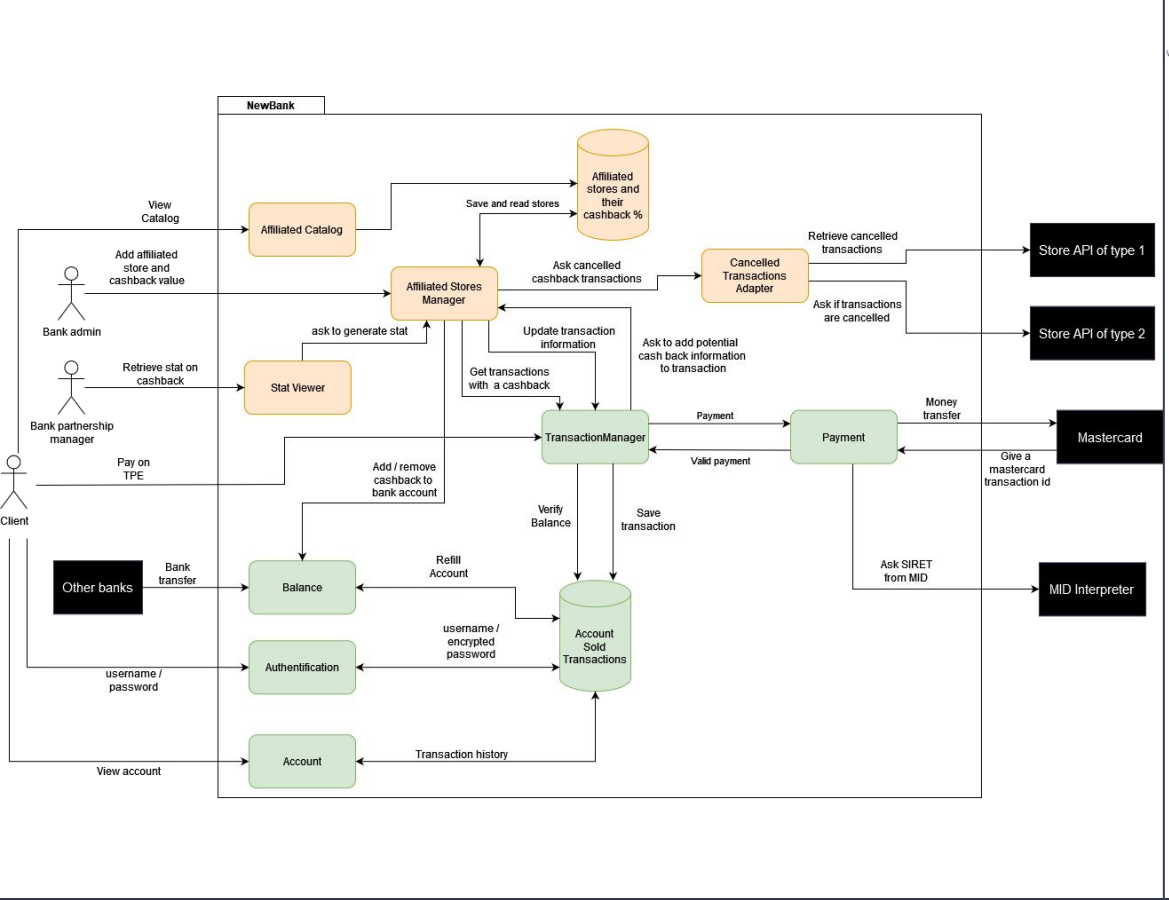
Architecture Logicielle

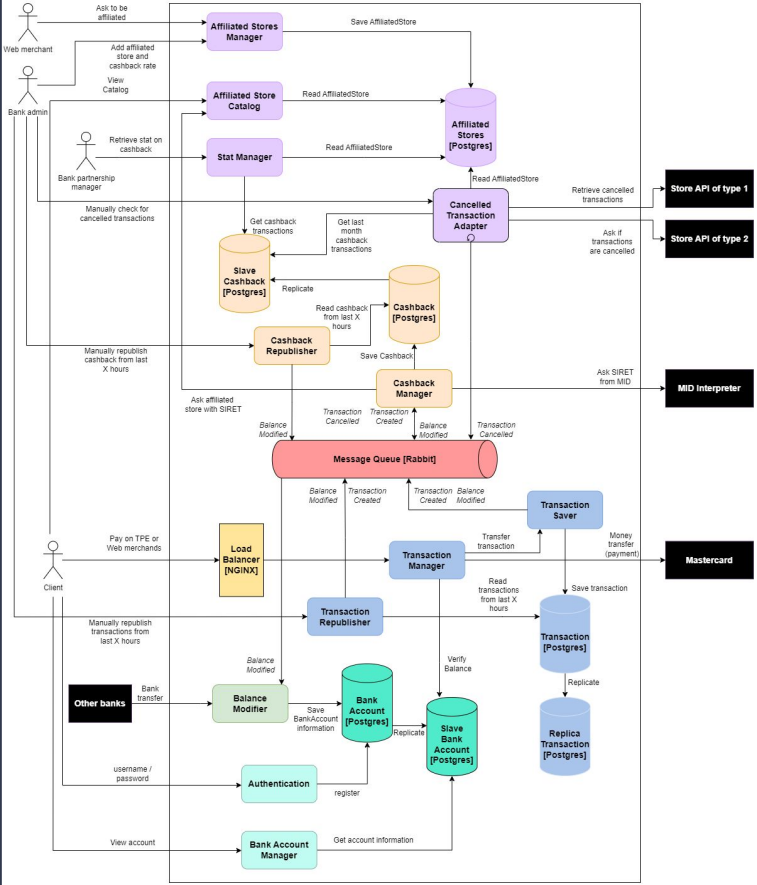
Ouverture à des petits marchands Web et au marché international

- Scalabilité : beaucoup d'opérations avec cashbacks
- Résilience : charge haute constante 24/24

