

Enum в API — коварство иллюзорной простоты

#грабли #architecture #microservices



Илья Сазонов

Руководитель разработки в сервисе рассрочек Всегда.Да

Разработчик в части бэкенда по B2B и B2C направлениям, также занимался патентными исследованиями и геймдевом.

Интересуется вопросами организации кода в стремительно развивающихся проектах с большим количеством участников.

Связаться: почта – poxvuibr@gmail.com; Хабр – habr.com/en/users/poxvuibr/



Федор Сазонов

Java-разработчик, независимый эксперт

За свою карьеру работал как над простым бэкендом для веб-страниц, так и над высоконагруженными серверами для MMO (мобильные шутеры). Консультирует по созданию игровых решений.

Работал в нескольких стартапах, занимался выстраиванием процессов разработки и доставки продукта до конечных пользователей.

Всегда.Да

Сервис для покупки товаров и услуг в рассрочку. С нами это так же просто, как покупка по карте или за наличные

- Сервис запущен в 2017 году **международной группой PPF**
- За три с небольшим года – из стартапа в прибыльную компанию. **37 млрд рублей – объем продаж в 2020 году**
- **5,5 млн клиентов**

Сайт компании:



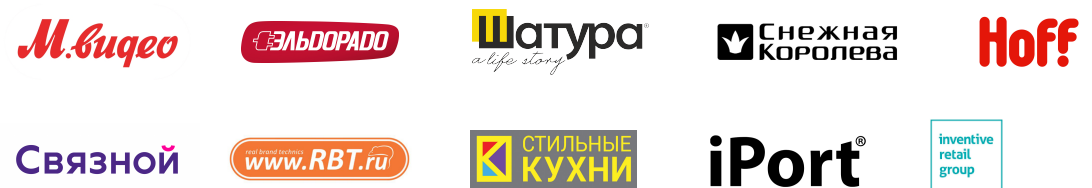
Среди партнеров:

Крупнейшие банки и МФО

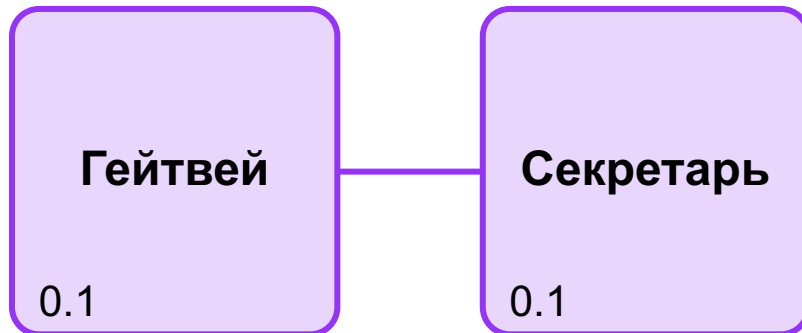


- **Продуктовый подход.** 12 agile-команд на 4 направления – сервис рассрочек, кредитование наличными, мобильное приложение и личный кабинет партнера
- **Современный стек:** Java, Javascript/Typescript, Kotlin и Swift
- Компания **активно растет** и уже сейчас является **лидером рынка** по количеству подключенных магазинов – 77К

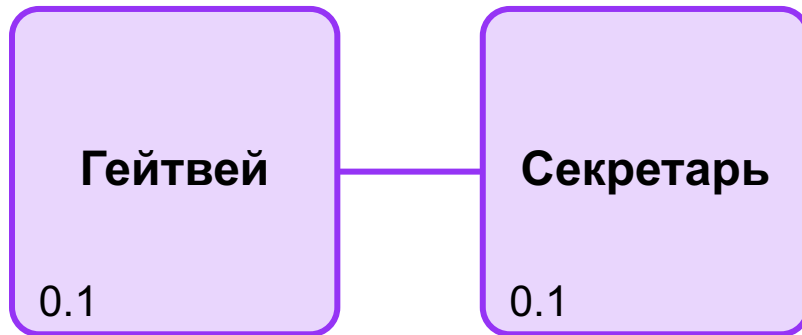
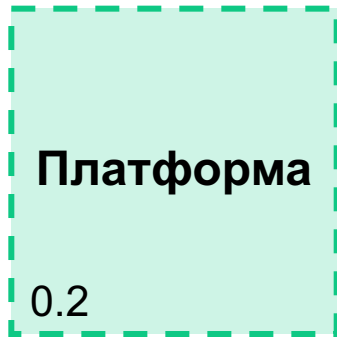
Более 40 федеральных сетей

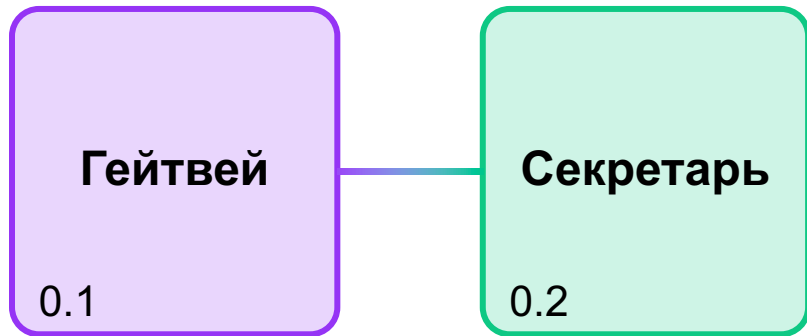
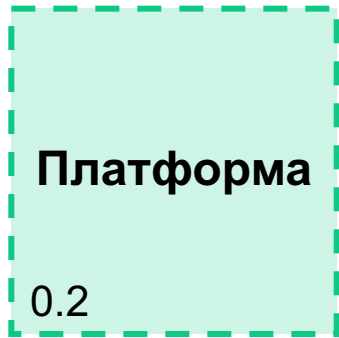


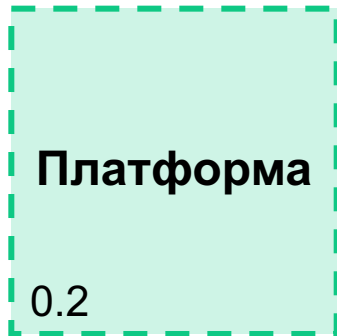




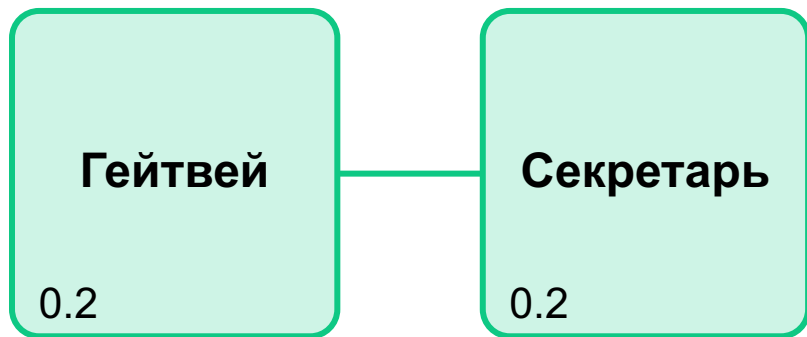
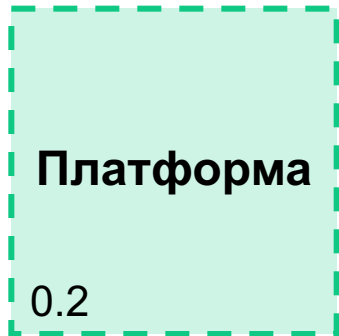
```
public String mapClientOccupation(ClientOccupation occupation) {  
    if (occupation == ClientOccupation.RETIRED) {  
        return ClientOccupation.RETIRED.name();  
    } else {  
        return ClientOccupation.WORKER.name();  
    }  
}
```

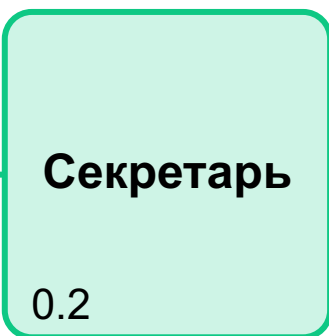
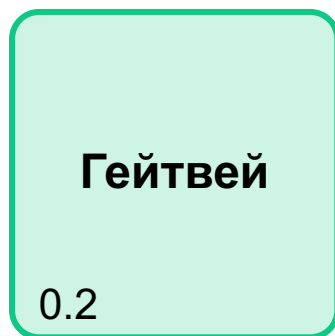
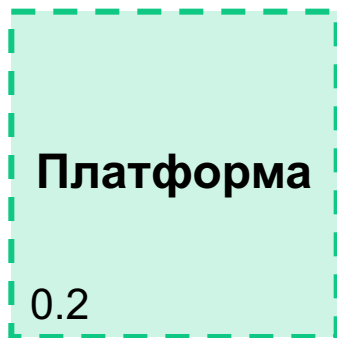


