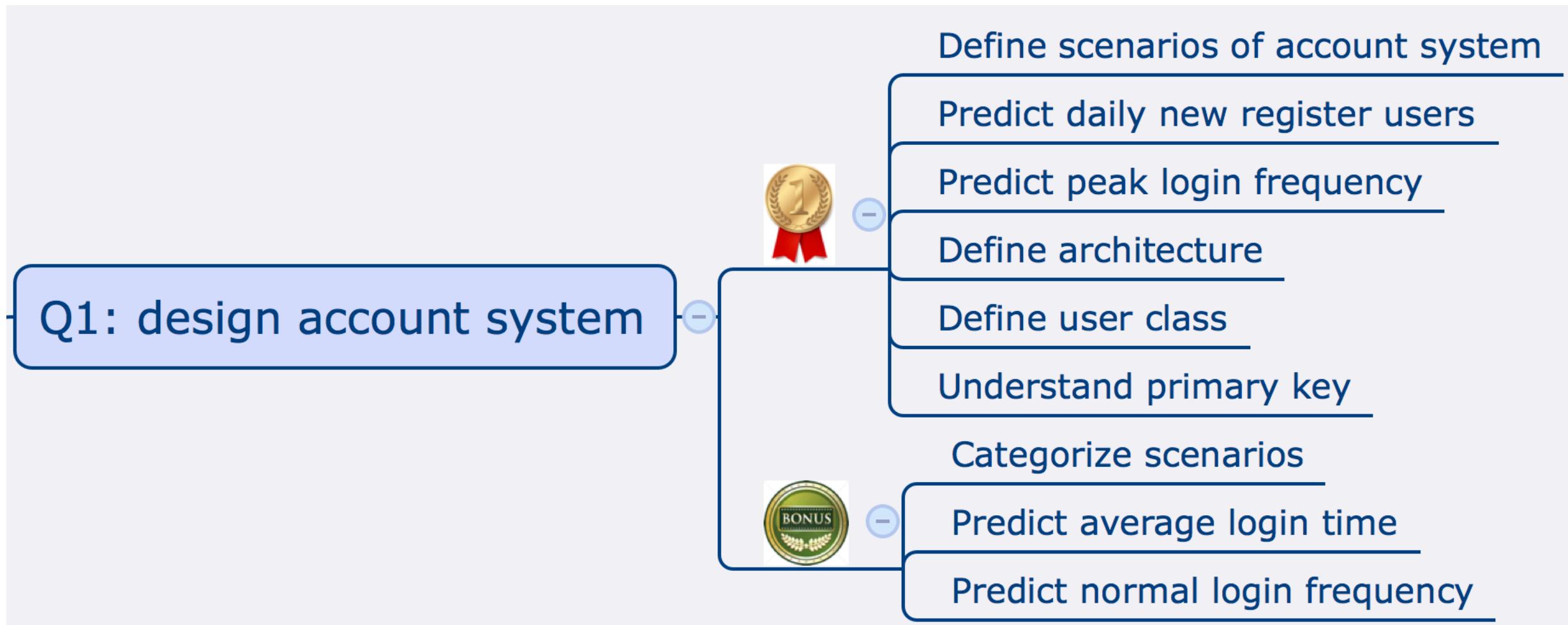


userID	name	password
4	Sangpo	12345
3	Steve	654321
0	Jobs	888888
21	Killer	123456
19	Me	123456

[Read More](#)

Novice, <http://url.cn/ZjEK0m>

Summary of Q1 (6+3)



How to prepare for our class?



1. Look at the question
 2. Answer it by yourself (with Google)
 3. Check your points with our summary
- Add more points as you go deeper with each question

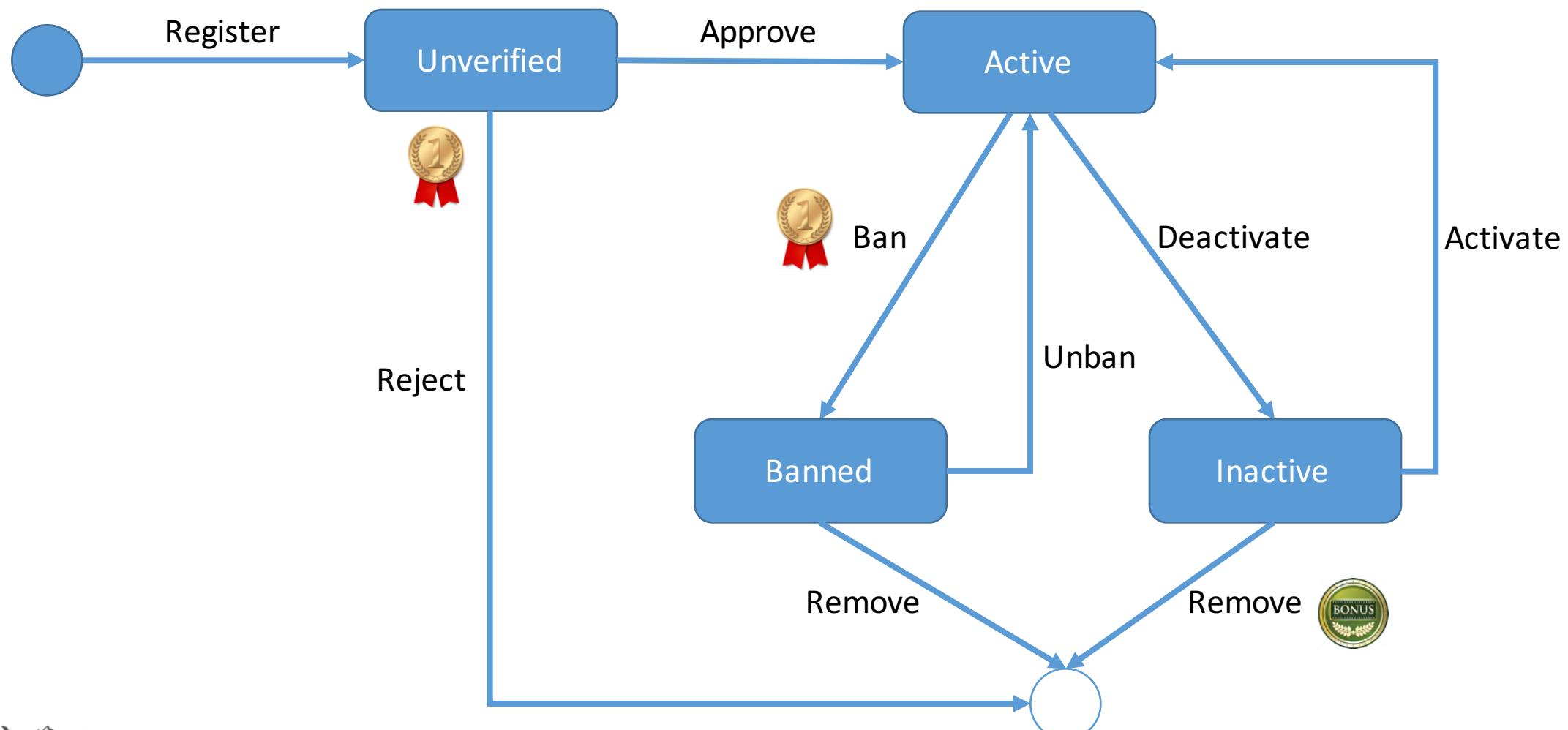
Q2

Interviewer: support user management functions, such as **verification** and **ban**?