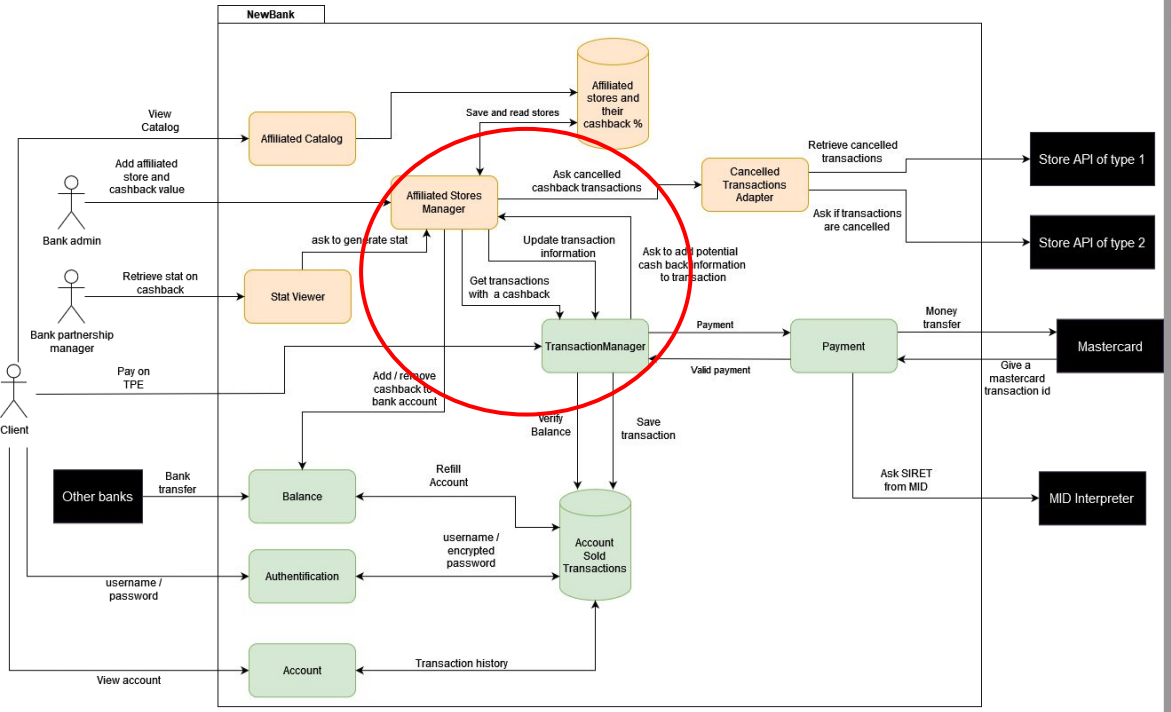
NewBank - V8 - Cashback

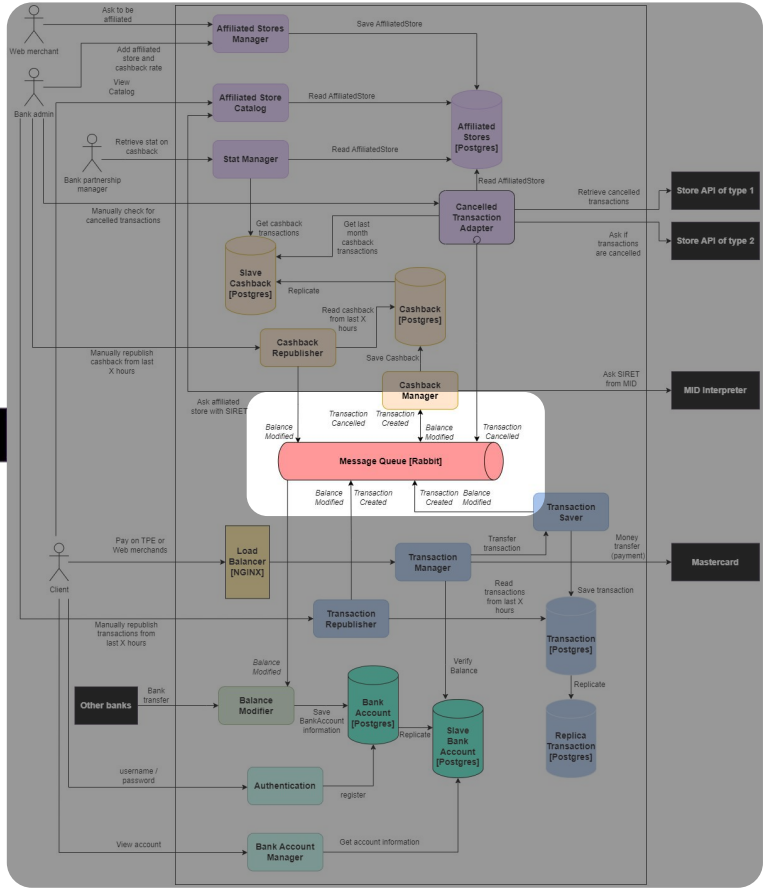
Antoine BUQUET - Benoit GAUDET - Ayoub IMAMI - Mourad KARRAKCHOU



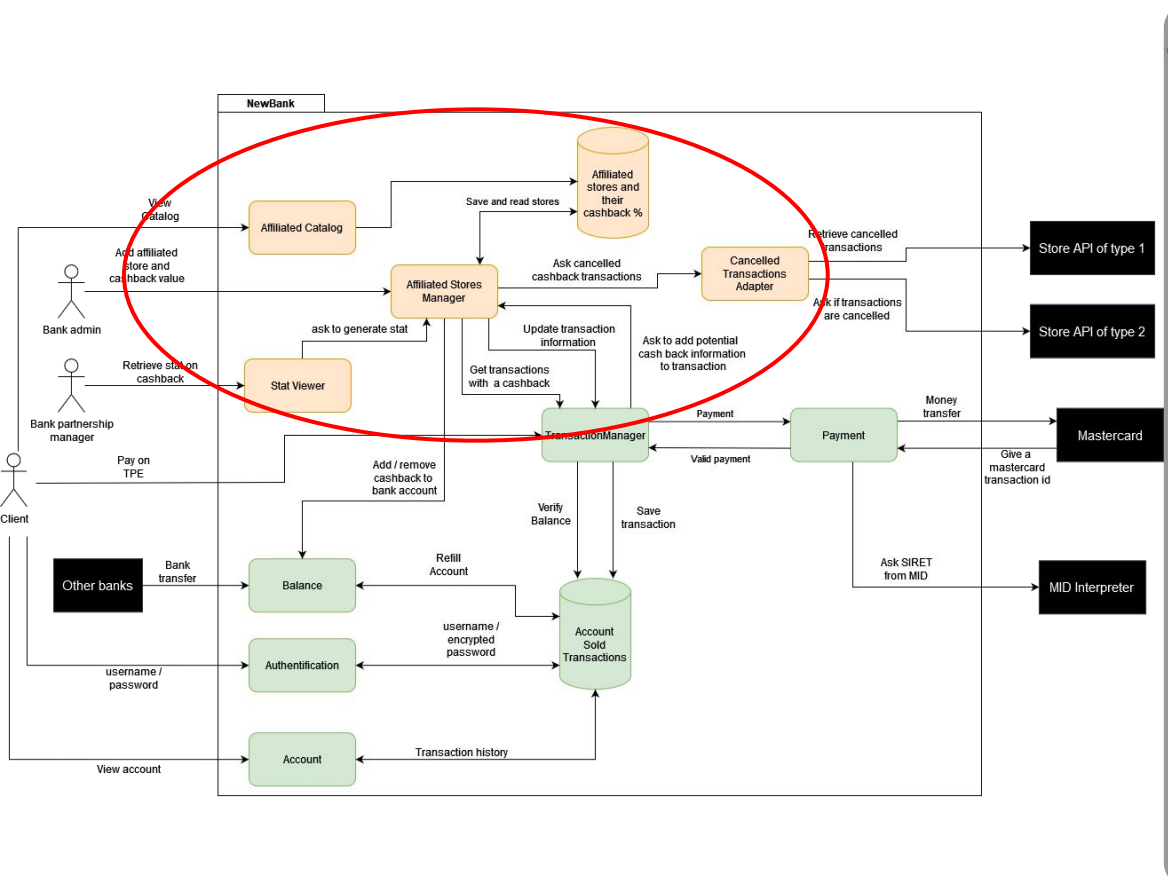


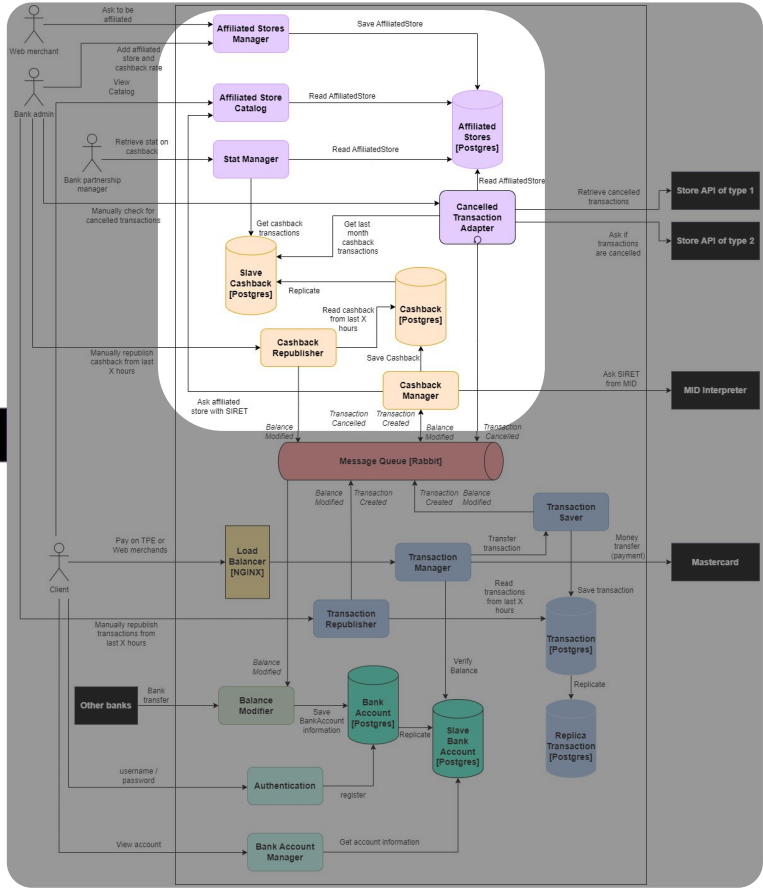
Diagrammes d’architecture précédent et actuel