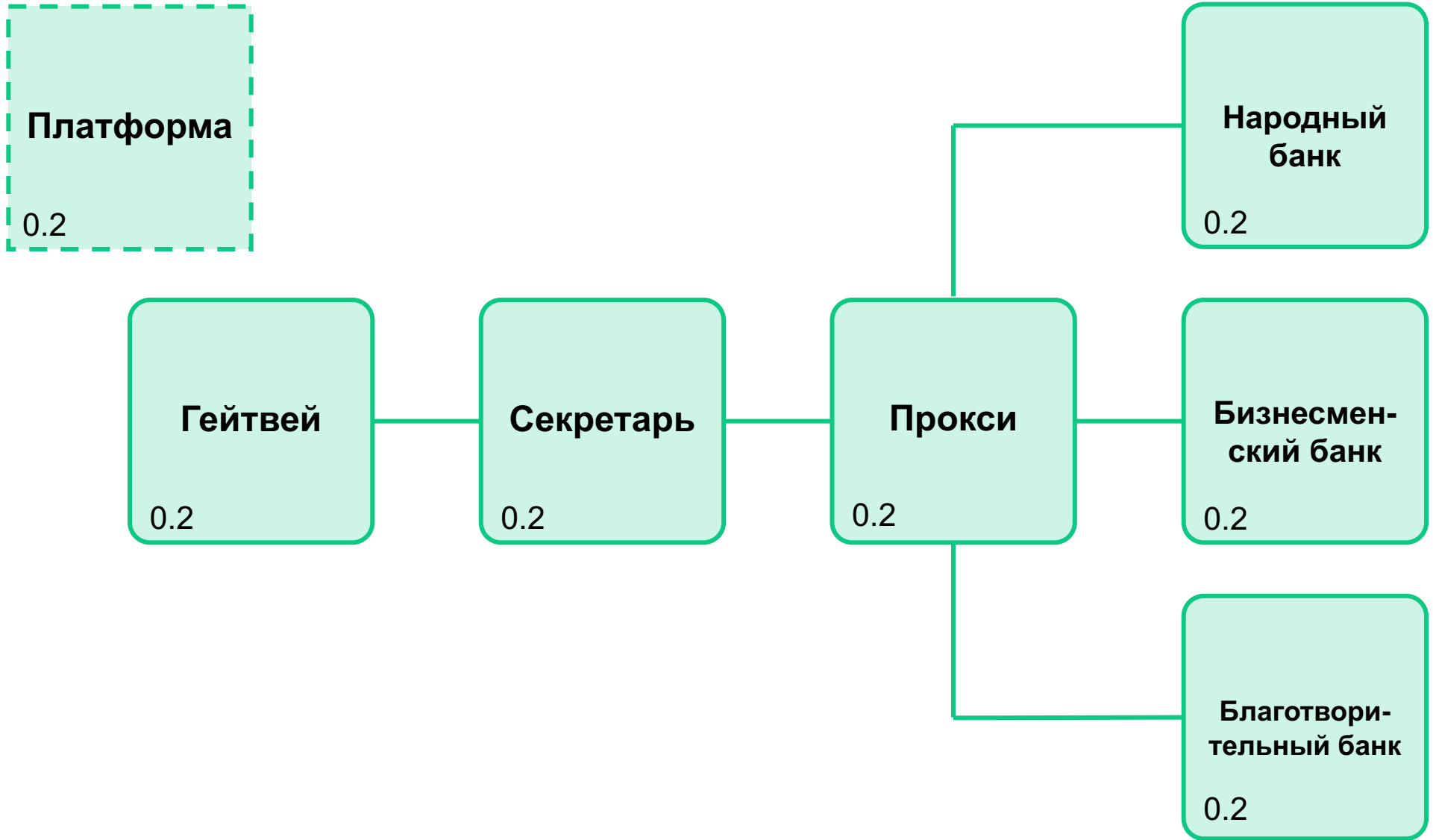


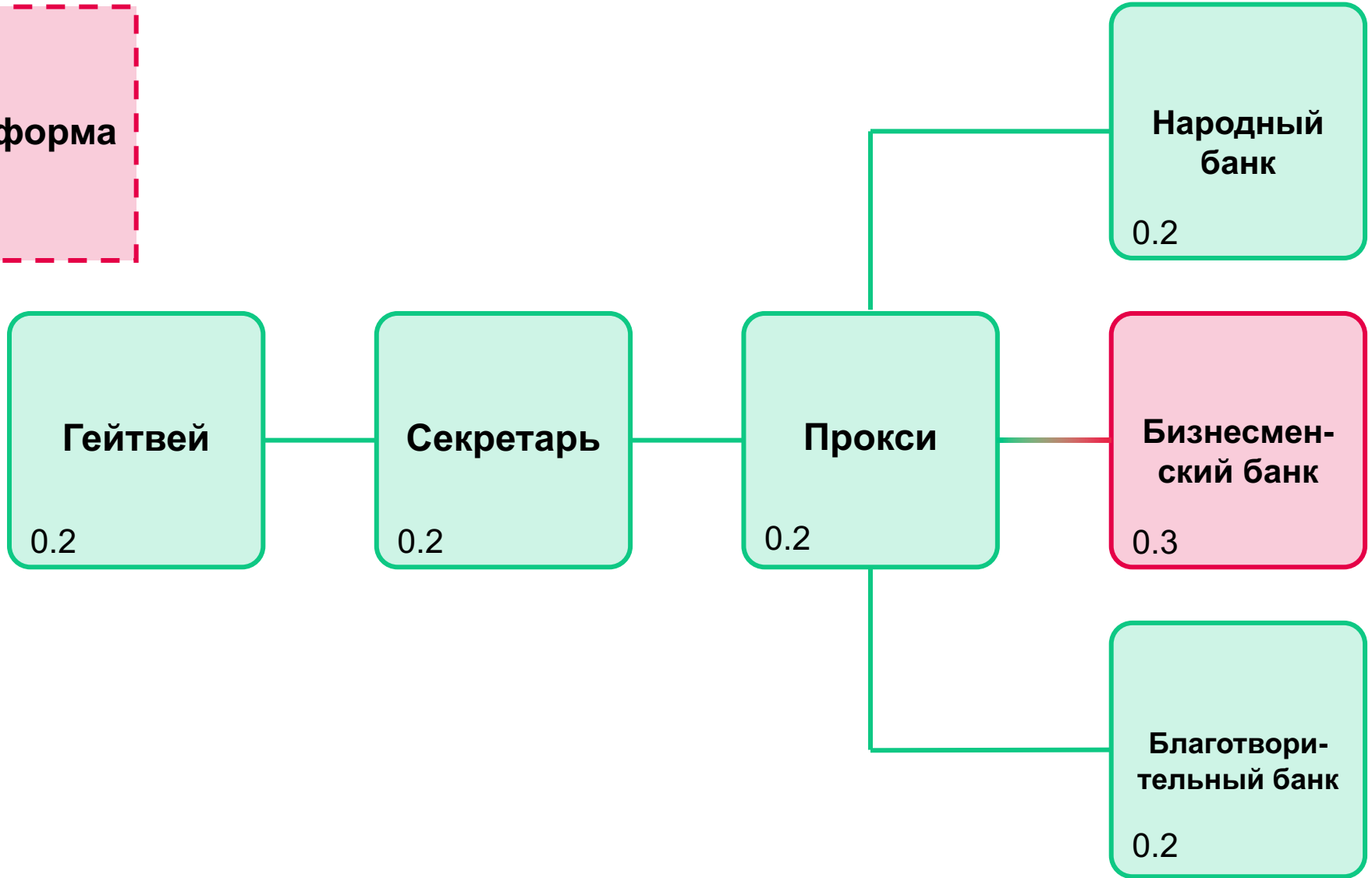


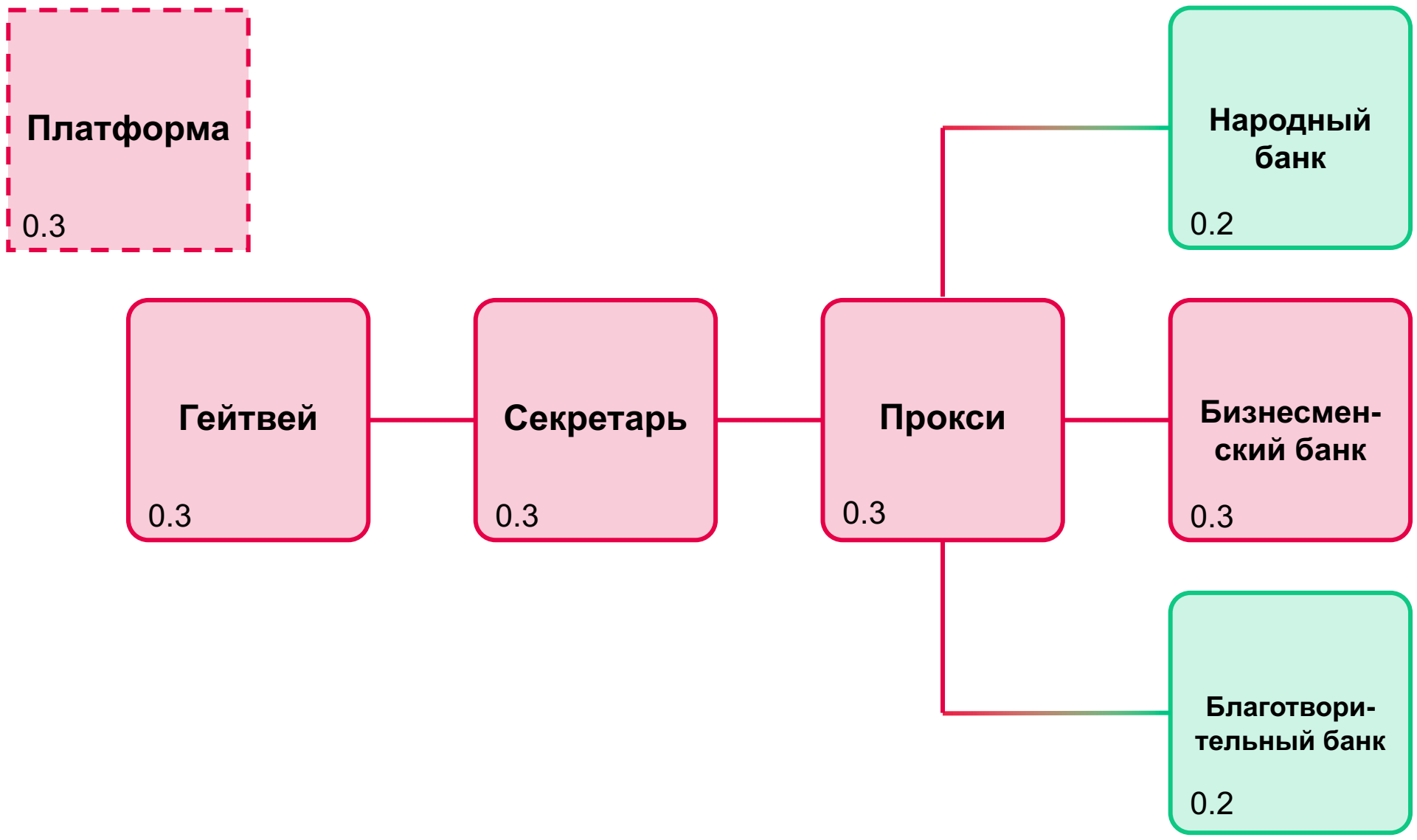
```
public String mapClientOccupation(ClientOccupation occupation) {  
    if (    occupation == ClientOccupation.RETIRED  
        || occupation == ClientOccupation.STUDENT ) {  
        return occupation.name();  
    } else {  
        return ClientOccupation.WORKER.name();  
    }  
}
```