State lifecycle graph (v1)



State lifecycle graph (v2)



Data – User (v2)

```
class User{
```

```
    private:
```

```
        int userID;
```



```
        char name[10];
```



```
        char hiddenPassword[10];
```



```
        int state;
```

```
};
```

[Read More](#)

Expert, <http://url.cn/HtwOCR>

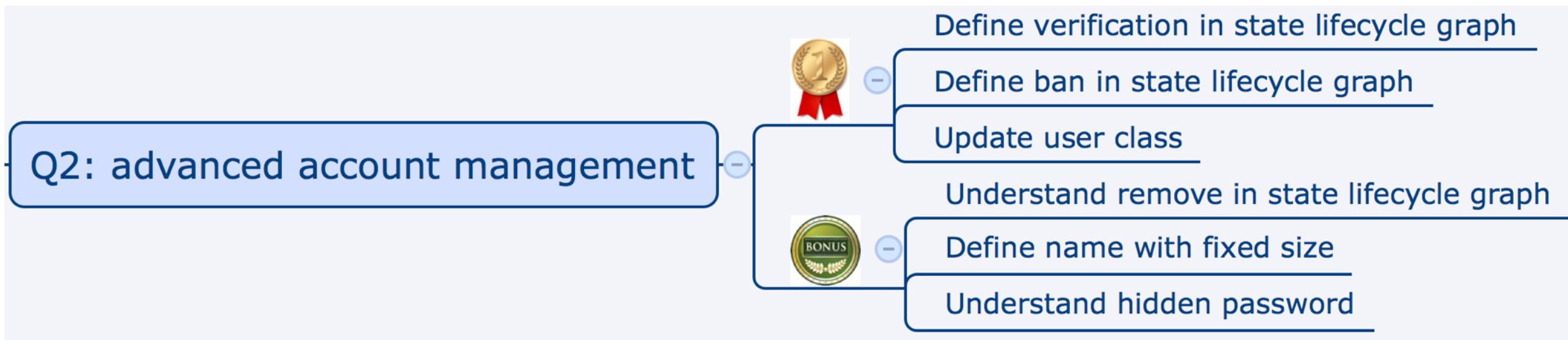
Expert, <http://url.cn/agKqsn>

Data – User Table (v2)

```
class UserTable{  
private:  
    vector<User> table;  
public:  
    ... Insert(...);  
    ... Delete(...);  
    ... Update(...);  
    ... Select(...);  
};
```

userID	name	hiddenPassword	state
4	Sangpo	A1V2F2G2F1	1
3	Steve	F4H1G1H7G1	2
0	Jobs	F1G4J5H1K1	3
21	Killer	M2J2J3N4M1	1
19	Me	M3J1B3N2N1	1

Summary of Q2 (3+3)

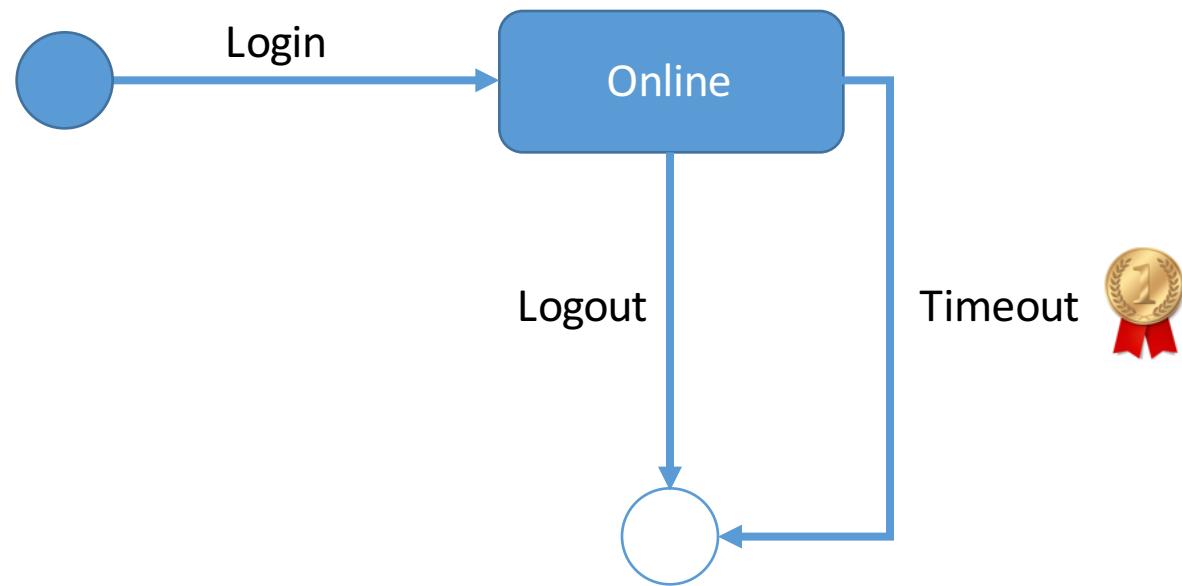


Q3

Interviewer: design the processes

- A user will **logout** automatically after a period of time
- A user can **login** from PC and iPad at the same time

Session lifecycle graph (v1)



Session

- It is the ticket you get when you have ordered the food in KFC
 - Session is a conversation between a user and a server
- You will have two tickets if you order twice in KFC
 - A user will have multiple sessions when he/she logins from different devices at the same time.
- You can only take your food with your ticket number
 - You should at least remember your sessionID

Read More

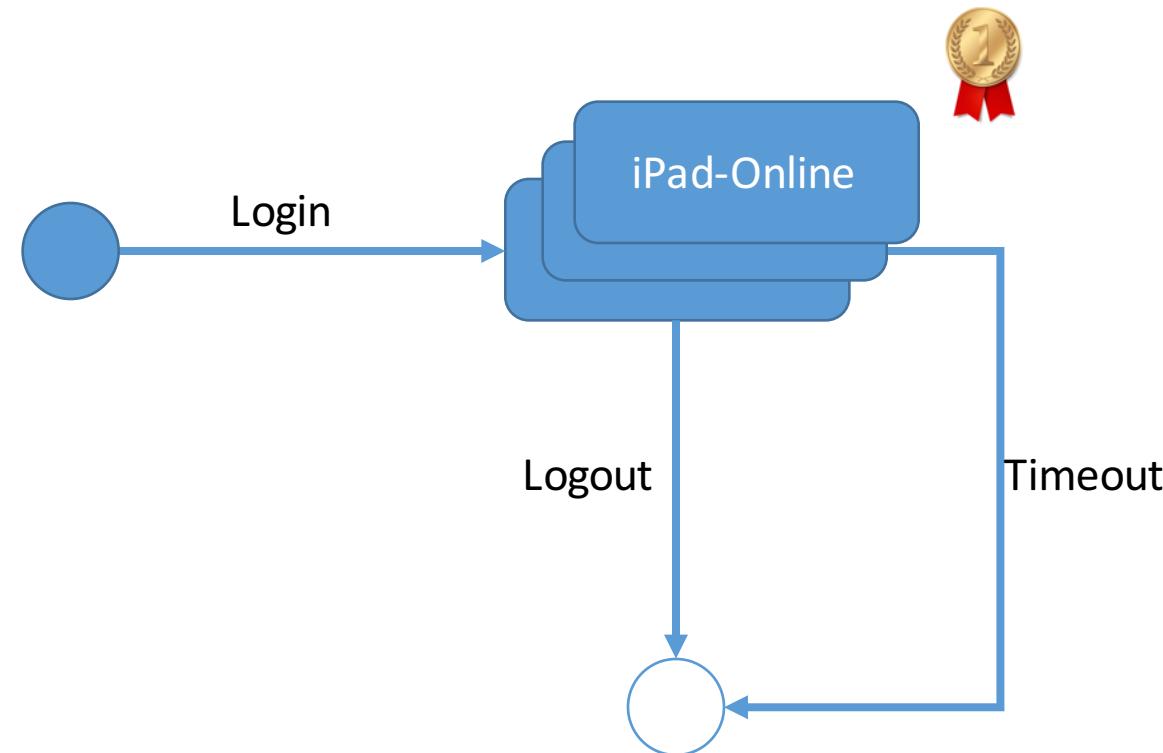
Novice, <http://url.cn/YTXtU6>

Expert, <http://url.cn/g8faDE>

Master, <http://url.cn/LsyovG>

Session lifecycle graph (v2)