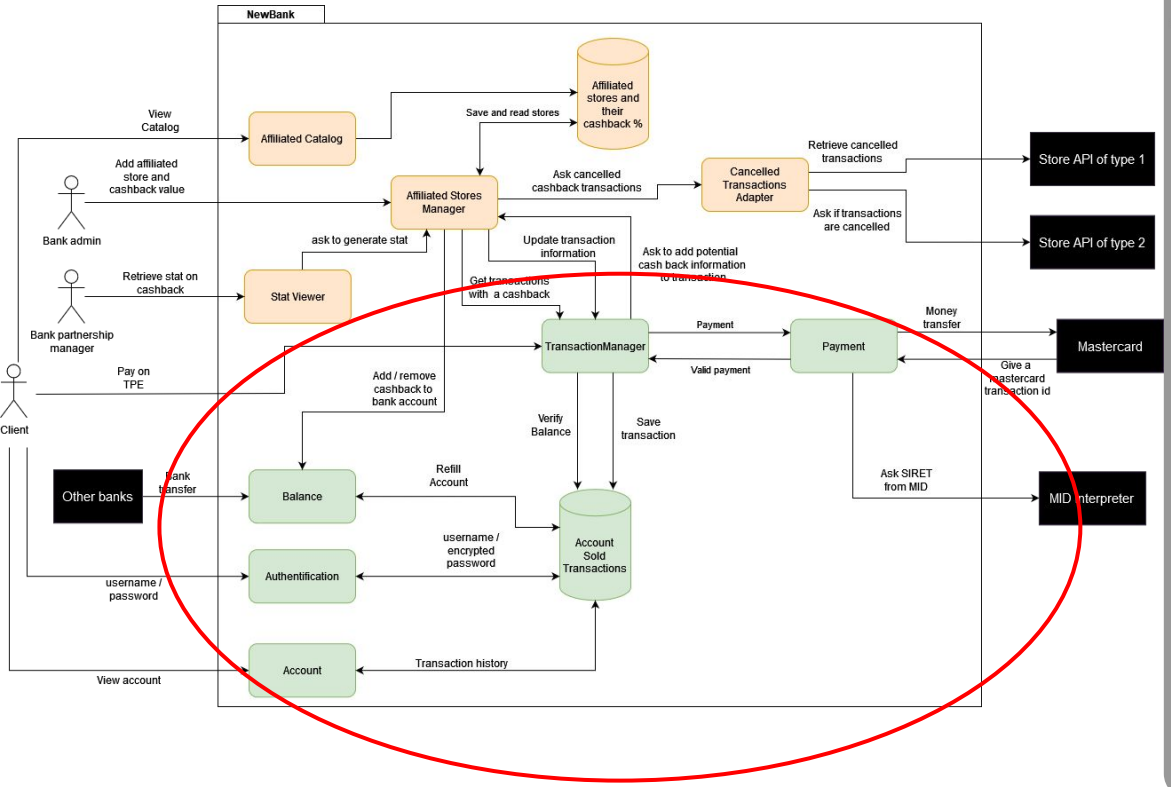


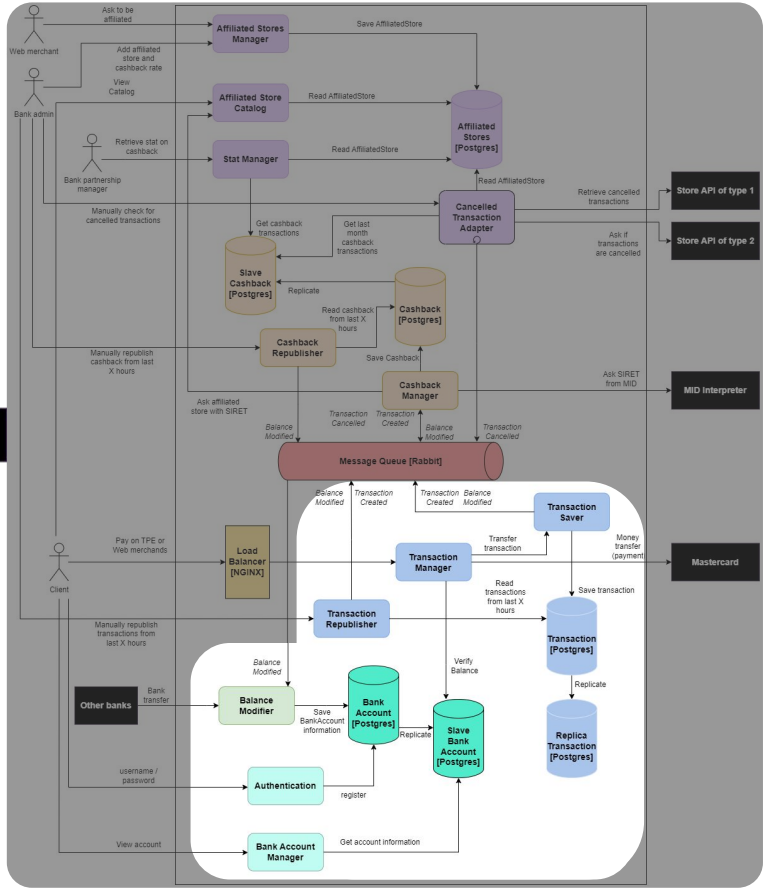
Implémentation d'une Message Queue RabbitMQ