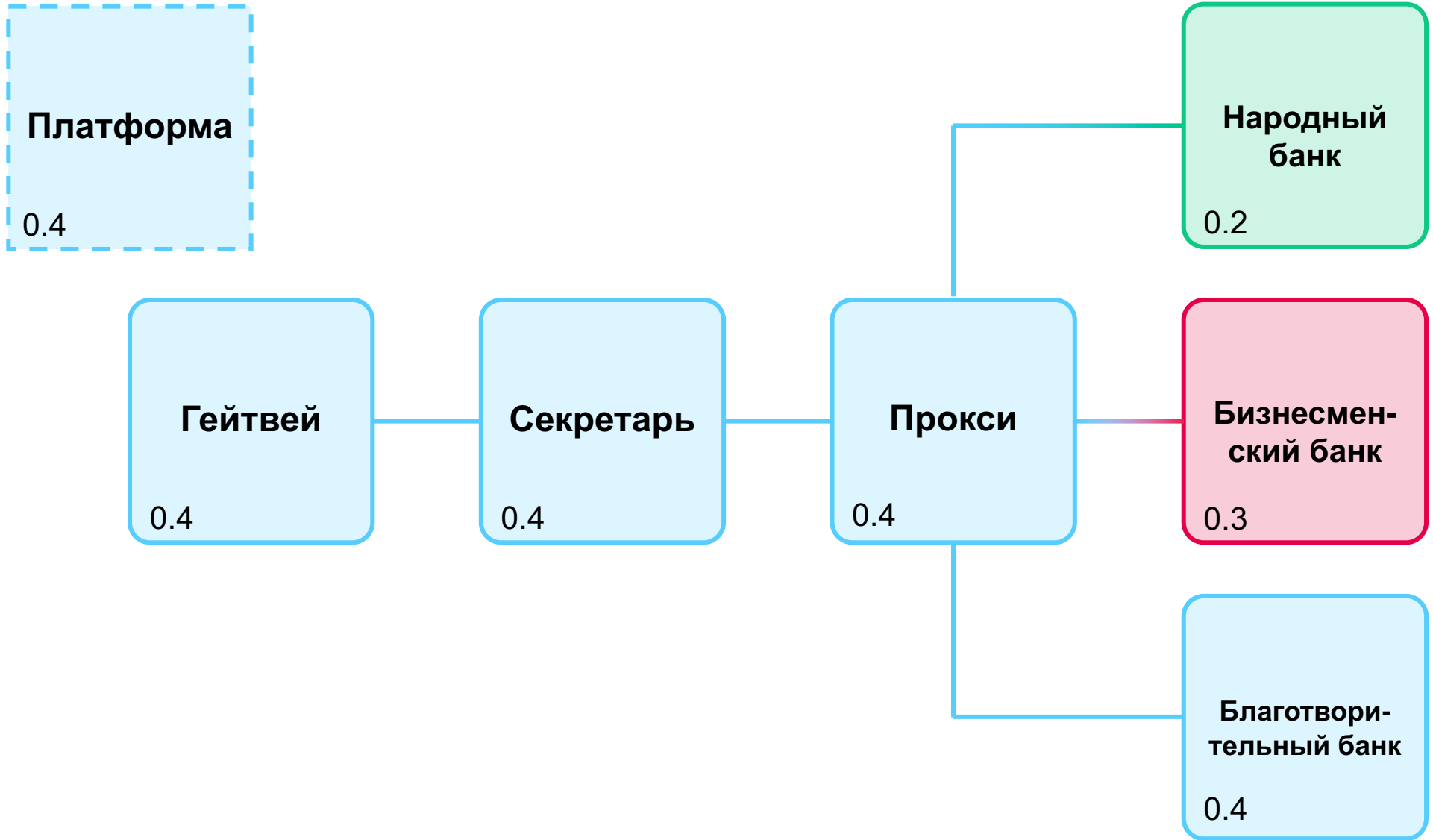



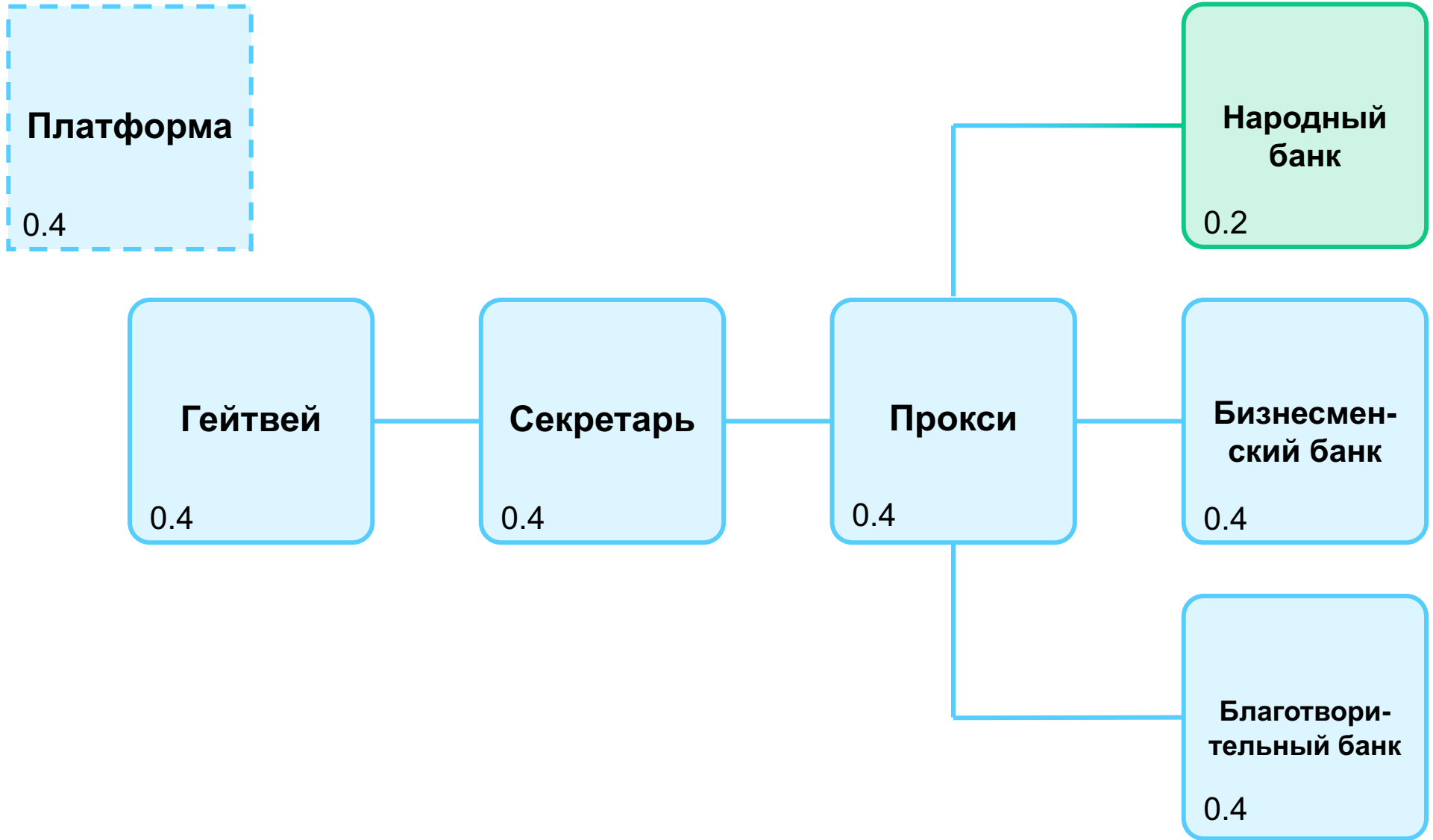


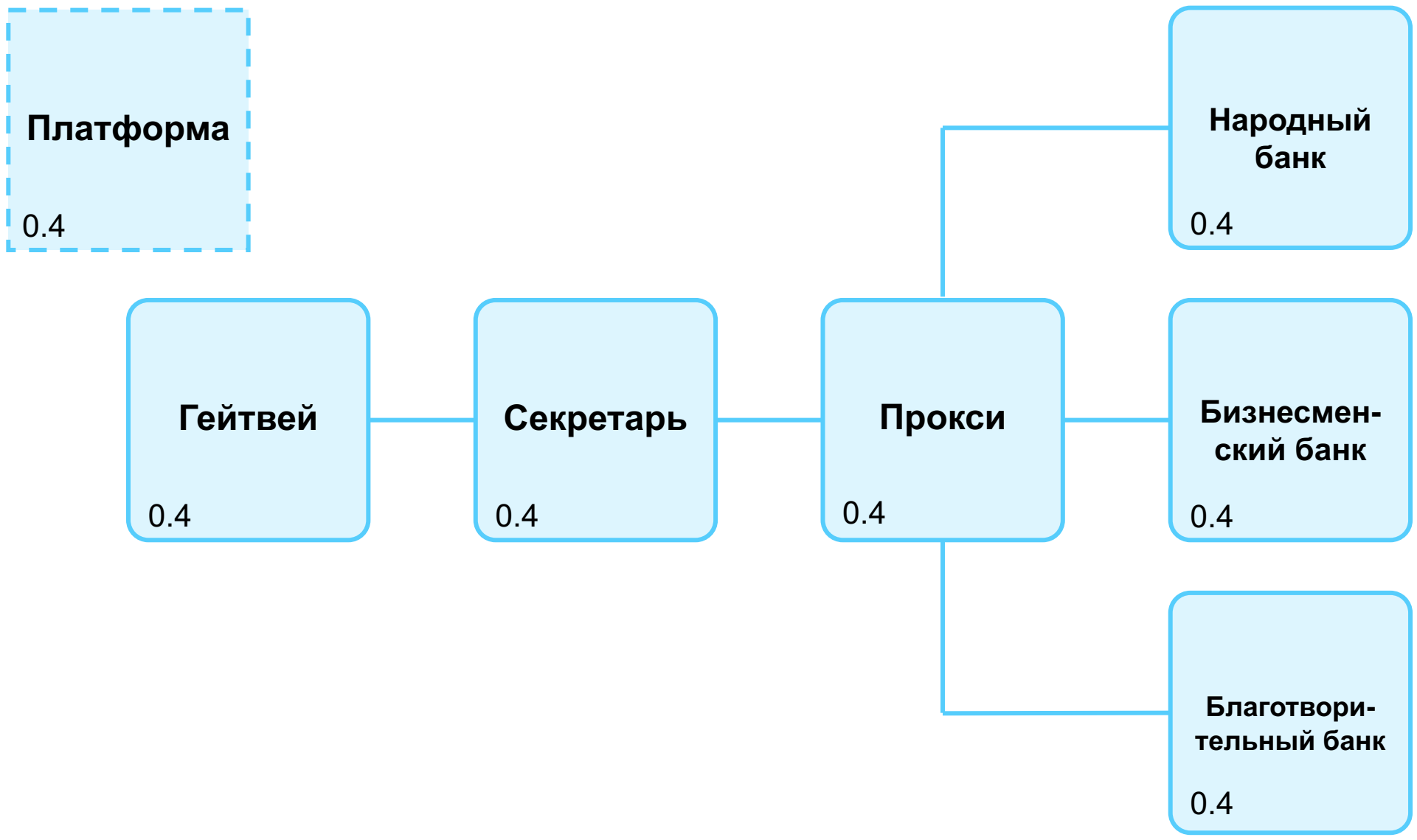