- Login from different devices at the same time



Data – User (v3)

```
class User{  
    private:  
        int userID; //Primary Key  
        char name[10];  
        char hiddenPassword[10];  
        int state;  
        vector<Session> sessionList;  
};
```

 class Session{
 private:
 int sessionID;
 int userID;
 int deviceCode; 
 time_t timeOut; 
};

Data – User Table (v3)

```
class UserTable{
    private:
        vector<User> table;
    public:
        ... Insert(...);
        ... Delete(...);
        ... Update(...);
        ... Select(...);
};
```

userID	name	hiddenPassword	state	sessionList
4	Sangpo	A1V2F2G2F1	1	...
3	Steve	F4H1G1H7G1	2	...
0	Jobs	F1G4J5H1K1	3	...
21	Killer	M2J2J3N4M1	1	...
19	Me	M3J1B3N2N1	1	...

Observation

userID	name	hiddenPassword	state	sessionList
4	Sangpo	A1V2F2G2F1	1	...
3	Steve	F4H1G1H7G1	2	...
0	Jobs	F1G4J5H1K1	3	...
21	Killer	M2J2J3N4M1	1	...
19	Me	M3J1B3N2N1	1	...



- We update sessionList more frequently
- The size of sessionList is dynamic

Data – User Table and Session Table (v4)



userID (Primary key)	name	hiddenPassword	state
4	Sangpo	A1V2F2G2F1	1
3	Steve	F4H1G1H7G1	2
0	Jobs	F1G4J5H1K1	3
21	Killer	M2J2J3N4M1	1
19	Me	M3J1B3N2N1	1

sessionID (Primary key)	userID (Foreign key)	deviceCode	timeOut
1010111101	4	1	22:12:13 15 03 2015
1010101010	4	3	10:12:33 16 03 2015
0101010001	4	1	12:22:23 16 03 2015
1101001010	0	1	17:21:05 16 03 2015
1101001001	19	1	12:05:51 16 03 2015
1000101001	19	2	14:33:33 16 03 2015

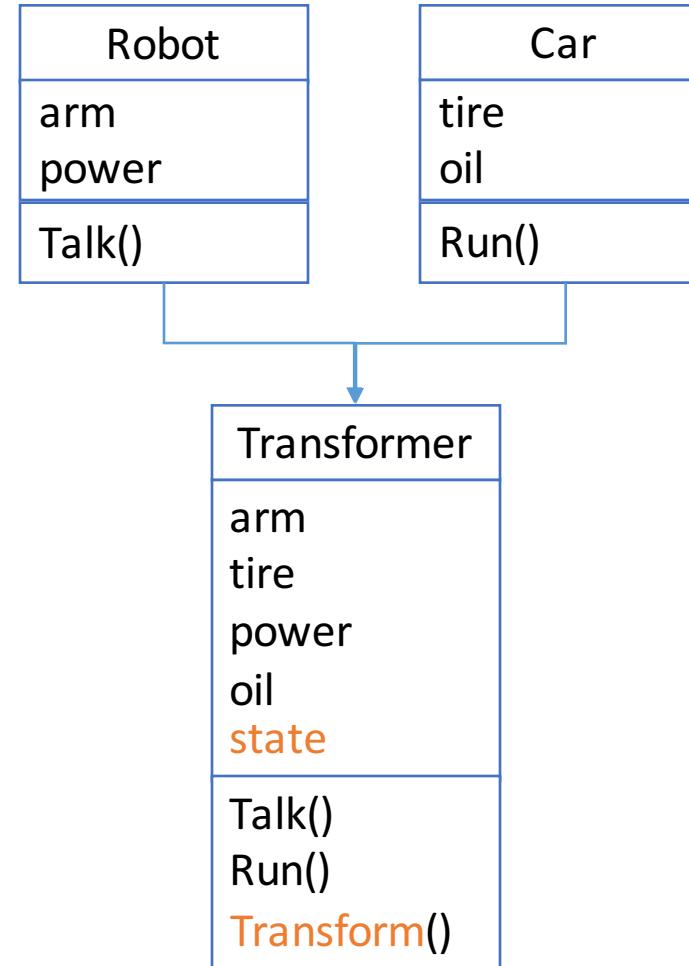
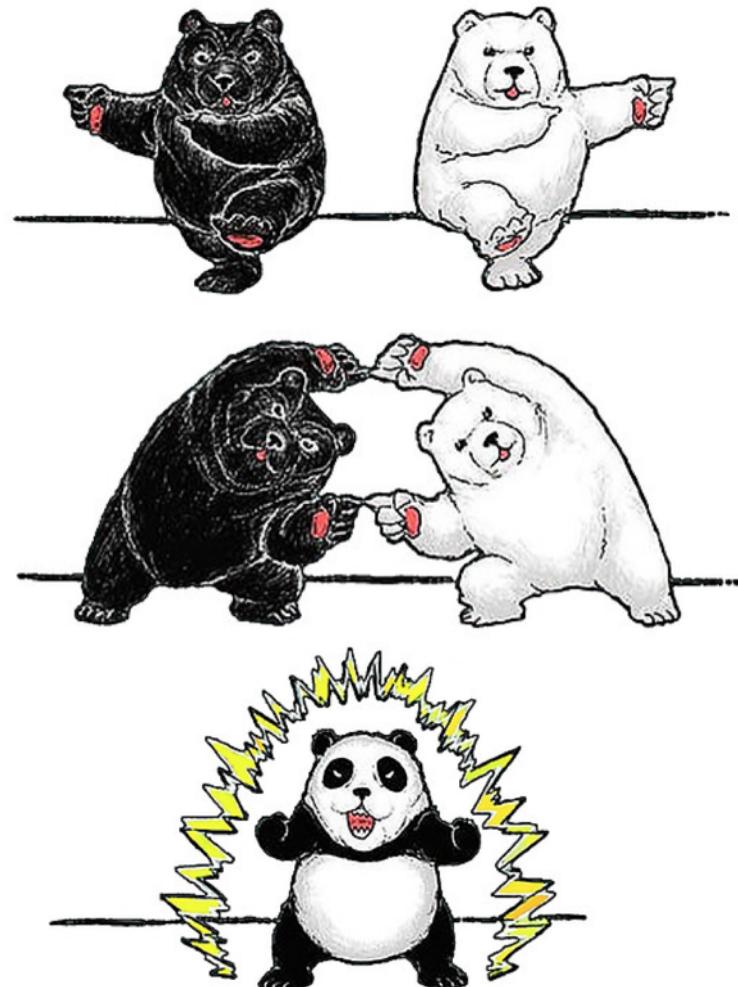
Data – User Table and Session Table (v4)

```
class UserTable{  
private:  
    vector<User> table;  
public:  
    ... Insert(...);  
    ... Delete(...);  
    ... Update(...);  
    ... Select(...);  
};
```

```
class SessionTable{  
private:  
    vector<Session> table;  
public:  
    ... Insert(...);  
    ... Delete(...);  
    ... Update(...);  
    ... Select(...);  
};
```

Can you see the **similarity** between the two tables?