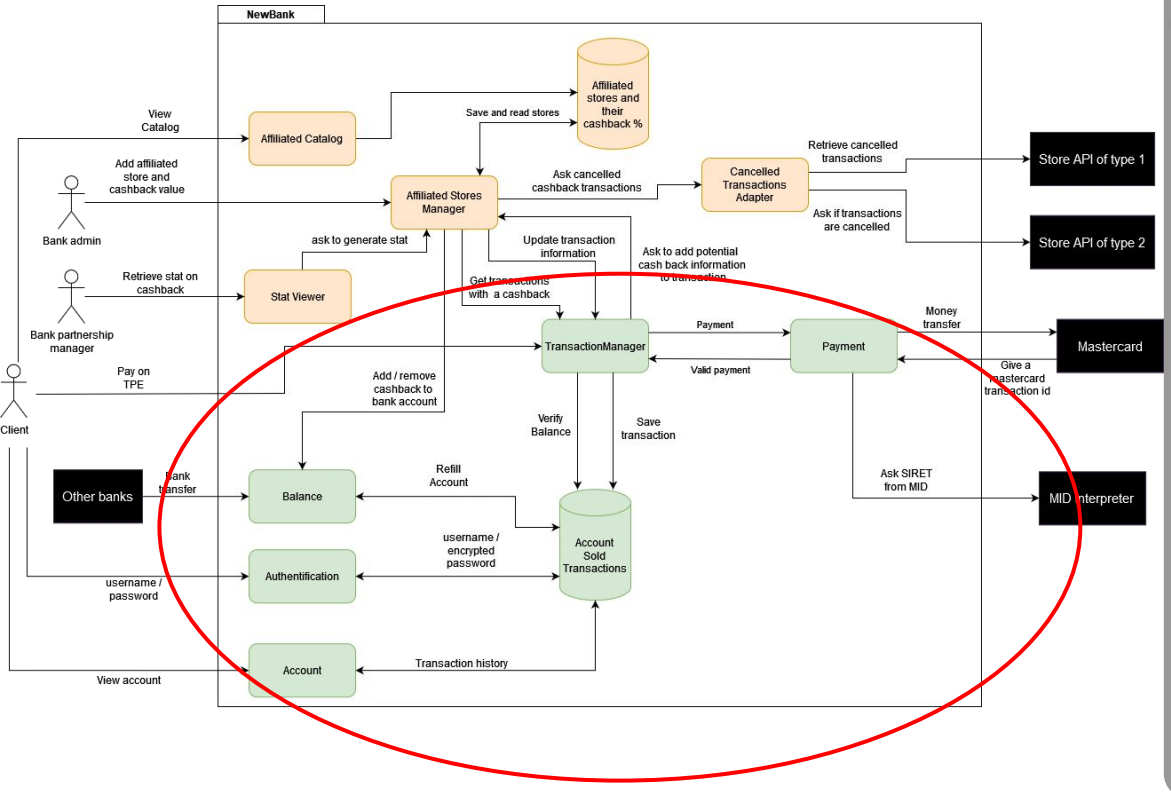


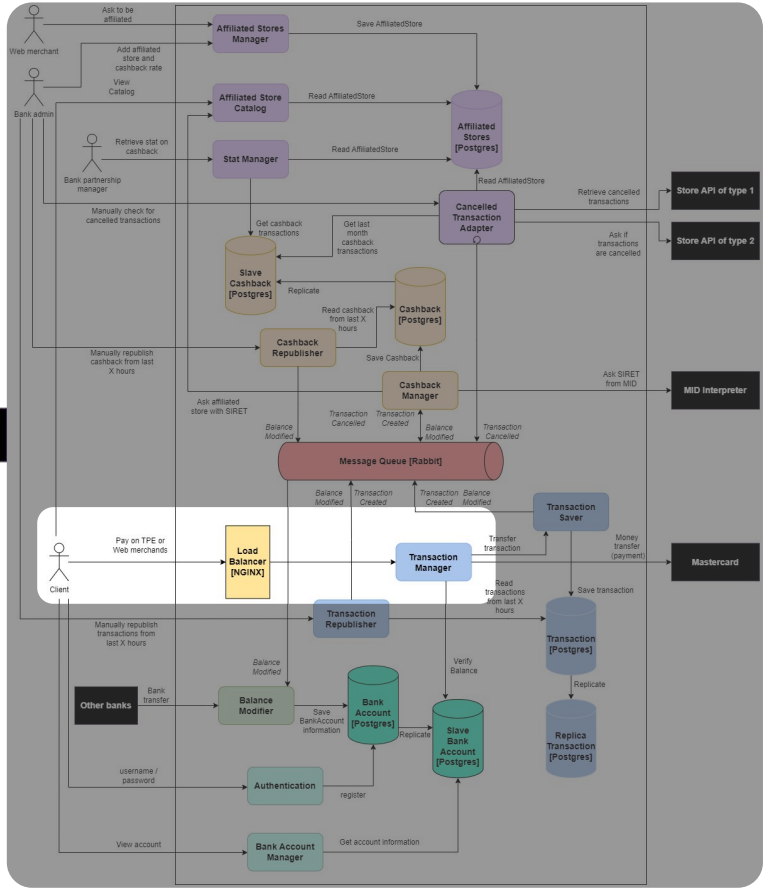
Séparations des responsabilités



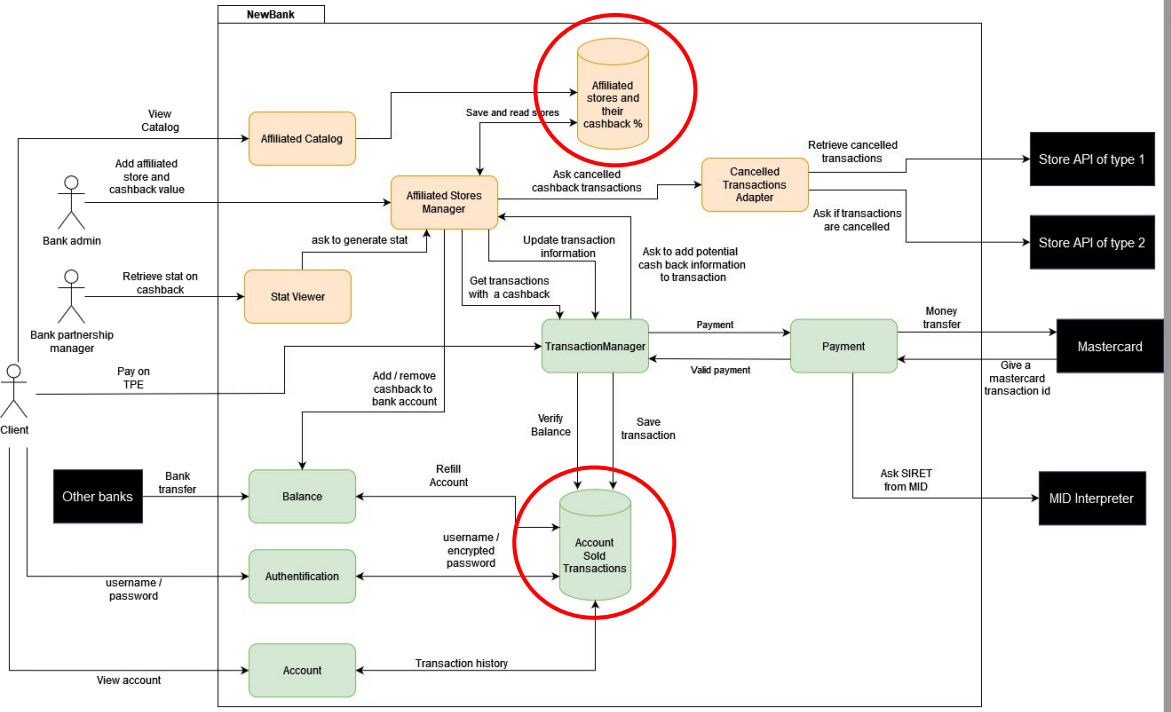


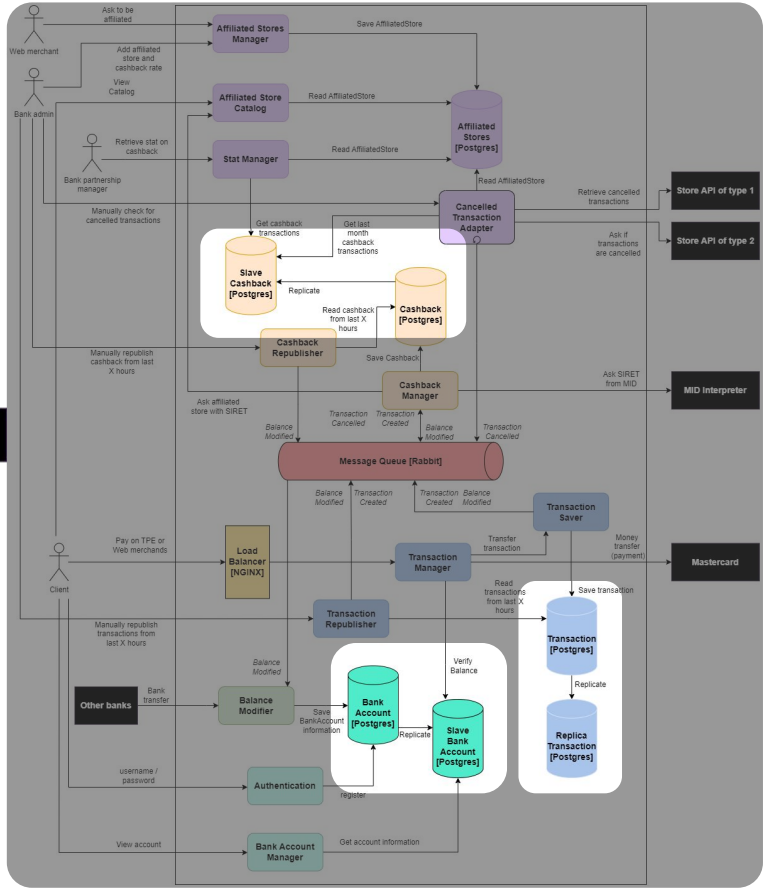
Séparations des responsabilités