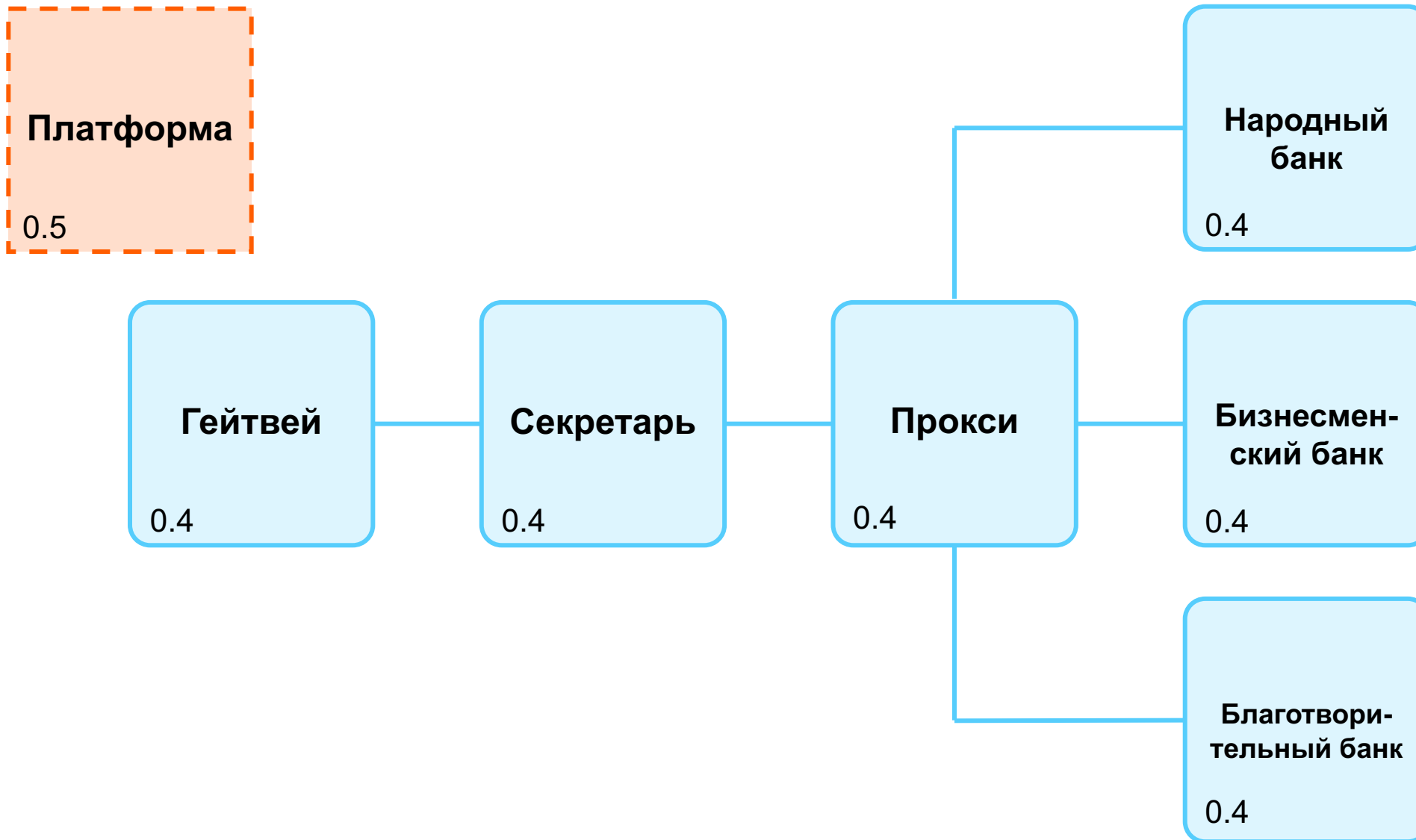


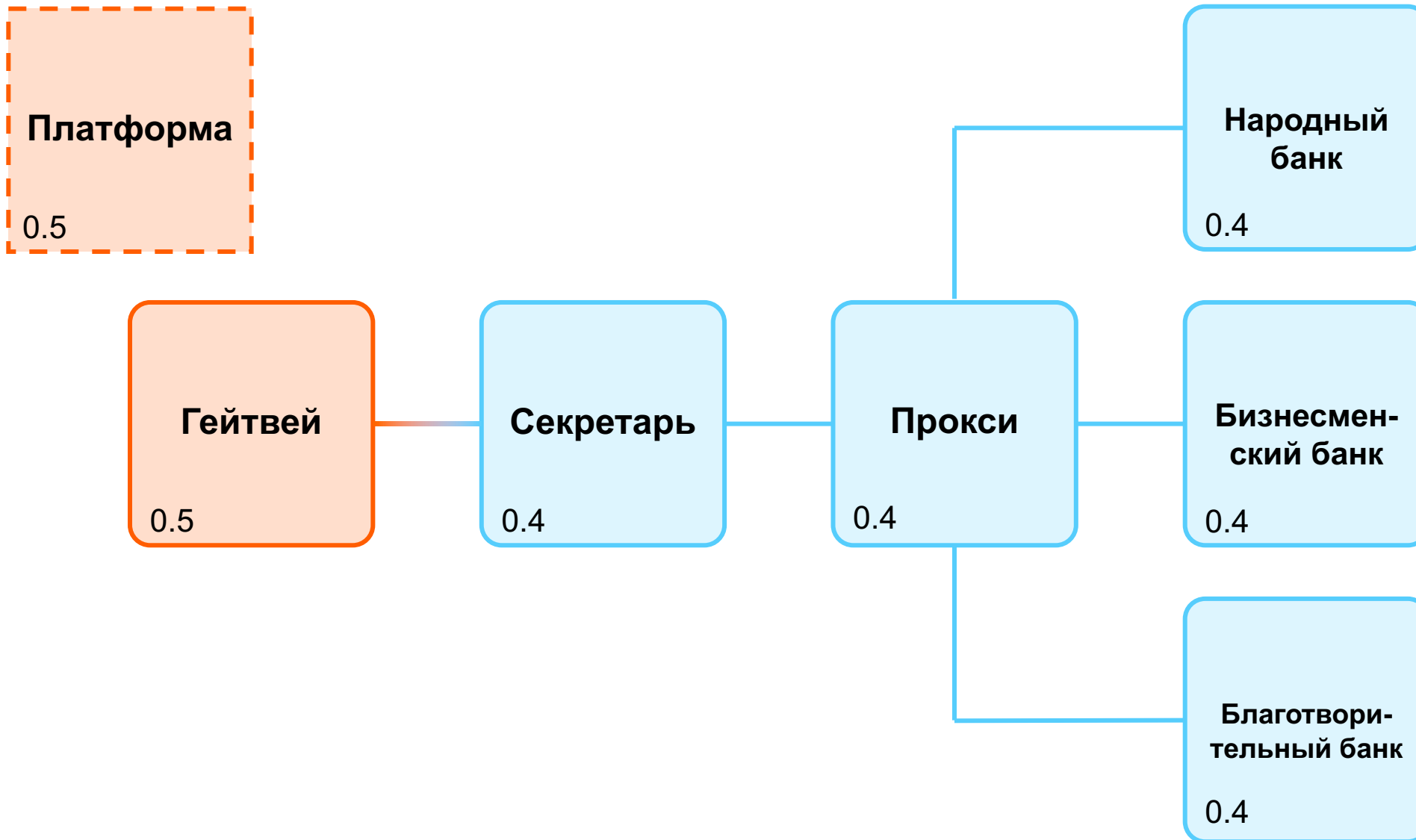


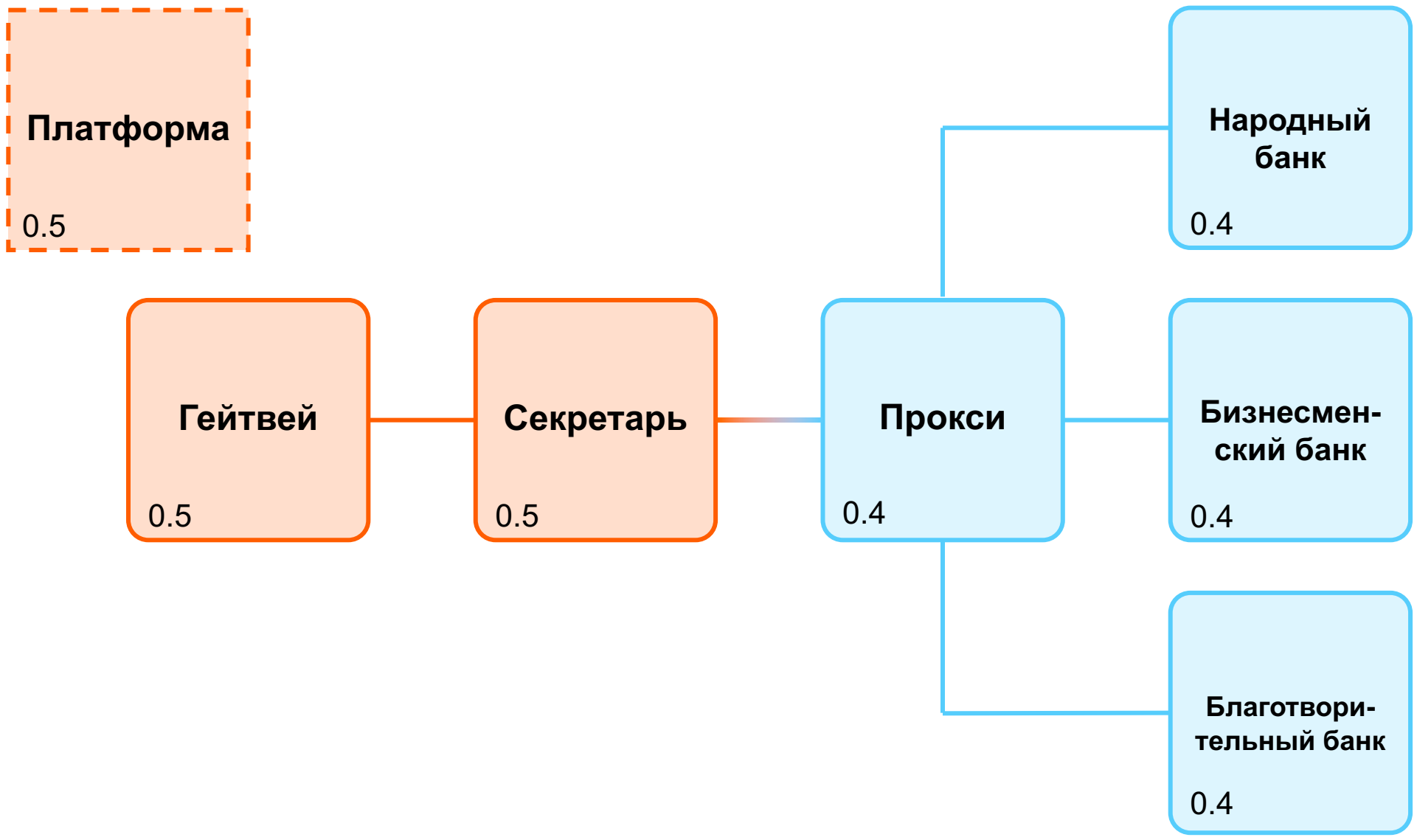