Inheritance



Read More

Novice, <http://url.cn/TW5RYp>
Expert, <http://url.cn/aaAS9U>
Master, <http://url.cn/bFkGUj>

Data – User Table (v5)

```
class Table{  
    private:  
        vector<Row*> table;  
    public:  
        ... Insert(...);  
        ... Delete(...);  
        ... Update(...);  
        ... Select(...);  
};
```



```
class UserTable: public Table{  
};
```



```
class SessionTable: public Table{  
};
```

Data – Row Table

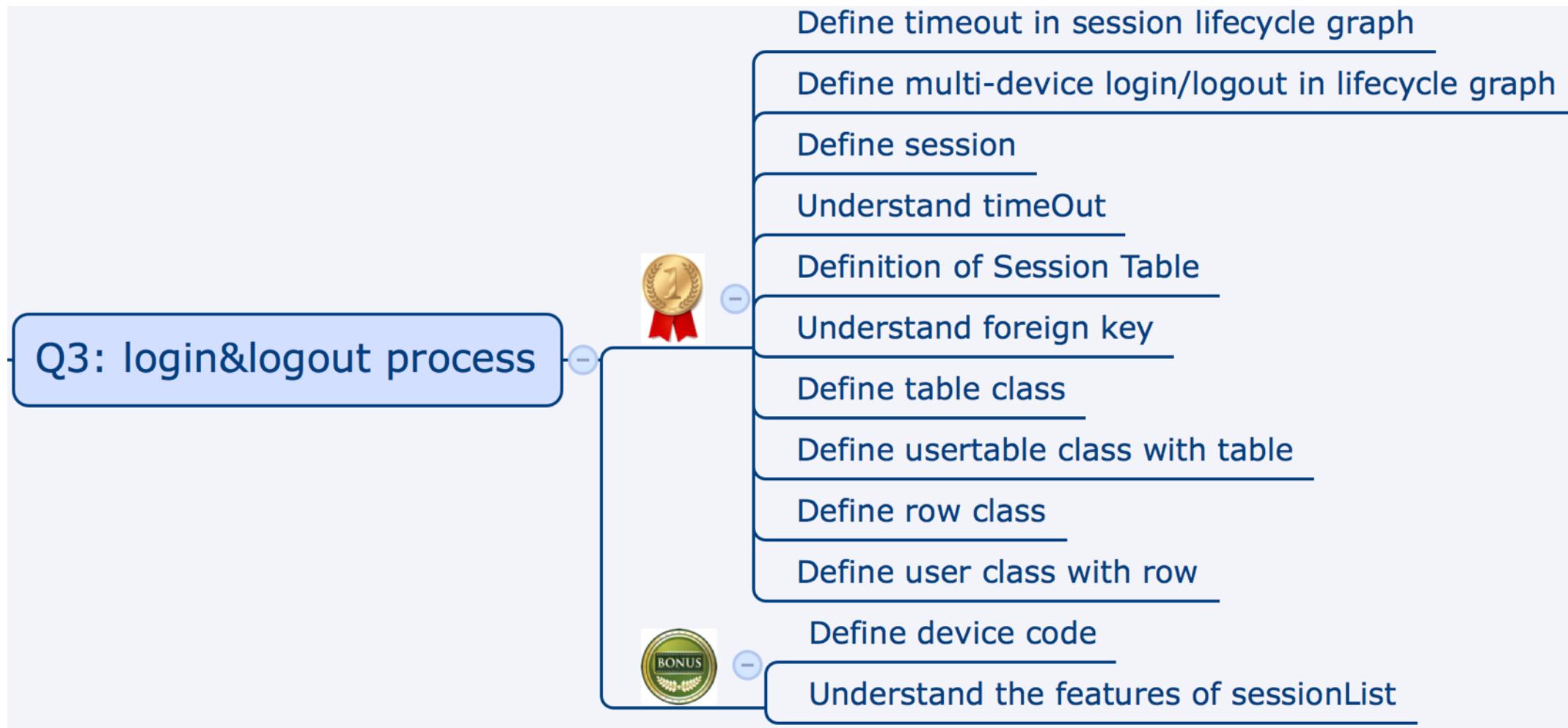
```
class Row{  
    private:  
        vector<Attribute*> row;  
};
```



```
class User: public Row{  
    ...  
};  
  
class Session: public Row{  
    ...  
};
```



Summary of Q3 (10+2)



Q4

Interviewer: find

- the name of the user whose `userID = 21`?
- the users whose `userIDs` are within `[4,20]`?

Challenge of lookup

- How to find the name of the user whose userID = 21?

- Algorithm

- Check 4 == 21 (No)
- Check 3 == 21 (No)
- Check 0 == 21 (No)
- Check 21== 21 (Yes)

- Time Complexity= $O(n)$

userID	name	hiddenPassword	state
4	Sangpo	A1V2F2G2F1	1
3	Steve	F4H1G1H7G1	2
0	Jobs	F1G4J5H1K1	3
21	Killer	M2J2J3N4M1	1
19	Me	M3J1B3N2N1	1

Index with hash

```
class UserTable{  
private:  
    vector<Row*> table;  
    Hash index;   
public:  
    ... Insert(...);  
    ... Delete(...);  
    ... Update(...);  
    ... Select(...);  
};
```

hashID	userID	pointer
0	21	
1	4	
2	3	
3	-	
4	0	
5	-	
6	19	

Lookup with hash

hashID	userID	pointer	userID	name	hiddenPassword	state
0	21		4	Sangpo	IFFDIFSLAF	1
1	4		3	Steve	JIJEFBIAFN	2
2	3		0	Jobs	DSFSDIENFA	3
3	-		21	Killer	FIDNFIANFD	1
4	0		19	Me	FDIAANFIDD	1
5	-					
6	19					