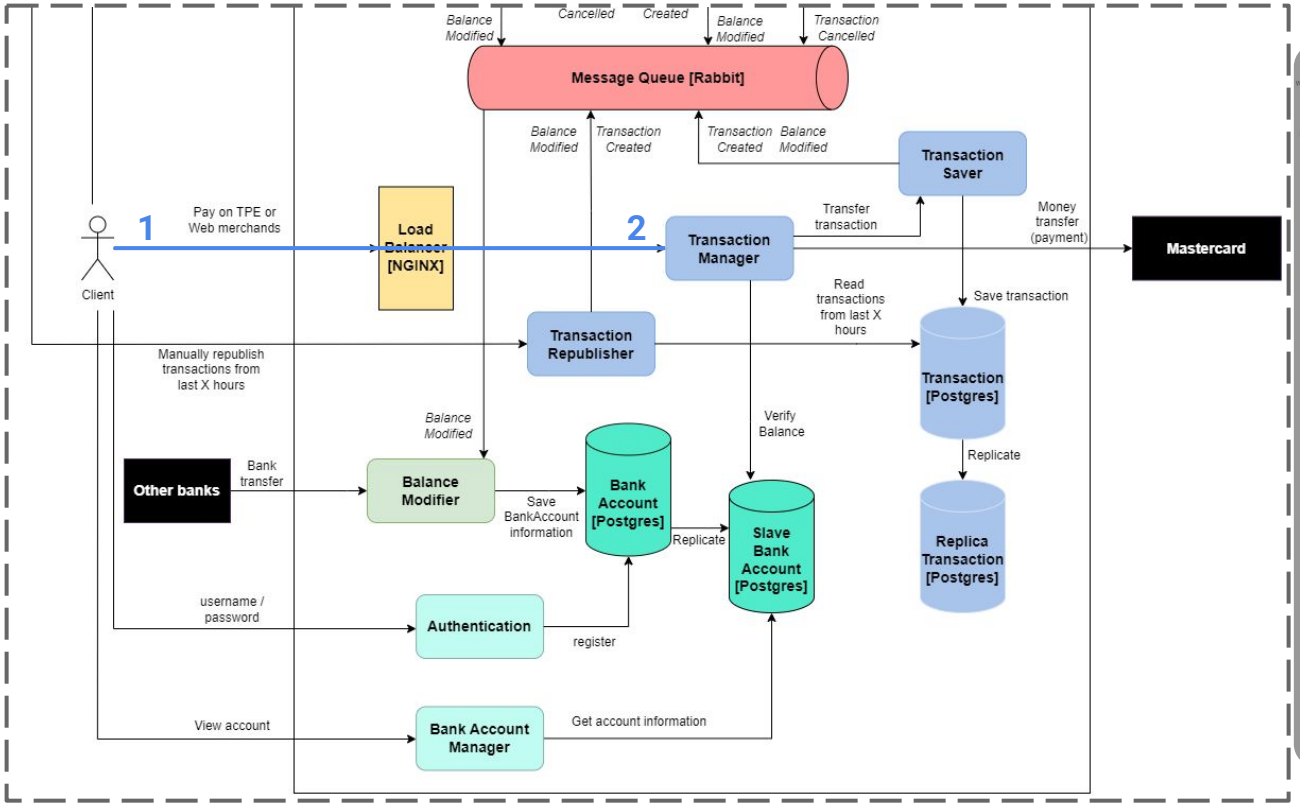


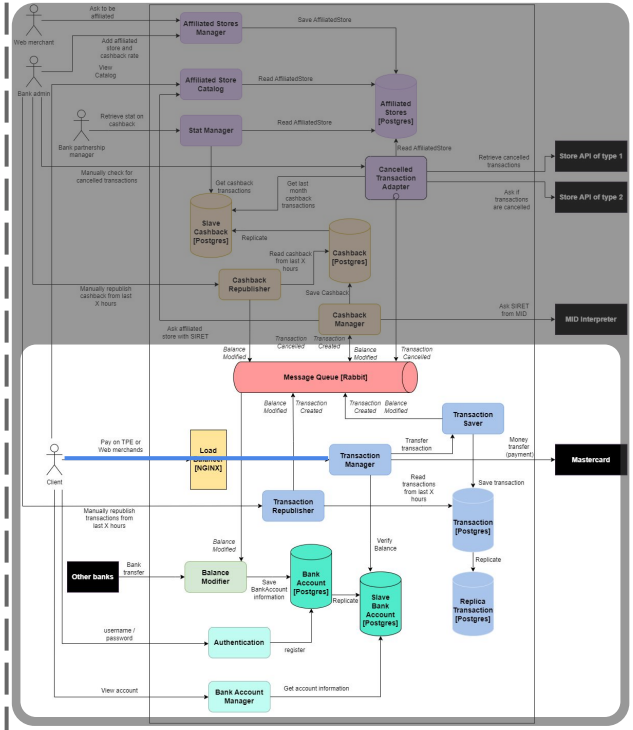
Implémentation d'un Load Balancer



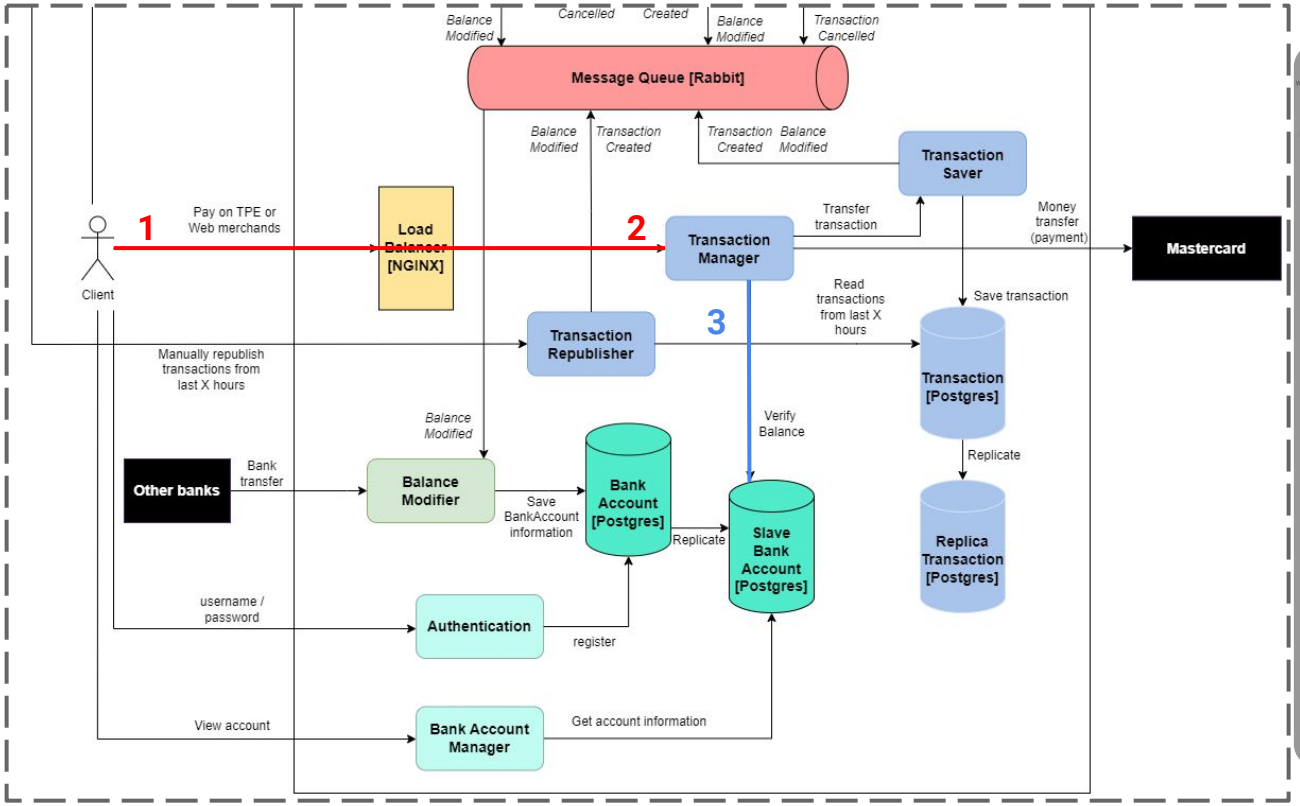


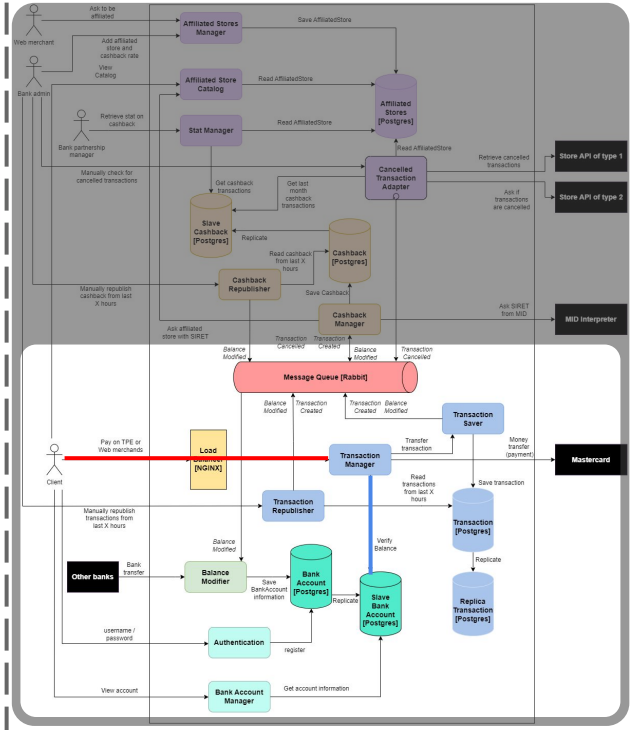
Implémentation du pattern Master-Slave