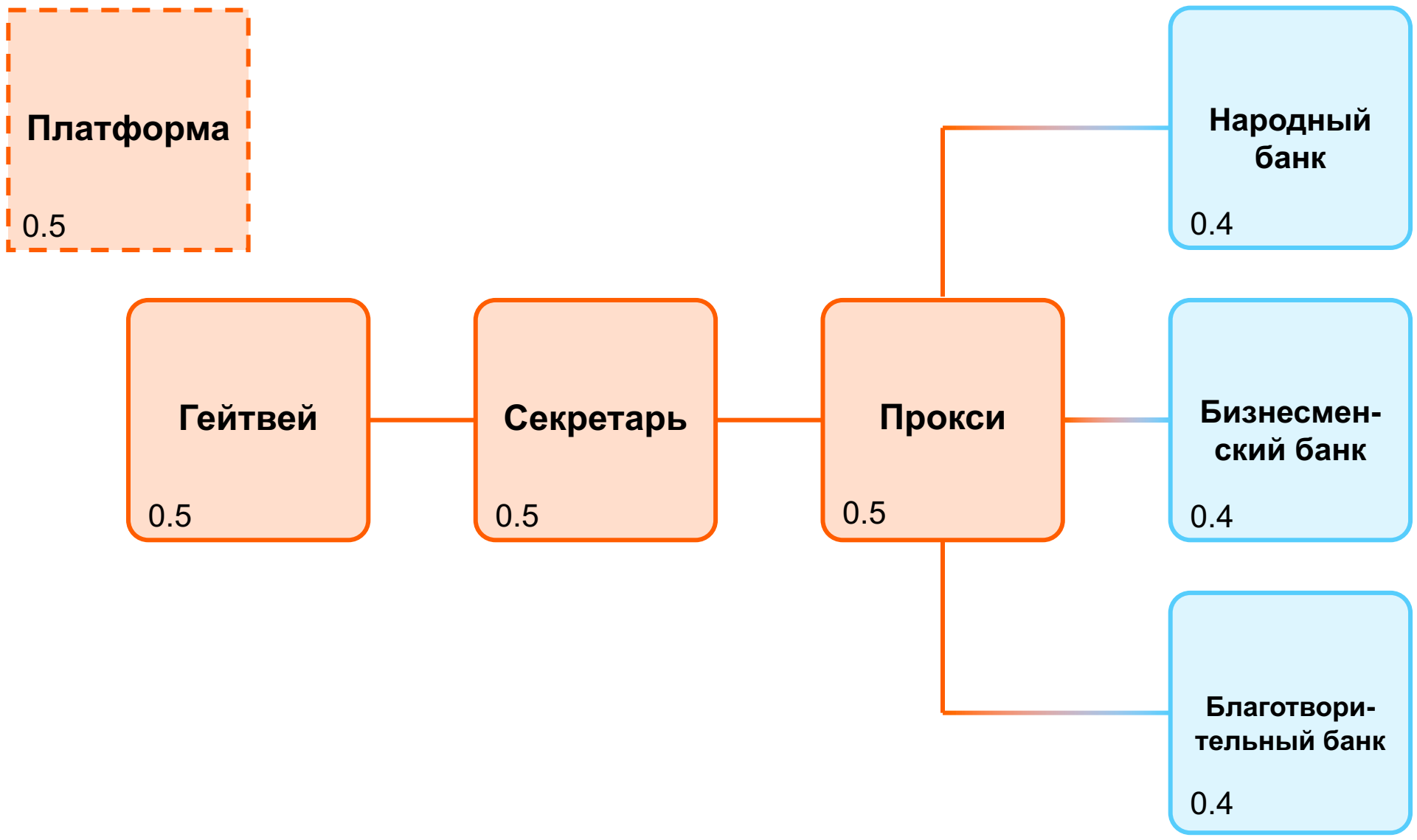


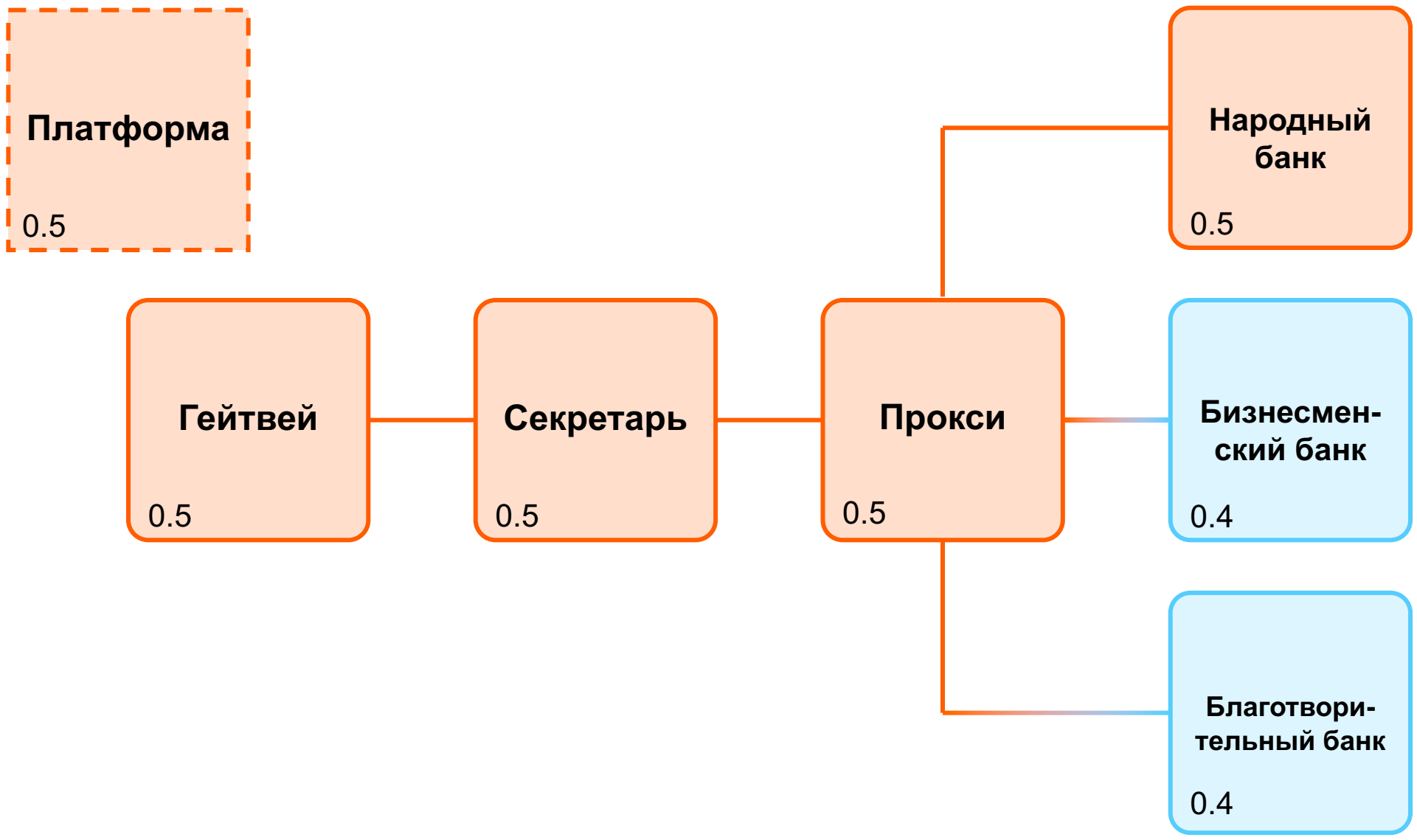


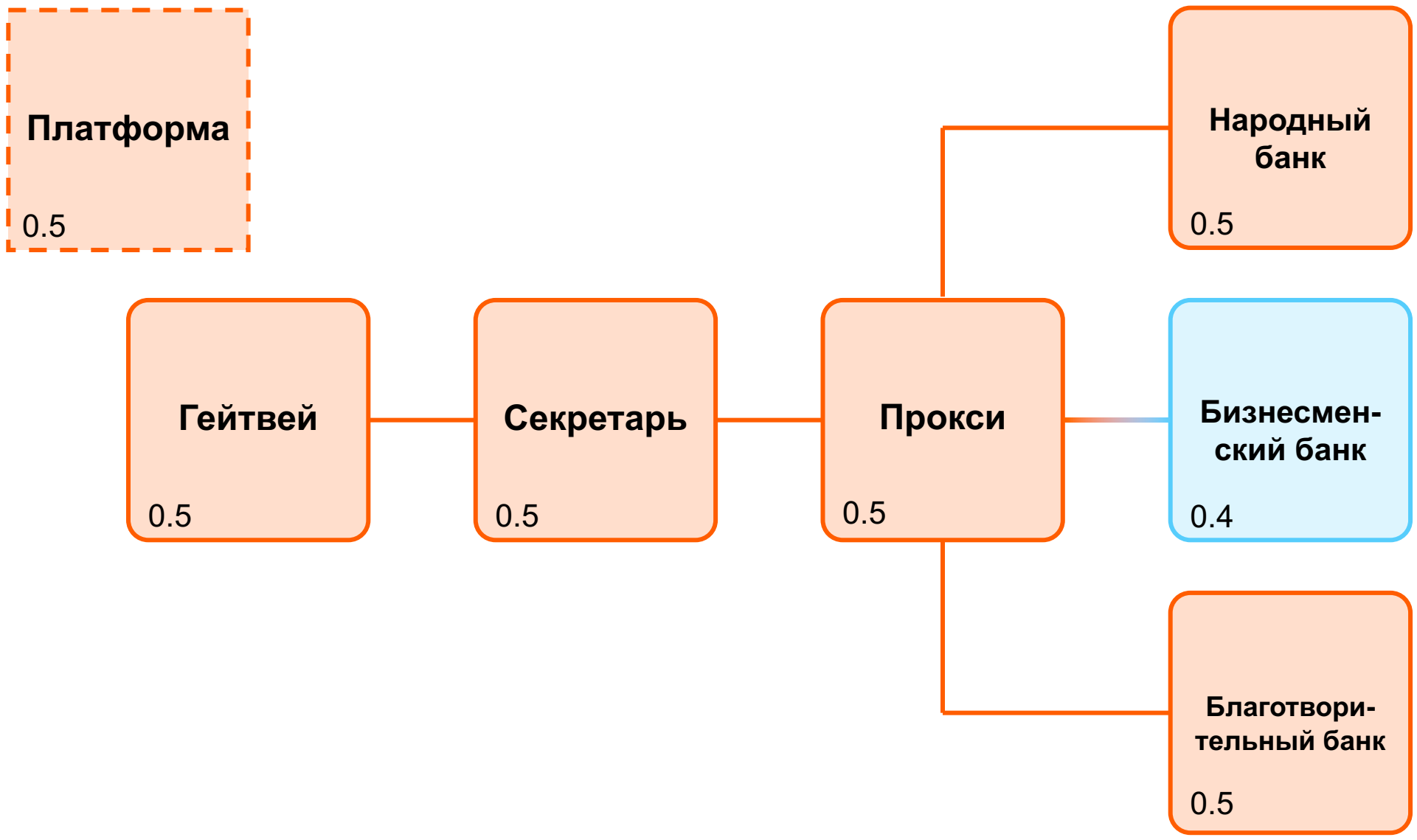