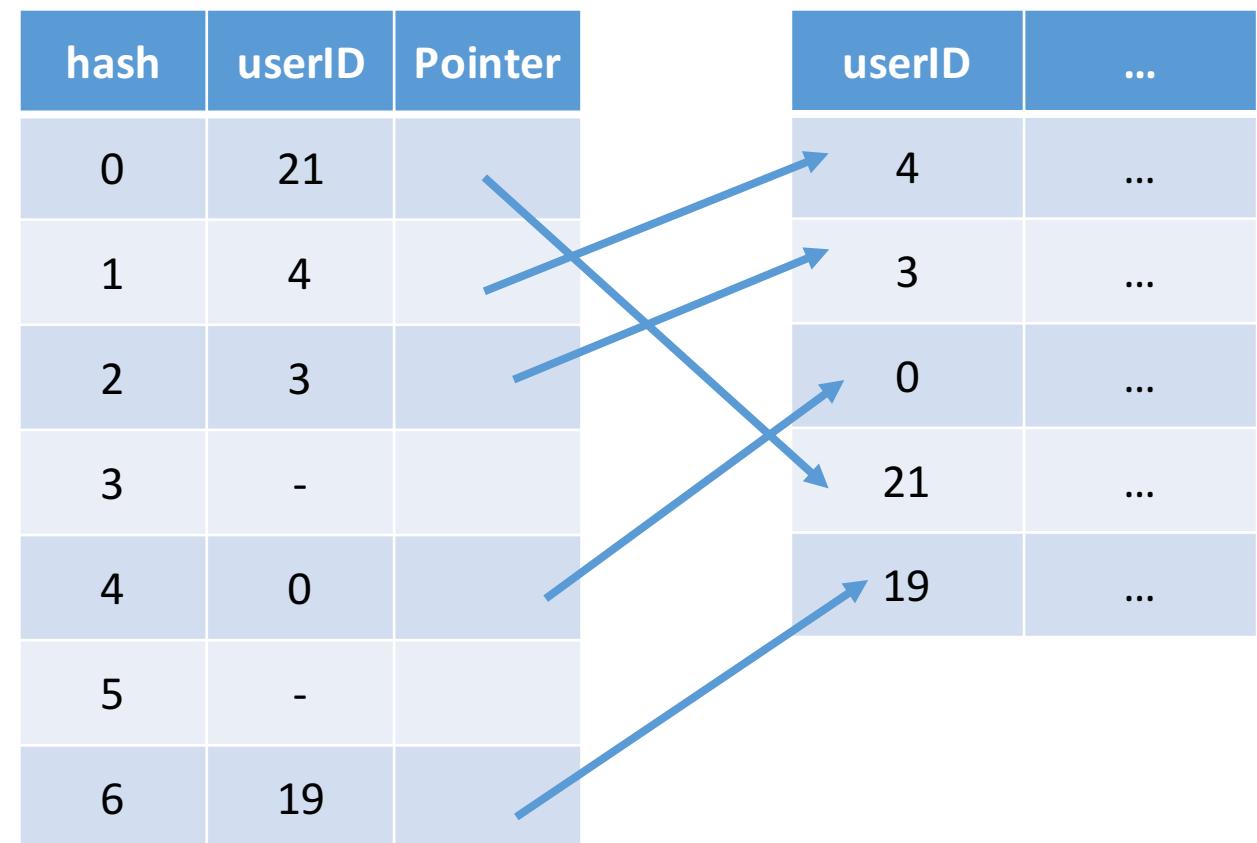
Time Complexity: O(1)
Space Complexity: O(hash size)

Challenge of range query

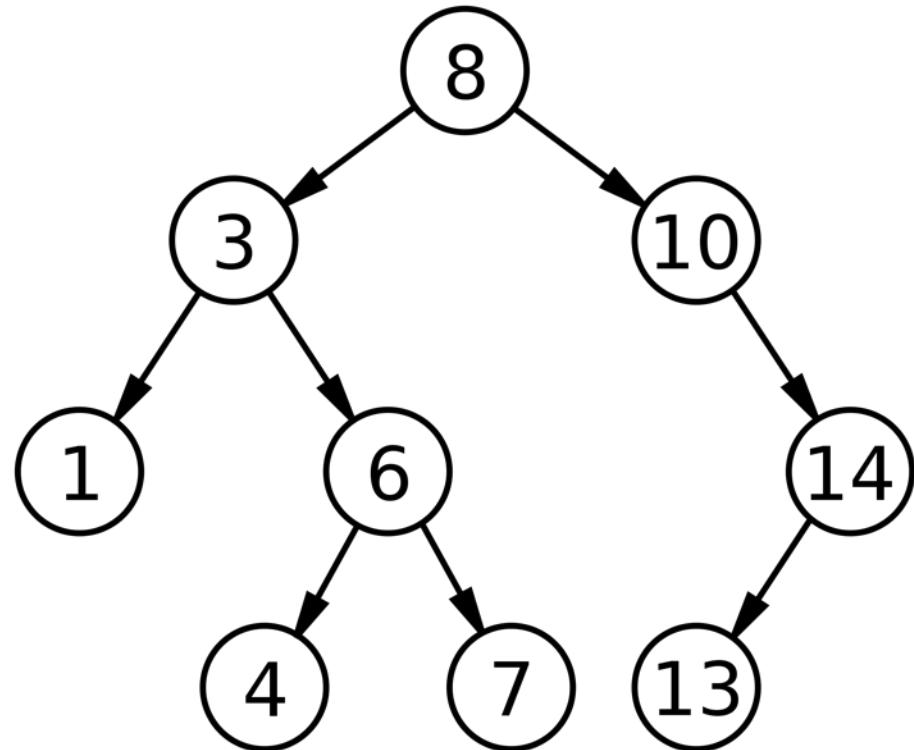
- How to find the users whose userIDs are within [4,20]?

- Algorithm

- Check 4 in [4,20] (Yes)
- Check 3 in [4,20] (No)
- Check 0 in [4,20] (No)
- Check 21 in [4,20] (No)
- Check 19 in [4,20] (Yes)
- Time Complexity = $O(n)$



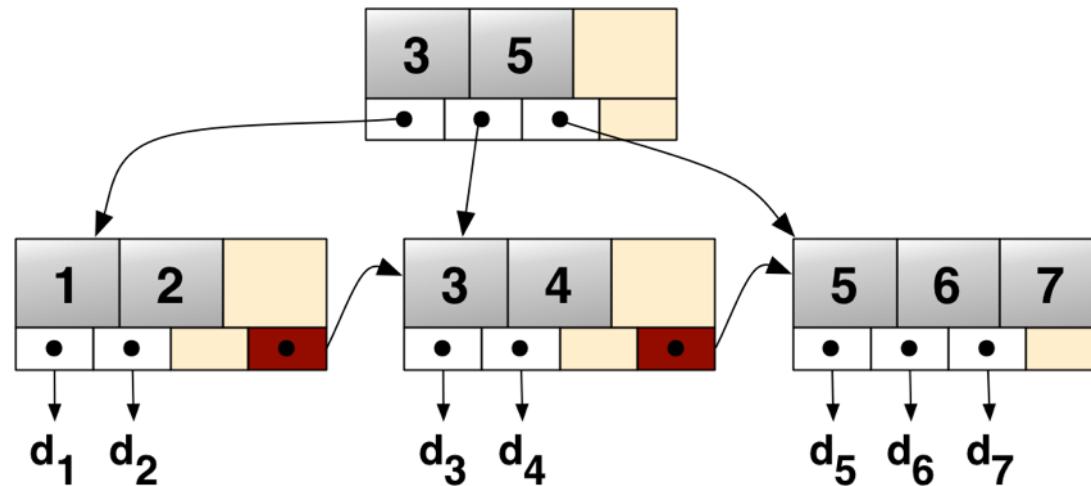
Index with Binary Search Tree



- A node's
 - Left sub-tree is smaller
 - Right sub-tree is larger
- Complexity
 - Space: $O(n)$
 - Time of insert: $O(\log_2 n)$
 - Time of delete: $O(\log_2 n)$
 - Time of lookup: $O(\log_2 n)$
 - Time of range query: $O(\log_2 n + k)$