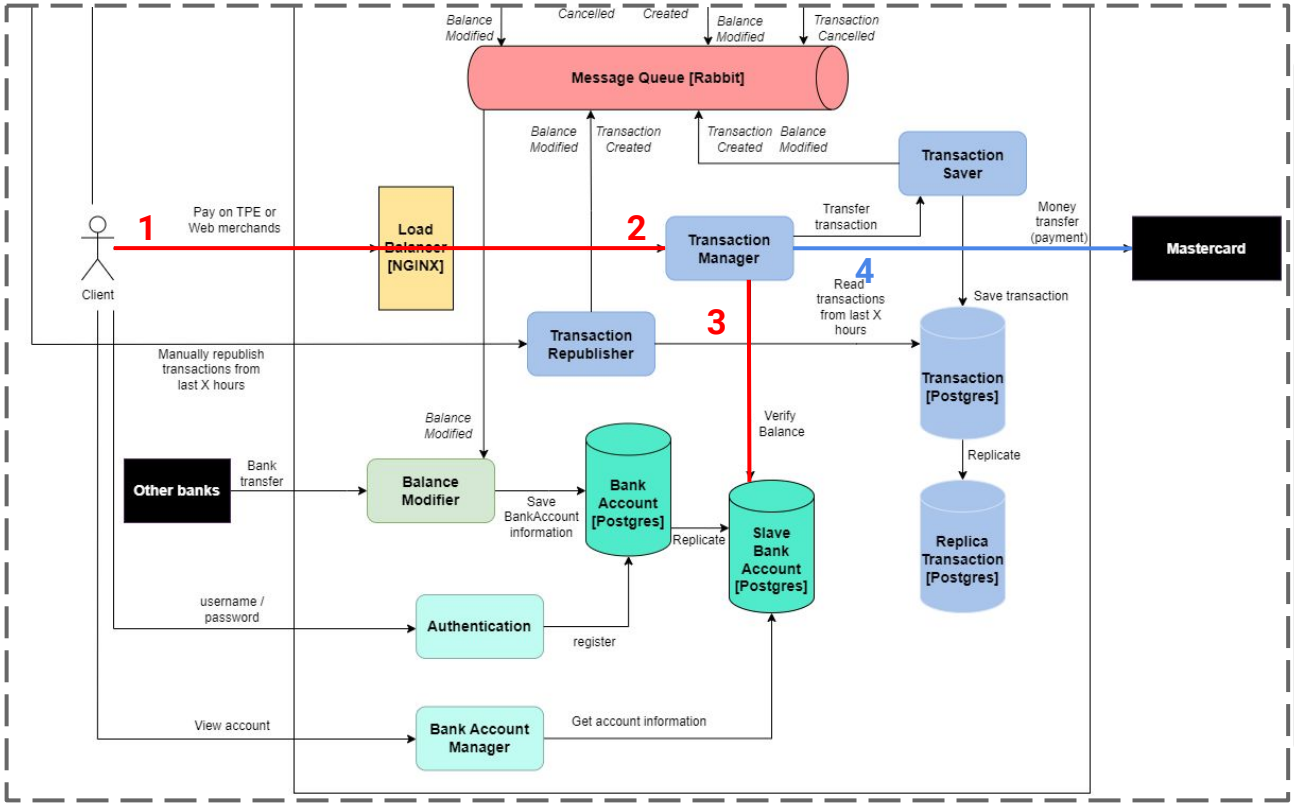


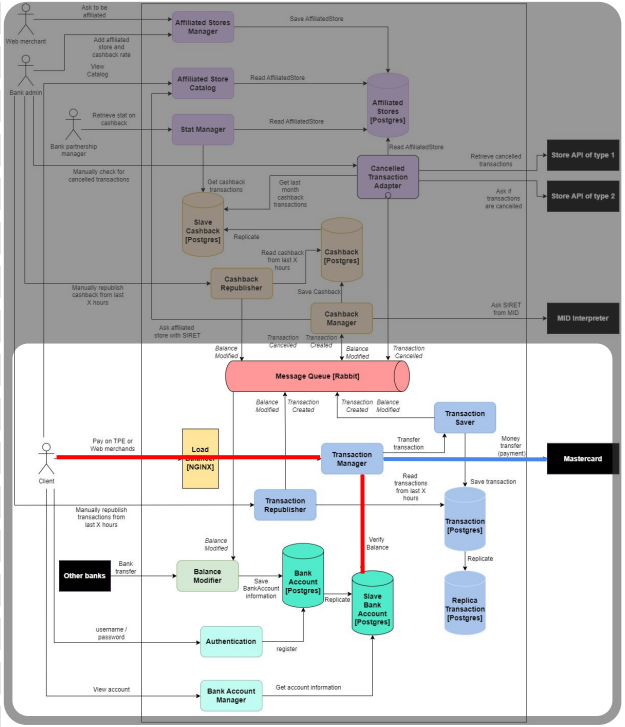
Scénario - Paiement avec Cashback



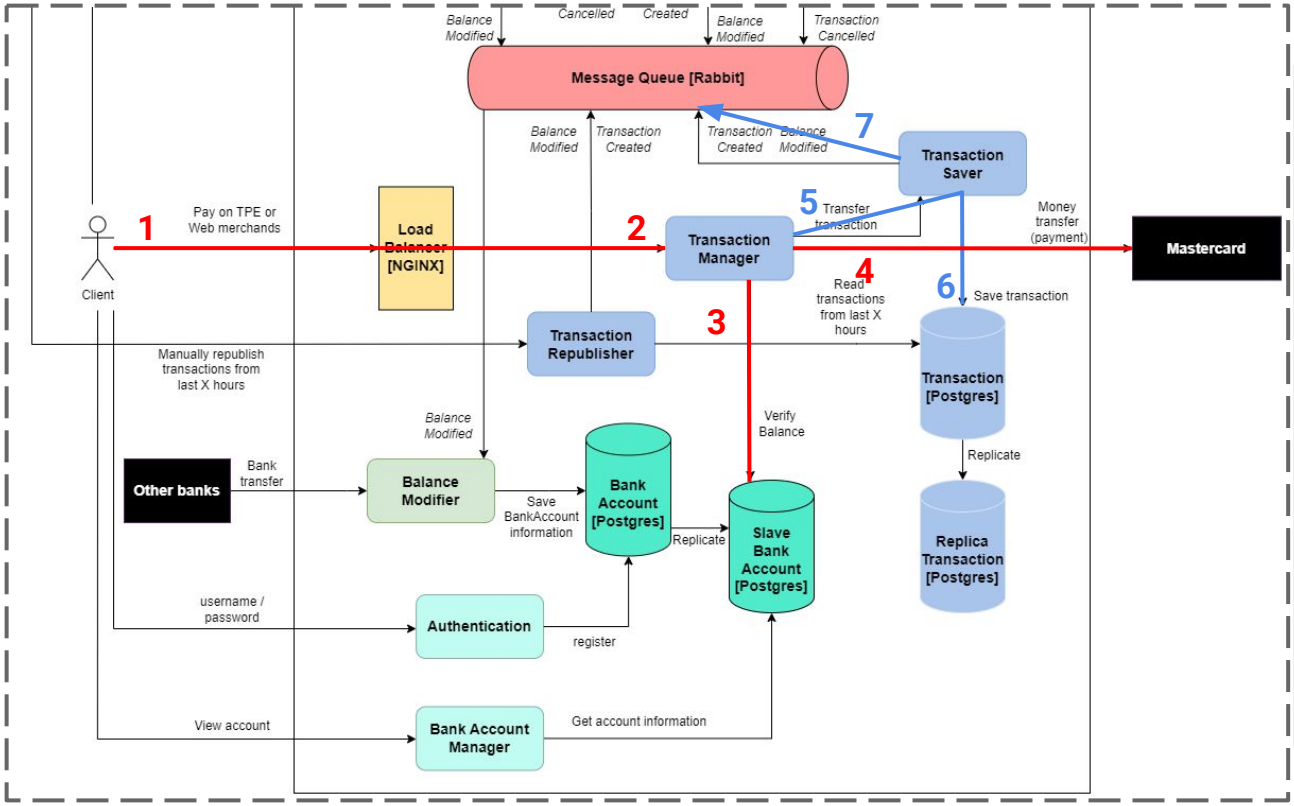


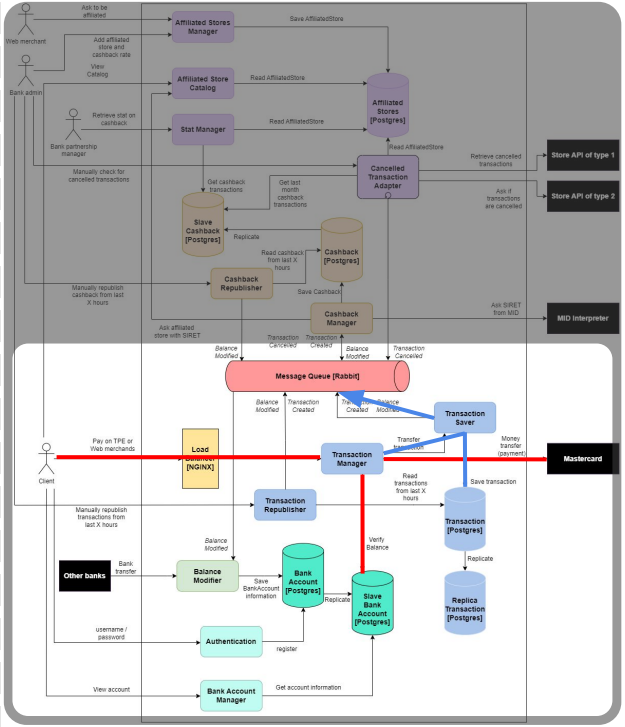
Scénario - Paiement avec Cashback



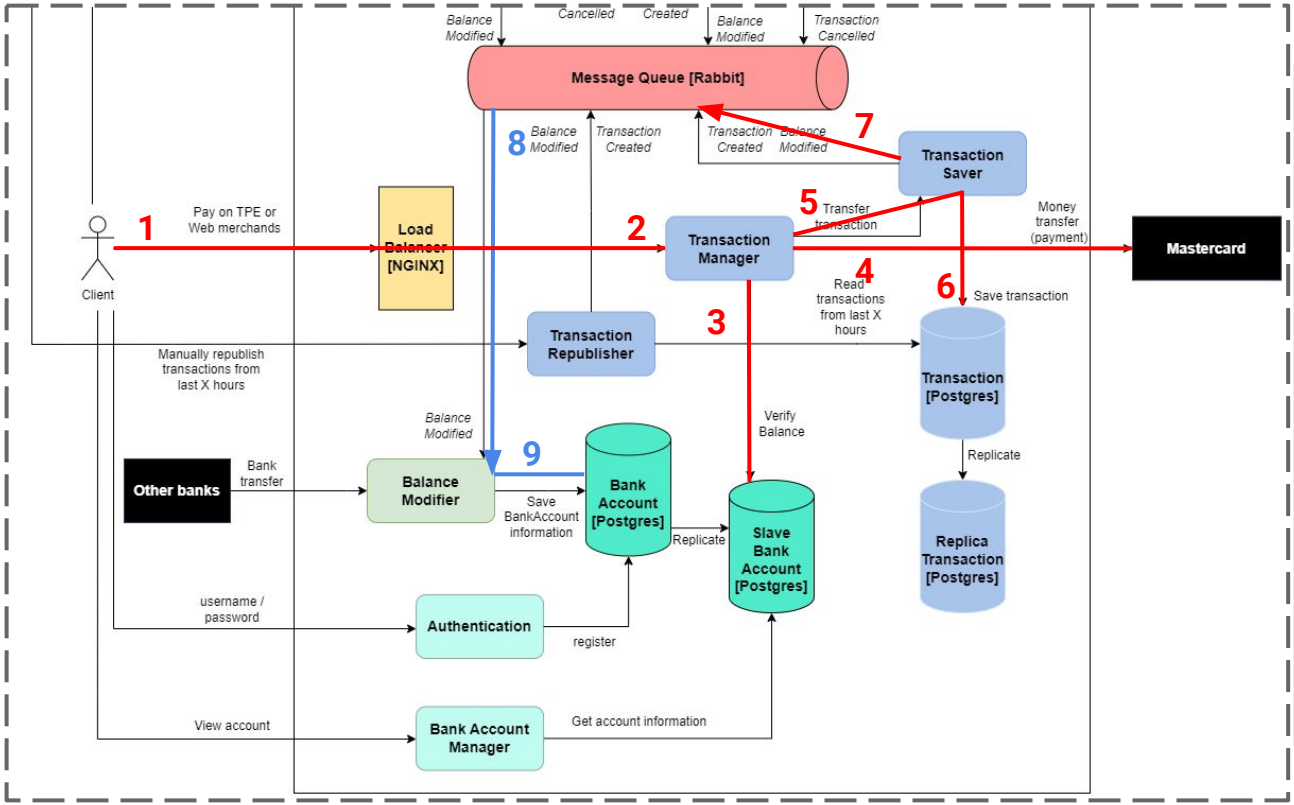