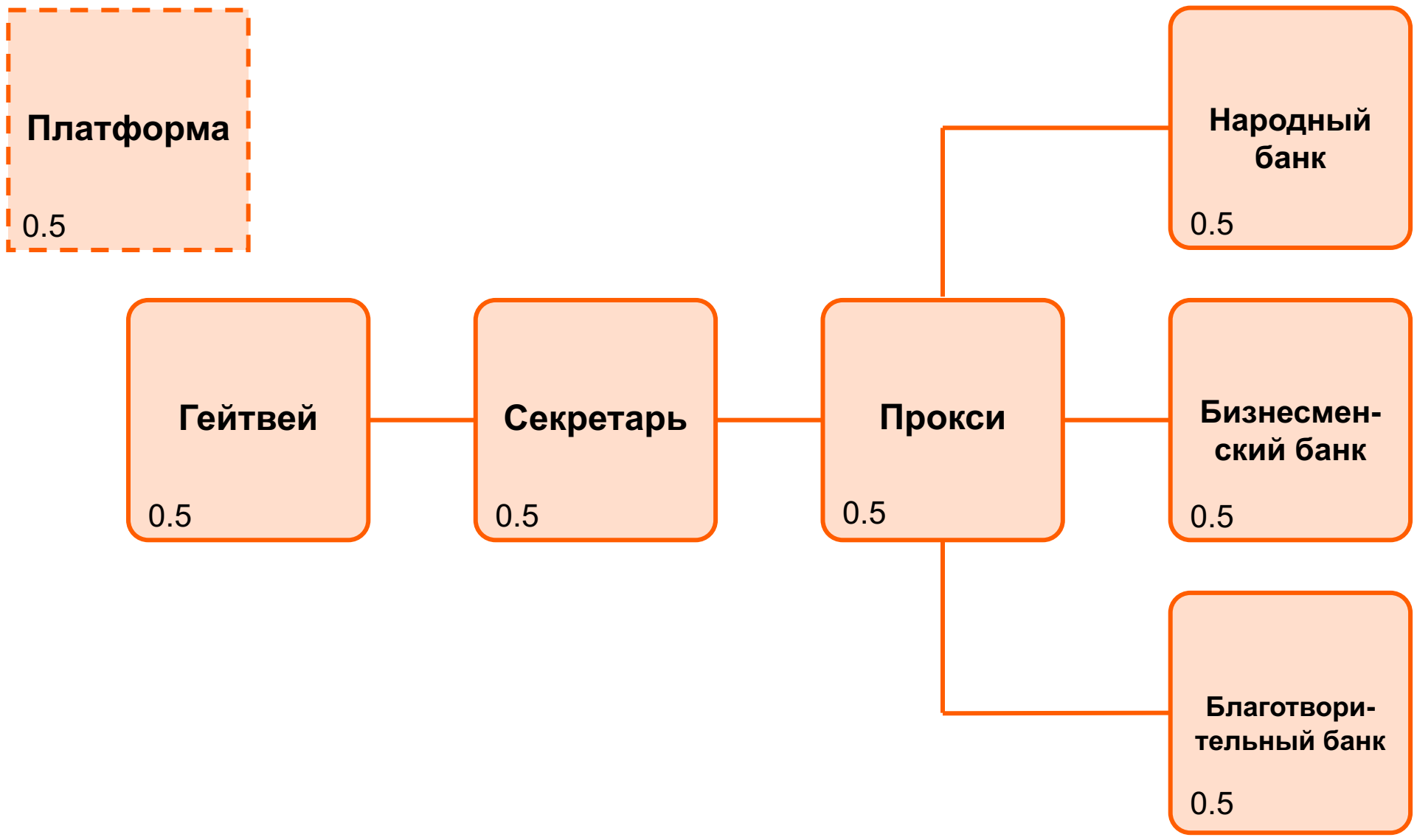


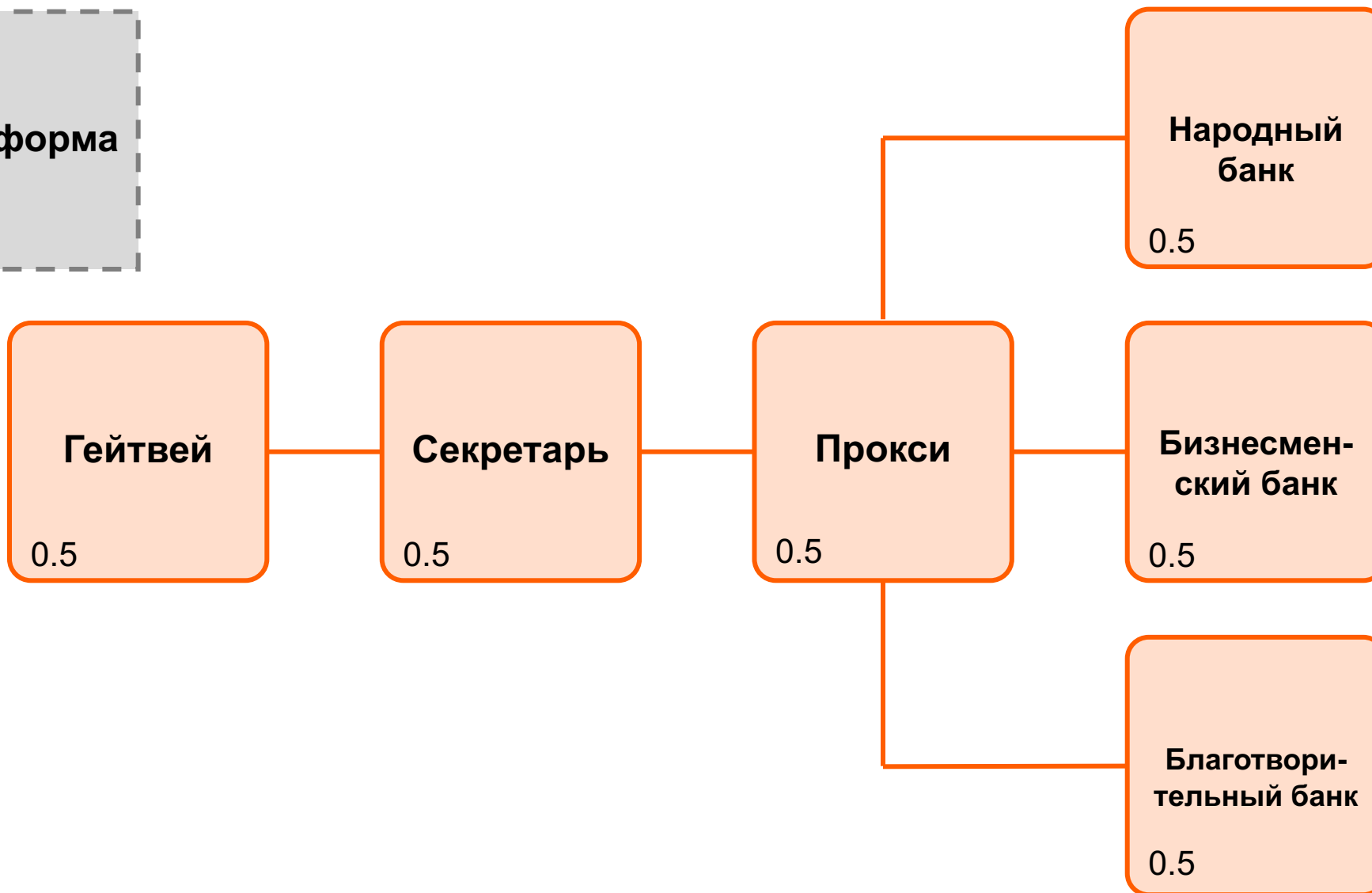












Платформа
0.5