Index with B+Tree

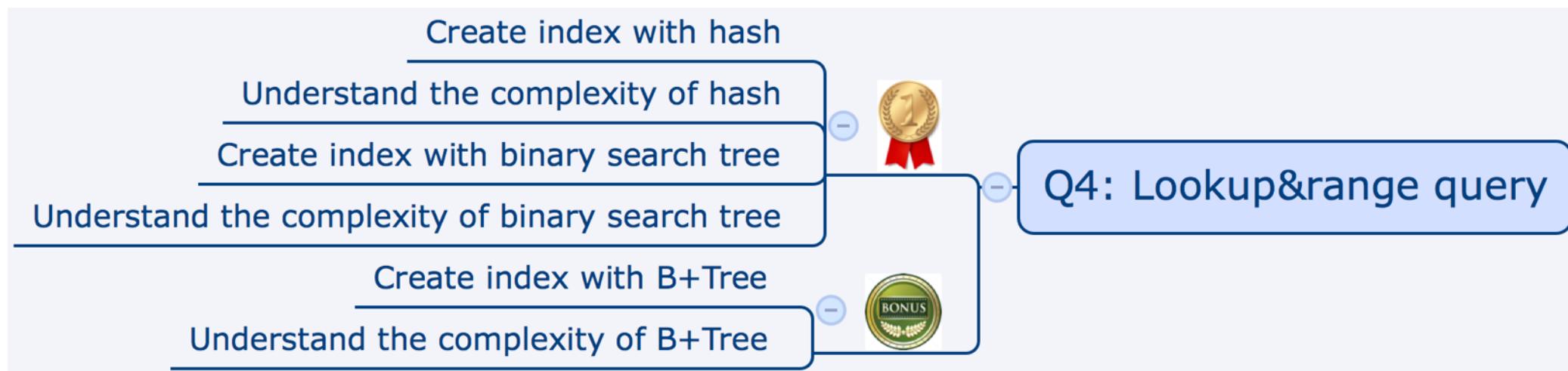


Read More

Master, <http://url.cn/UnKnTc>
Master, <http://url.cn/DA3mKn>
Master, <http://url.cn/3sLkWT>

- Complexity
- Space: $O(n)$
- Time of insert: $O(\log_b n)$
- Time of delete: $O(\log_b n)$
- Time of lookup: $O(\log_b n)$
- Time of range query: $O(\log_b n+k)$
- Advantage: disk friendly with continuous reading

Summary of Q4 (4+2)





A. Slouched Sitting



B. Forward Head Posture



C. Sitting at Attention



D. Proper Sitting Posture

Q5

Interviewer, support payment on **membership**



\$10

For less than the price of most albums,
you get access to the full Apple Music
library, expert recommendations, our
take on the best new music, and
unlimited skips on our radio stations.

Data - Membership

userID	money	endTime
3	20	16_07_2016

```
class Membership{  
      
    private:  
        int userID;  
        double money;  
        time_t endTime;  
      
    public:  
        ... addMoney(...);  
        ... buyMember(...);  
};
```

Interviewer: what is the problem? (5.1)

- When user3 purchases Membership

userID	money	endTime
3	20	16_07_2016

- Good case
 - $\text{money} = \text{money} - 10 = 10$
 - $\text{endTime} += 1\text{month} = 16_08_2016$
 - Finally
 - $\text{money} = 10$
 - $\text{endTime} = 16_08_2016$
- Bad case
 - $\text{money} = \text{money} - 10 = 10$
 - System goes down
 - Finally
 - $\text{money} = 10$
 - $\text{endTime} = 16_07_2016$

Solution: add transaction with log



- Fixed bad case
 - Write LOG
 - Transaction_1123: BEGIN
 - money=20; endTime=16_07_2016
 - money = money - 10 = 10
 - System goes down
 - System goes up
 - Read LOG
 - Transaction_1123: BEGIN
 - money=20; endTime=16_07_2016
 - Recover
 - money = 20
 - endTime = 16_07_2016
 - Finally
 - money = 20
 - endTime = 16_07_2016

[Read More](#)

Expert, <http://url.cn/Sp4ORB>
Master, <http://url.cn/Ni1lxL>
Diamond, <http://url.cn/OAwnmr>

Solution: add transaction with log

- Good case
 - Write LOG
 - Transaction_1123: BEGIN
 - money=20; endTime=16_07_2016
 - money = money - 10 = 10
 - endTime += 1month = 16_08_2016
 - Write LOG
 - Transaction_1123: END
 - money=10; endTime=16_08_2016
- Finally
 - money = 10
 - endTime=16_08_2016

Interviewer: what is the problem? (5.2)

UserTable

userID	name	hiddenPassword	state
4	Sangpo	A1V2F2G2F1	1
3	Steve	F4H1G1H7G1	2
0	Jobs	F1G4J5H1K1	3
21	Killer	M2J2J3N4M1	1
19	Me	M3J1B3N2N1	1

MembershipTable

userID	Money	endTime
3	20	16_07_2016
4	100	21_12_2016
4	40	13_06_2015
0	70	14_12_2015
18	70	11_12_2015
19	0	08_07_2015

Solution with checker

```
class Checker{  
    private:  
        vector<function*> checkList;  
    public:  
        bool register(function* checkFunction);  
};
```



```
Checker checker;  
checker.register( ...checkUnknownUser(...){...} )  
checker.register( ...checkDuplicatedUser(...){...} )
```

Interviewer: what is the problem? (5.3)

- When user3 purchases Membership twice

userID	money	membership
3	20	16_07_2016

First time:

- 1.1 Read: money = 20
- 1.2 Read: endTime = 16_07_2016
- 1.3 Write: money = money-10 = 10
- 1.4 Write: endTime += 1month = 16_08_2016

Second time:

- 2.1 Read: money = **10**
- 2.2 Read: endTime = 16_08_2016
- 2.3 Write: money = money-10 = 0
- 2.4 Write: endTime += 1month = 16_09_2016

money = **0**
endTime = 16_09_2016

Bad case

- When user3 purchases Membership twice

First time:

1.1 Read: money = 20

1.2 Read: endTime = 16_07_2016

1.3 Write: money = money-10 = 10

1.4 Write: endTime += 1month = 16_08_2016

Second time:

2.1 Read: money = 20

2.2 Read: endTime = 16_08_2016

2.3 Write: money = money-10 = 10

2.4 Write: endTime += 1month = 16_09_2016

userID	money	membership
3	20	16_07_2016

money = 10
endTime = 16_09_2016

Solution with lock (fixed bad case)



First time:

1.0 Lock money&endTime

1.1 Read: money = 20

1.2 Read: endTime = 16_07_2016

1.3 Write: money = money-10 = 10

1.4 Write: endTime += 1month = 16_08_2016

1.5 Release money&endTime

Second time

2.0 Lock money&endTime

(sleep)

ZzZzz

zZzZz

zzZZz

(wakeup)

2.1 Read: money = 10

2.2 Read: endTime = 16_08_2016

2.3 Write: money = money-10 = 0

2.4 Write: endTime += 1month = 16_09_2016

2.5 Release money&endTime

money = 0

endTime = 16_09_2016

Solution with lock (good case)

First time:

- 1.0 Lock money&endTime
- 1.1 Read: money = 20
- 1.2 Read: endTime = 16_07_2016
- 1.3 Write: money = money-10 = 10
- 1.4 Write: endTime += 1month = 16_08_2016
- 1.5 Release money&endTime

Second time:

- 2.0 Lock money&endTime
- 2.1 Read: money = 10
- 2.2 Read: endTime = 16_08_2016
- 2.3 Write: money = money-10 = 0
- 2.4 Write: endTime += 1month = 16_09_2016
- 2.5 Release money&endTime

money = 0
endTime = 16_09_2016

***STOP: 0x000000D1 (0x00000000, 0xF73120AE, 0xC0000008, 0xC0000000)

A problem has been detected and Windows has been shut down to prevent damage
to your computer

DRIVER_IRQL_NOT_LESS_OR_EQUAL

If this is the first time you've seen this Stop error screen, restart your
computer. If this screen appears again, follow these steps:

Check to make sure any new hardware or software is properly installed. If this is a
new installation, ask your hardware or software manufacturer for any windows updates
you might need.

If problems continue, disable or remove any newly installed hardware or software.
Disable BIOS memory options such as caching or shadowing. If you need to use Safe
Mode to remove or disable components, restart your computer, press F8 to select
Advanced Startup Options, and then select Safe Mode.