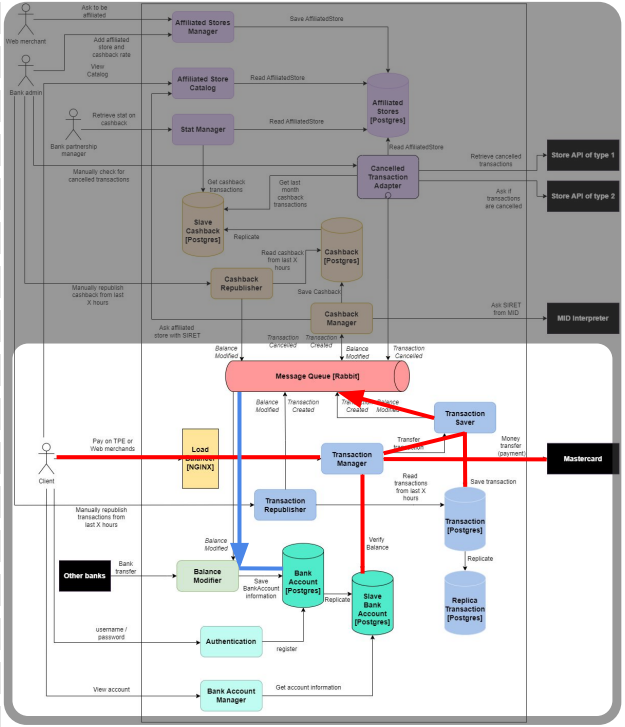
Scénario - Paiement avec Cashback



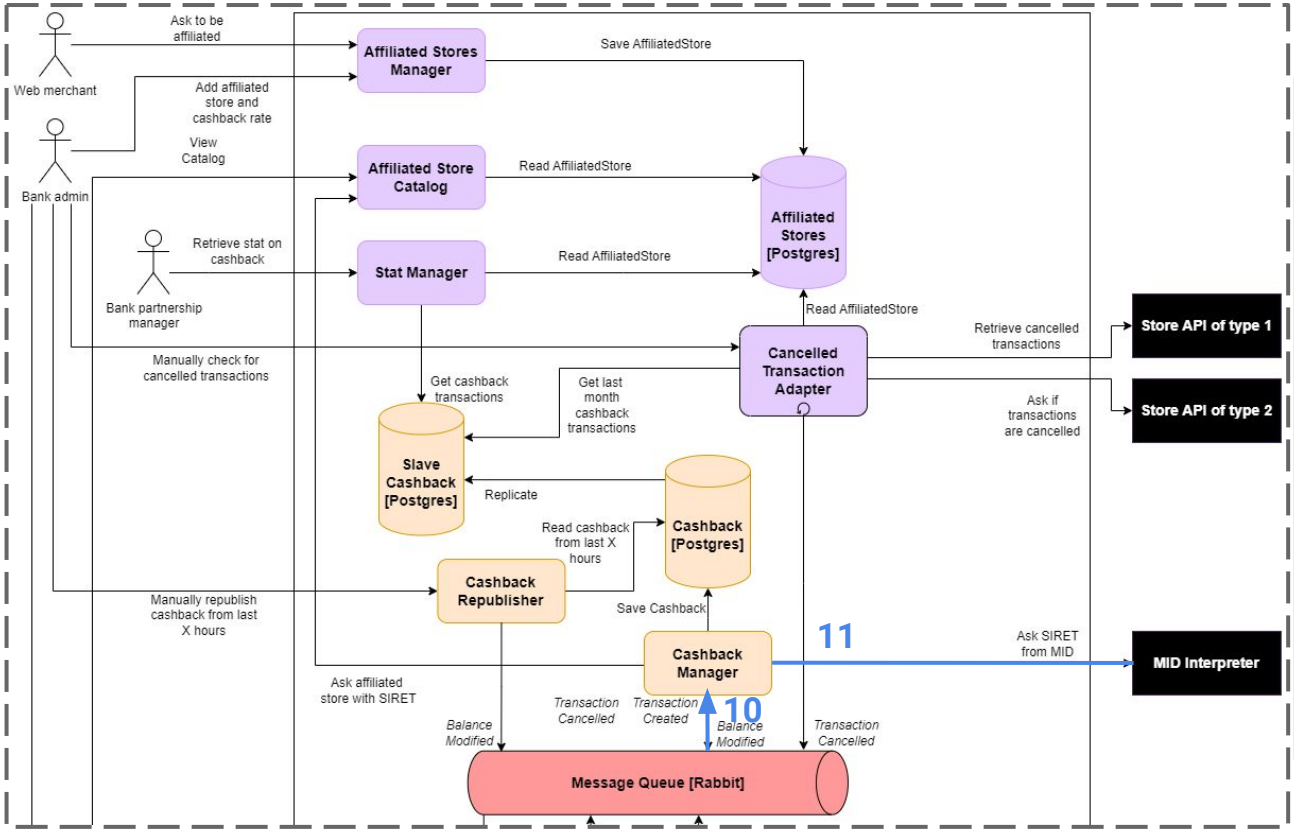


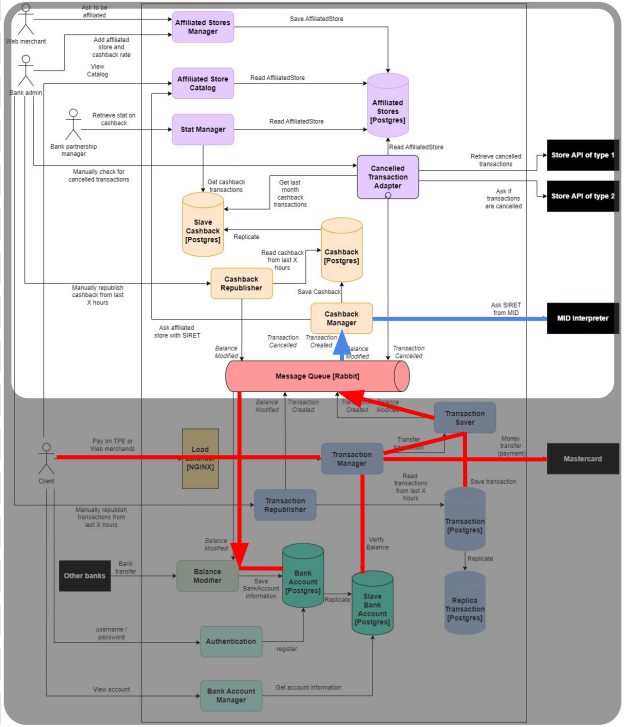
Scénario - Paiement avec Cashback



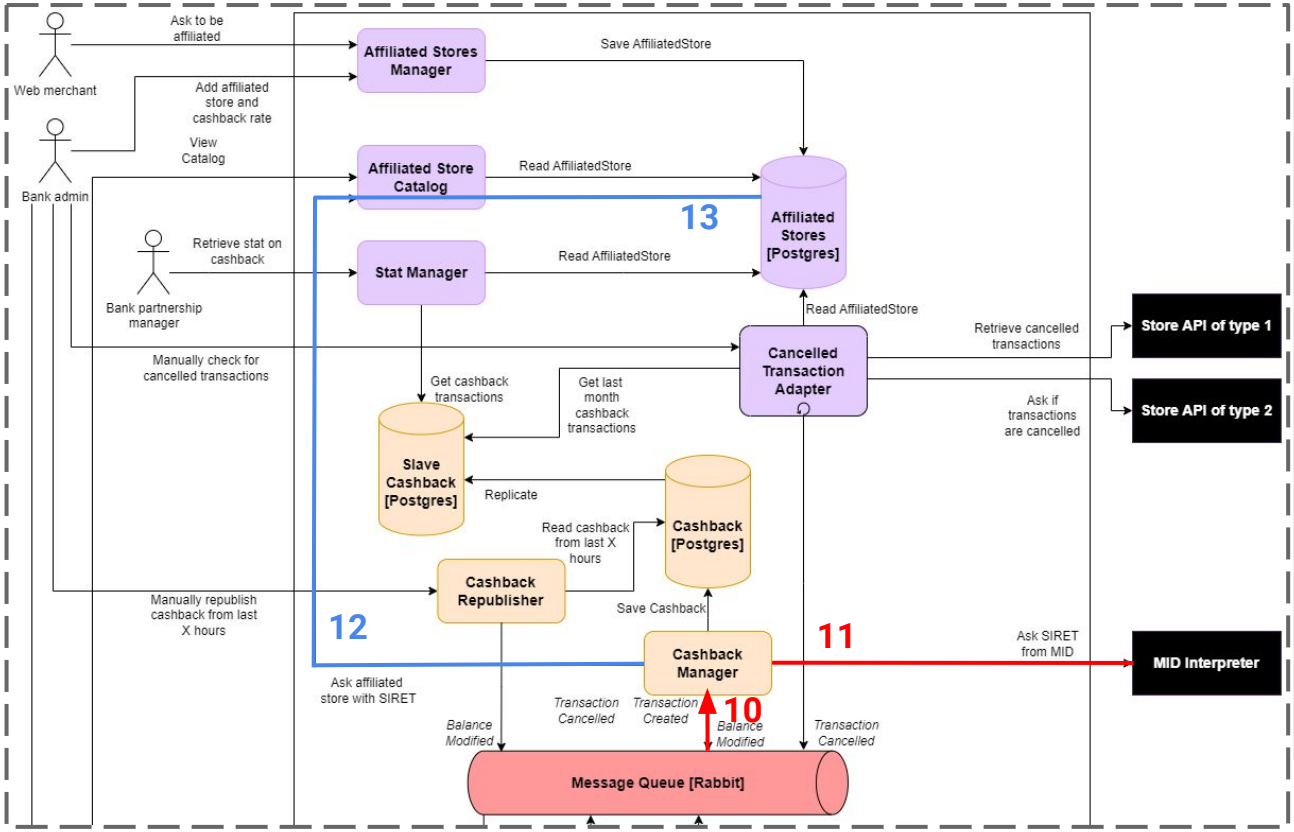