Народный банк
0.5

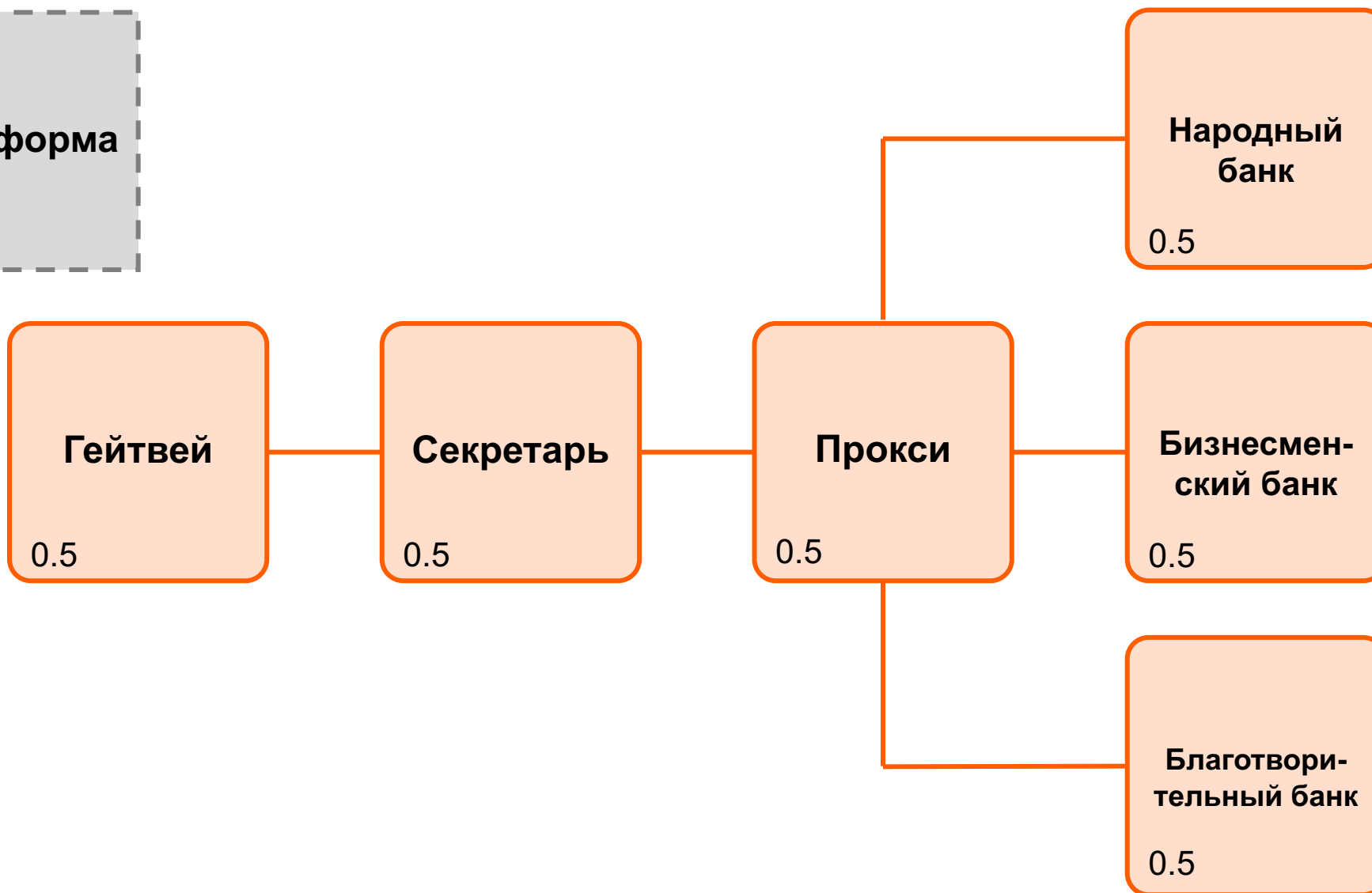
```
<artifactId>enum-demo</artifactId>  
<version>0.0.4</version>
```