**** ABCD.SYS - Address F73120AE base at C0000000, DateStamp 36B072A3

Kernel1 Debugger Using: COM2 (Port 0x2F8, Baud Rate 19200)

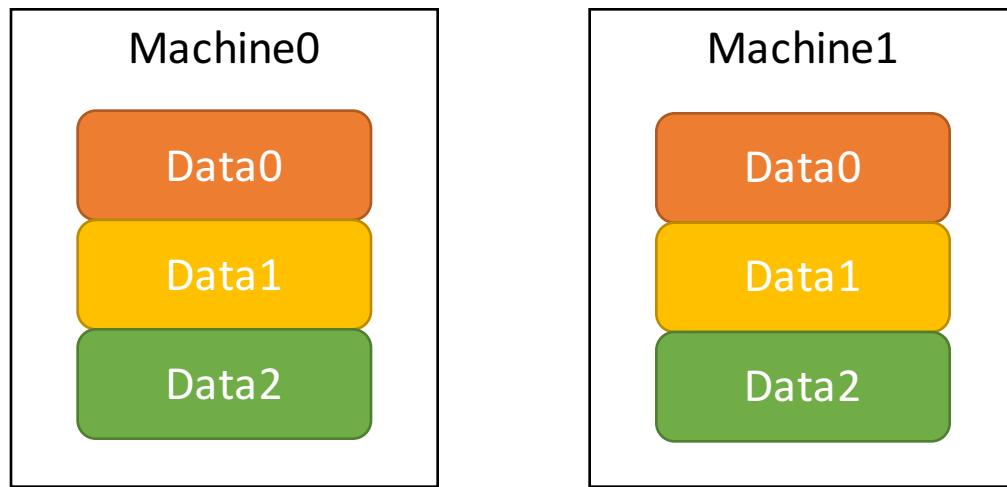
Beginning dump of physical memory

Physical memory dump complete. Contact your system administrator or
technical support group.

Interviewer: what is the problem? (5.4)



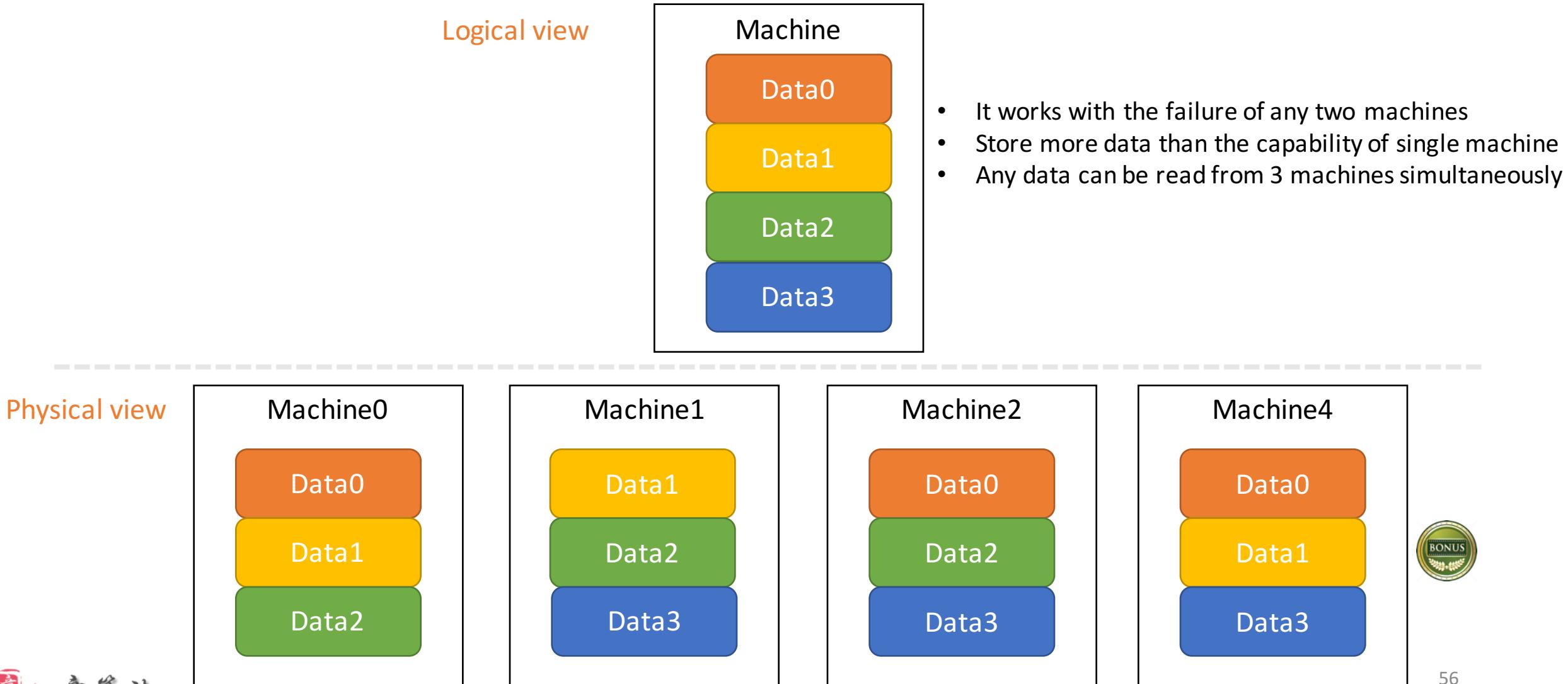
Solution with duplication (v1)



Read More

Master, <http://url.cn/bsBiDk>
Master, <http://url.cn/d1BrZU>

Duplication with more data and machines (v2)



One more step: ACID

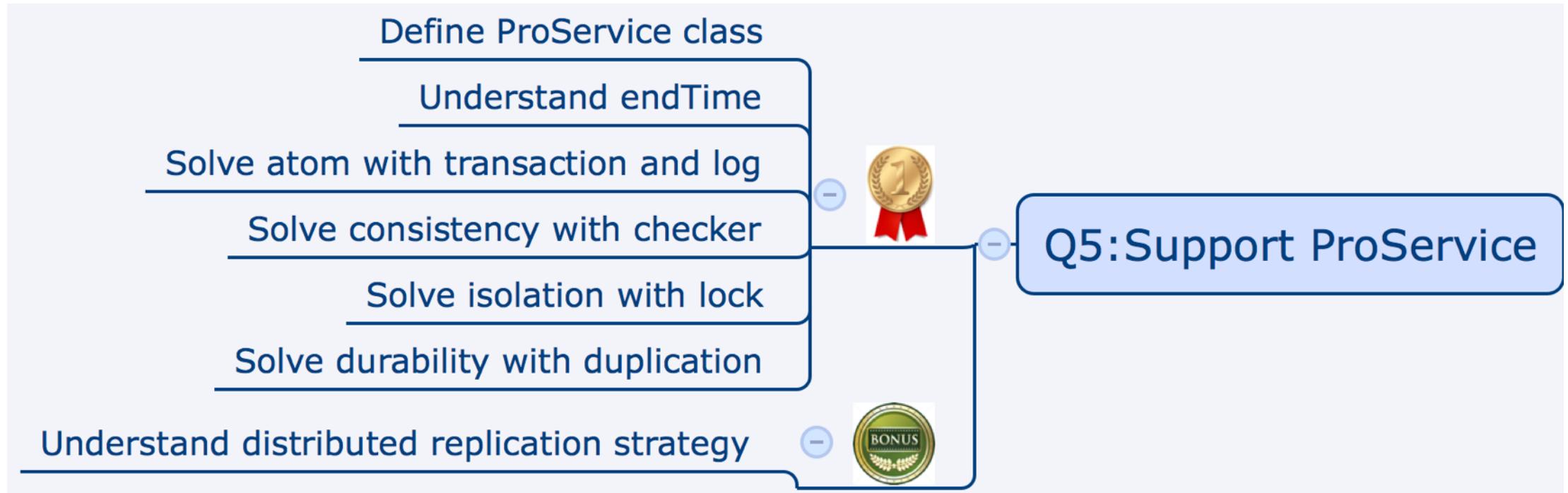
- Q5.1: **Atomicity**, all or nothing
- Q5.2: **Consistency**, valid according to all defined rules
- Q5.3: **Isolation**, a kind of independency between transactions
- Q5.4: **Durability**, stored permanently