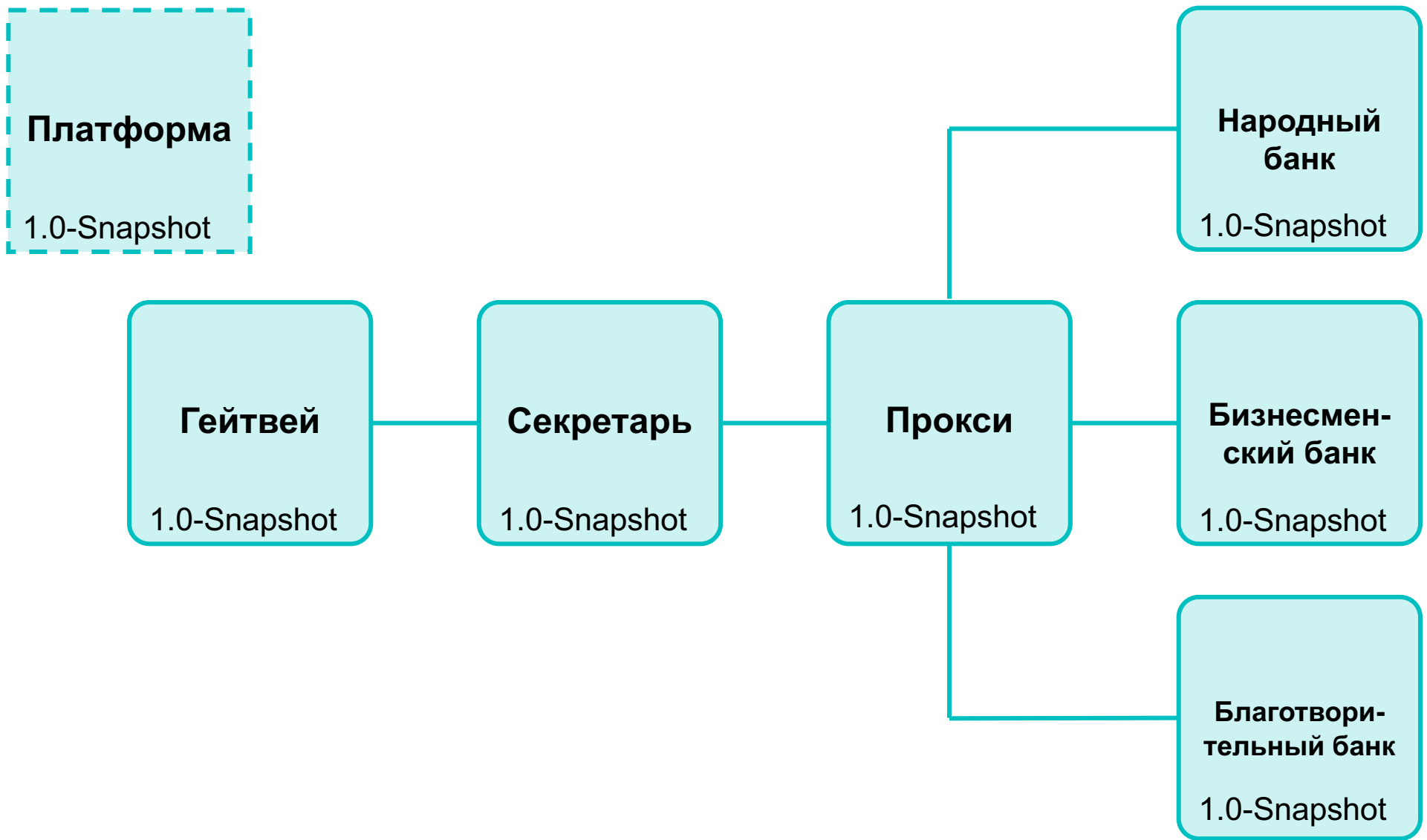
12 12
13 13
14 14

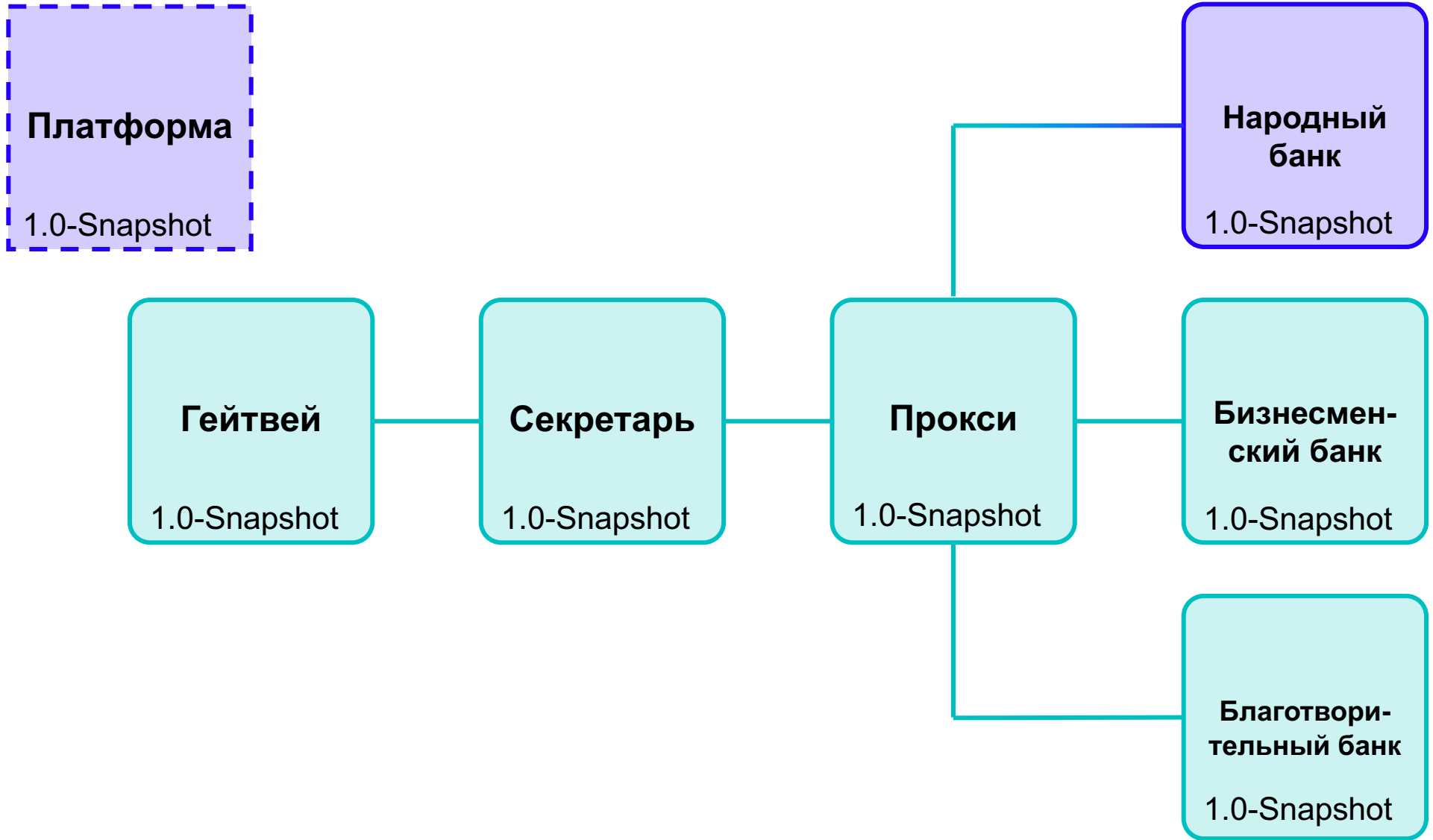
```
<artifactId>enum-demo</artifactId>  
<version>0.0.5</version>
```



Благотворительный банк
0.5



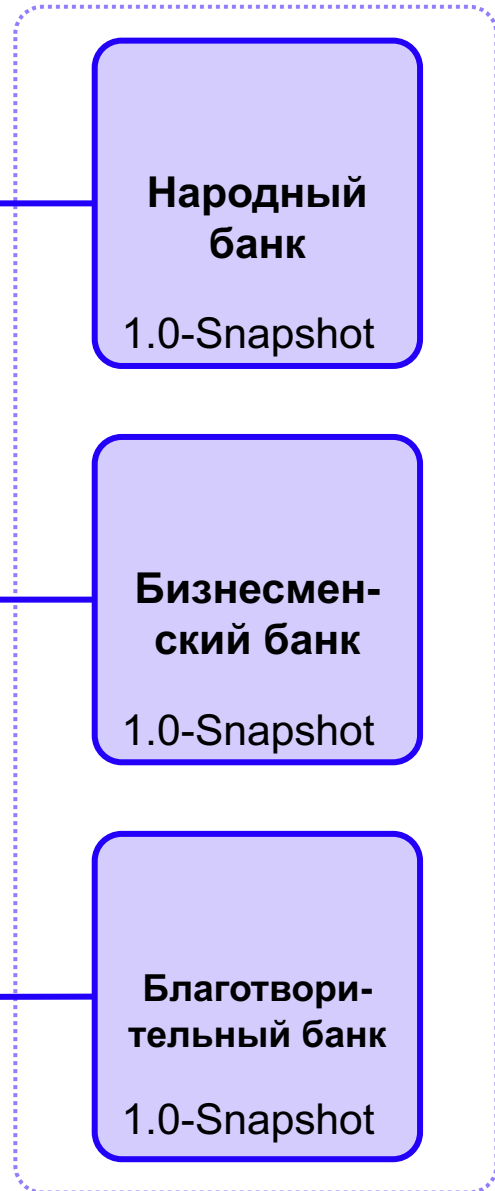
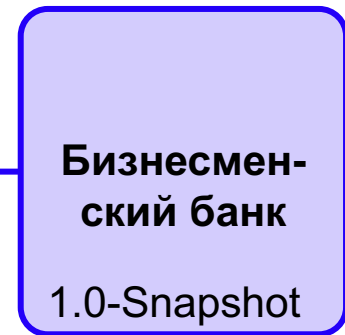
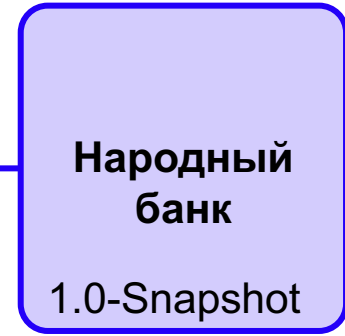
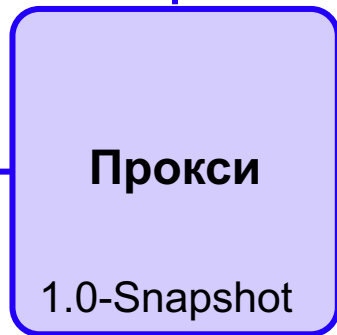
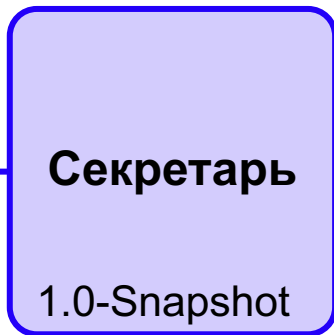


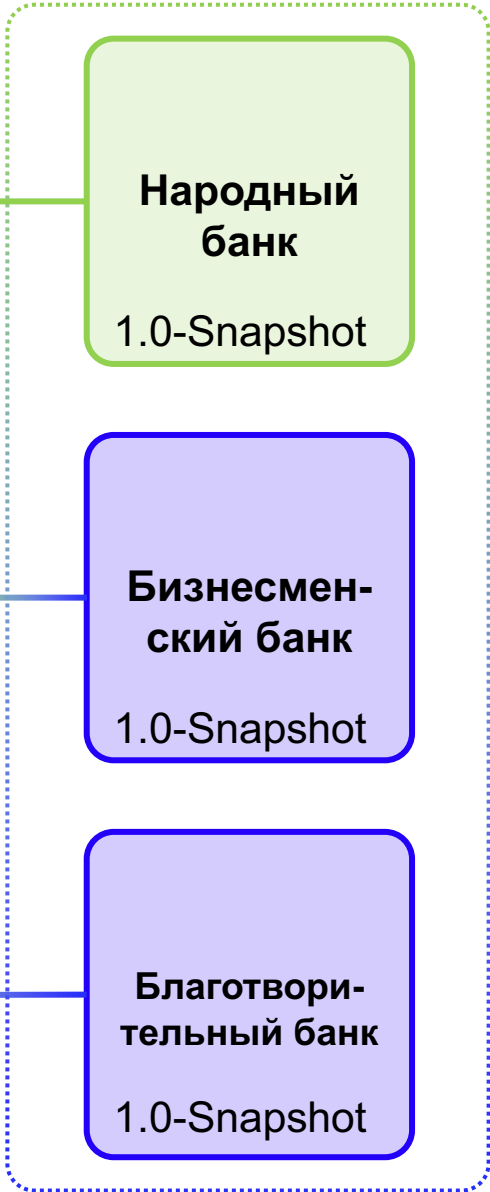
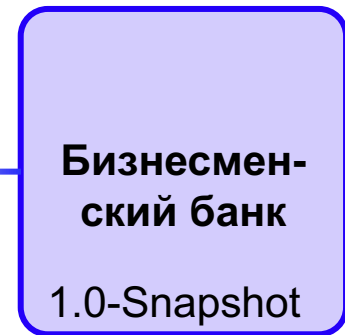



```
public enum ClientOccupation {  
    WORKER,  
    RETIRED,  
    STUDENT,  
    BUSINESSMAN,  
    UNEMPLOYED,  
    MEGA_BUSINESSMAN,  
    UNKNOWN,  
}
```

```
public String mapClientOccupation(  
    ClientOccupation occupation  
) {  
    switch (occupation) {  
        case RETIRED:  
            return "RETIRED";  
        case STUDENT:  
            return "STUDENT";  
        case UNKNOWN:  
            return "WORKER";  
        default:  
            return "WORKER";  
    }  
}
```

1.0-Snapshot







API курильщика

```
@Data
public class SimpleRequest {

    private Occupation occupation;

    private Gender gender;

}
```

```
public enum Occupation {

    WORKER, RETIRED;

}
```

API здорового человека

```
@Data
public class SimpleRequest {

    private Occupation occupation;

    private Gender gender;

}
```

```
@EnumApi(enumClass = PreparedOccupation.class)
public interface Occupation {

}
```

```
public enum PreparedOccupation implements Occupation {

    WORKER, RETIRED;

}
```

Выводы



Enum в API - зло

Каждое добавление константы каскадно влияет на всю систему



Enum можно передавать, но не надо принимать

Относитесь либерально к тому, что принимаете из сети и консервативно к тому, что передаёте в сеть



Строки — не идеальная альтернатива

Если объявить тип поля как строку, туда можно будет передать что угодно. Также придётся долго править уже существующий код.



Jackson и Spring позволяют сделать Rest api без недостатков enum

Мы объявим тип поля интерфейсом, а enum — одна из его реализаций.

Спасибо!