Read More

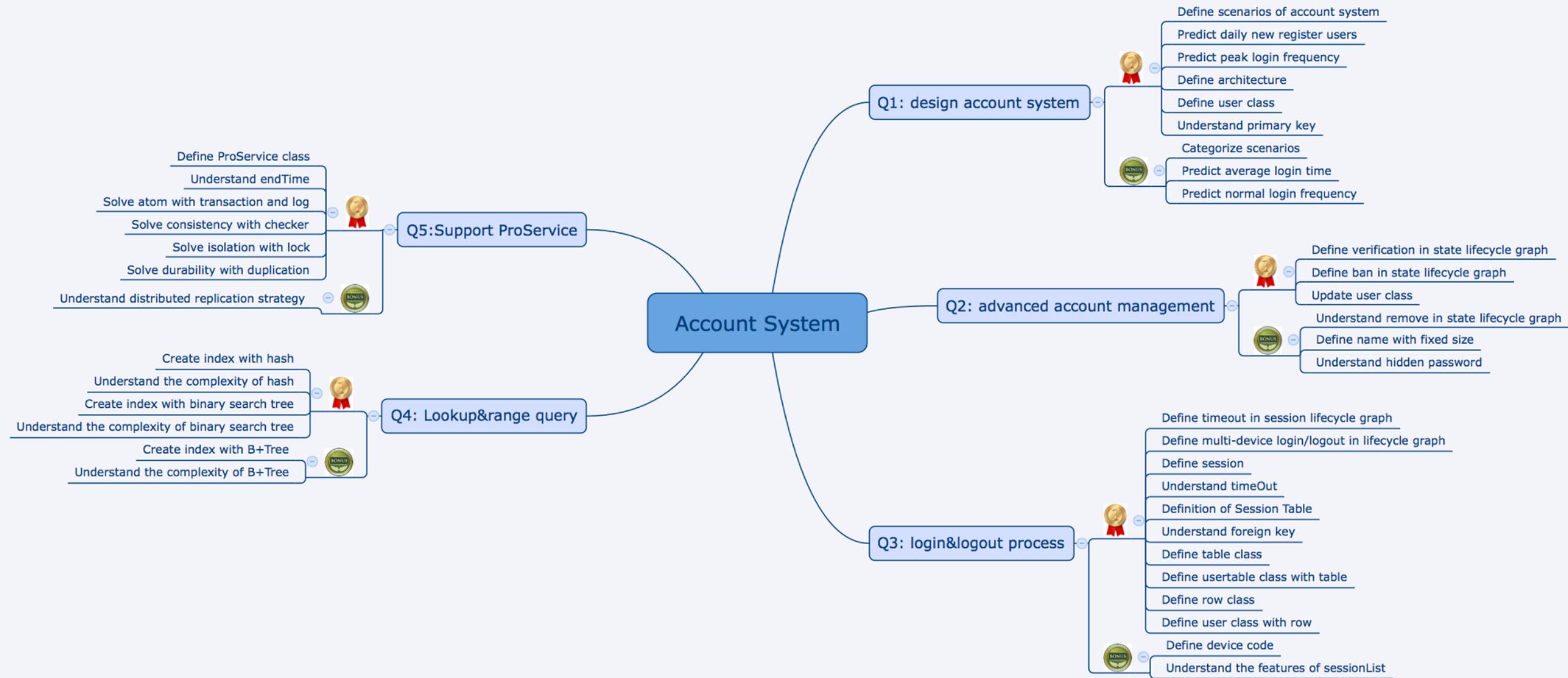
Novice, <http://url.cn/UzRaJS>
Master, <http://url.cn/SJdvIr>



Summary of Level 5 (6+1)



Summary of class



Keyword

- Primary/foreign key
- Table
- Row
- Attribute
- Index
- Transaction
- Log
- Lock
- Lifecycle graph
- Binary search tree
- B+ Tree
- Atomicity
- Consistency
- Isolation
- Durability
- Session

One more thing

Interviewer: design PayPal

[Read More](#)

Novice, <http://url.cn/iTJ1JF>

Expert, <http://url.cn/2eHHsU>

Expert/Master, <http://url.cn/kPMc7X>

Let's do it together ☺

QA



关注微信/微博，获取最新面试题及权威解答

微信: [ninechapter](#)

微博: <http://www.weibo.com/ninechapter>

官网: www.jiuzhang.com

Enthronement

By Chögyam Trungpa, January 22, 1973

From <<The Myth of Freedom and the Way of Meditation>>

登基

秋阳创巴，01.22.1973

节选自《自由的迷思》

Parents are very kind,
But I am too young to appreciate it.
The highland mountains and valleys are beautiful,
But having never seen the lowlands, I am stupid.

Having striven for mind's nourishment,
Sharpening the spearhead of intellect,
I discovered permanent parents
Whom I can never forget.

Having no one to influence my outlook,
I display my primordial nature
And adopt the style of a youthful prince.
This is due to the only father guru.

I am busy working for others.
Prajna, penetrating all obstacles,
Has made the prince old and wise,
Fearing no one.

父母是慈悲的，
但年少的我不懂得感激，
高原的山谷是美丽的，
但没见过平原的我是无知的。

追寻心的滋养，
磨砺智慧的锋芒，
我遇到了永恒的父母，
永难忘。

没人能影响我的显现，
我本初的自然
成为了年轻的王子。
这是因为我的唯一的父亲。

我为他人忙碌。
智慧穿透障碍，
成熟了王子，
毫无畏惧。

Dancing in space,
Clad in clouds,
Eating the sun and holding the moon,
The stars are my retinue.

The naked child is beautiful and dignified.
The red flower blooms in the sky.
It is ironic to see the formless dancer,
Dancing to the trumpet without a trumpeter.

At the palace of red ruby,
Listening to the utterance of the seed syllable,
It is joyful to watch the dance of illusion,
The seductive maidens of phenomena.

The warrior without a sword,
Riding on a rainbow,
Hears the limitless laughter of transcendent joy,
The poisonous snake becomes amrita.

在空中舞蹈，
披着云的霓裳，
吞下太阳握住月亮，
所有的星星都是我的随从。

赤裸的孩子是美丽而高贵的。
红色的花朵在天空绽放。
居然看到了无形的舞者，
随着没有号手的小号舞蹈。

在红宝石的宫殿，
听着种子字的轰鸣，
愉快的看着在虚幻中摇曳的，
现实的撩人的舞娘。

无需宝剑的勇士，
骑着彩虹，
听着无休无止的超然的笑声，
喝着毒液酿造的甜酒。

Drinking fire, wearing water,
Holding the mace of the wind,
Breathing earth,
I am the lord of the three worlds.

饮着火，穿着水，
握着风的权杖，
呼吸着大地，
我是三界的王。

Homework

- <http://www.jiuzhang.com/qa/88/>