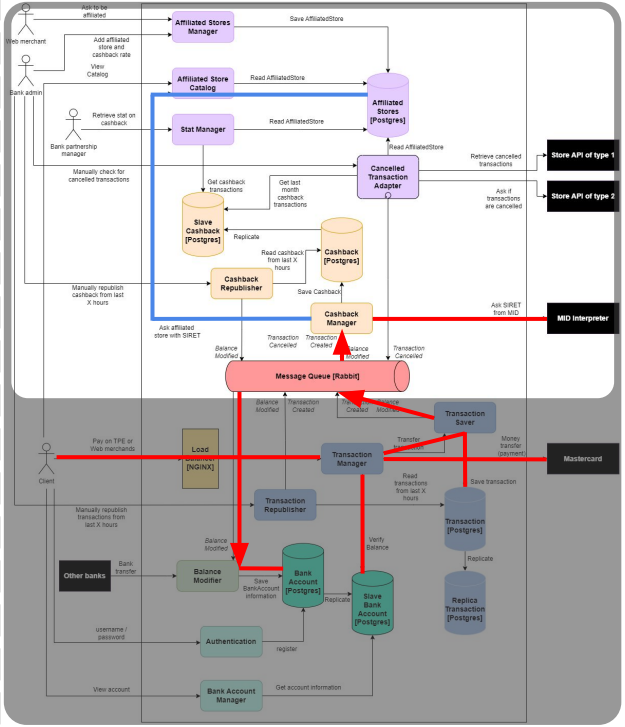
Scénario - Paiement avec Cashback



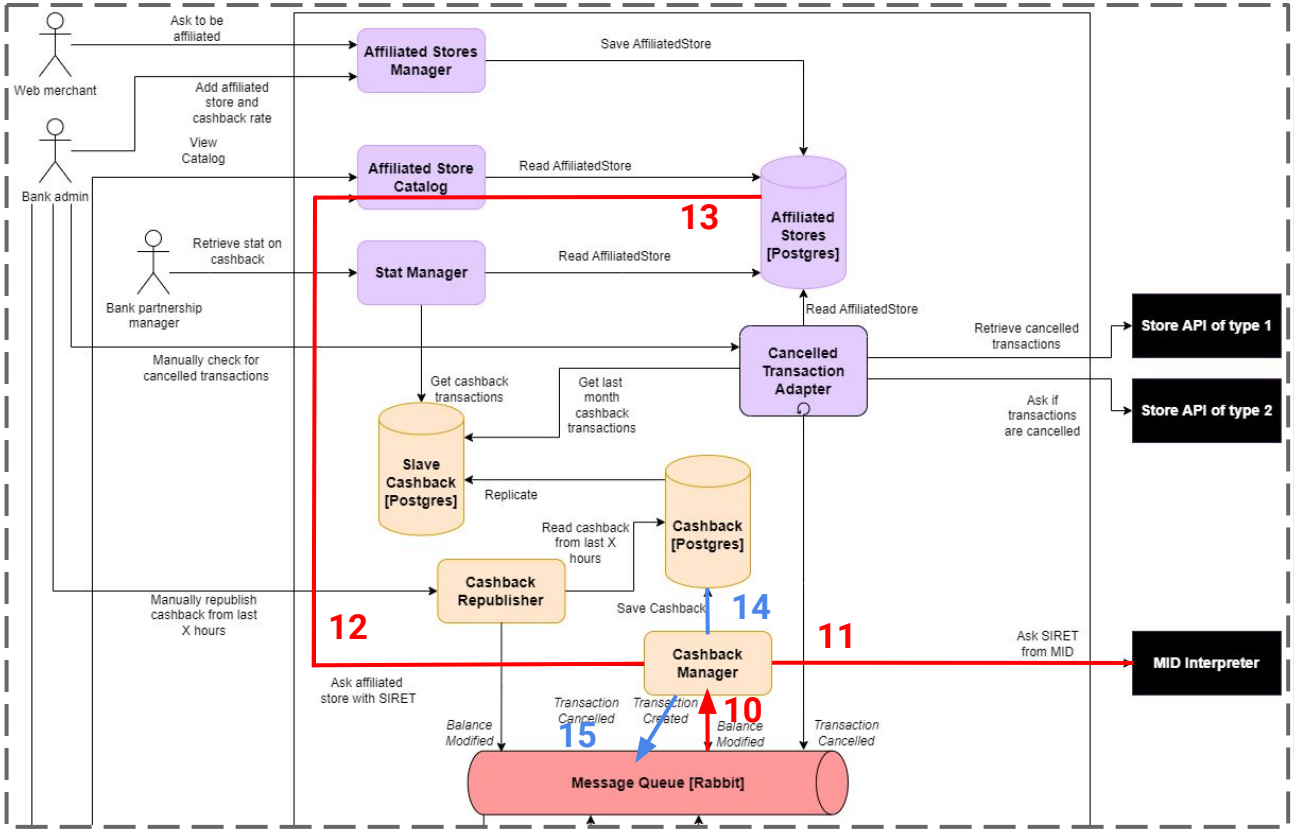


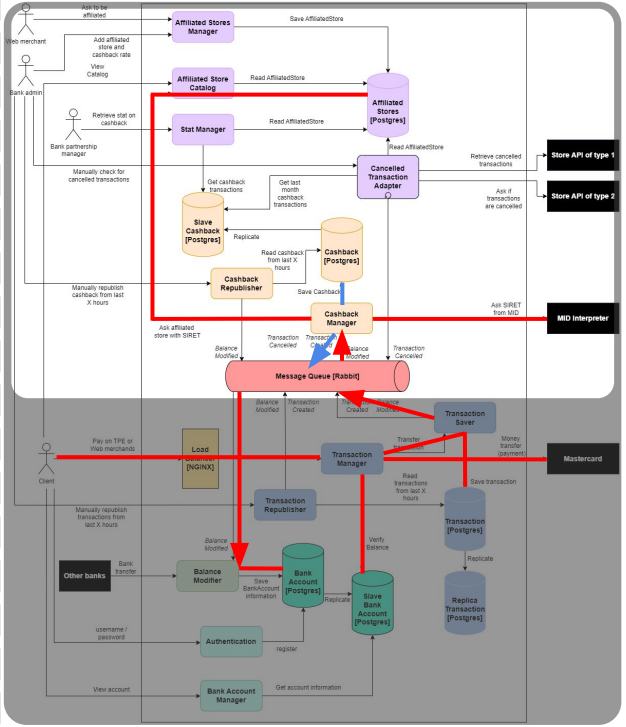
Scénario - Paiement avec Cashback



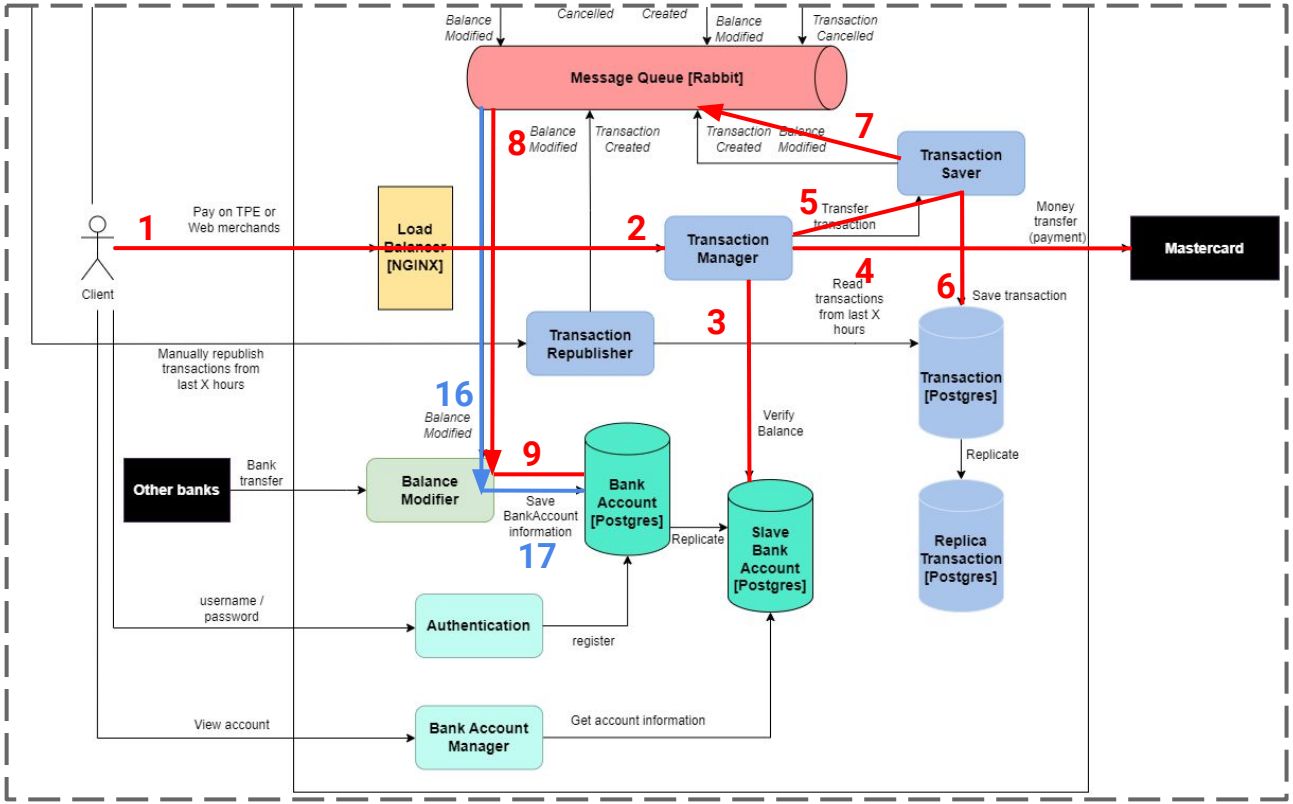


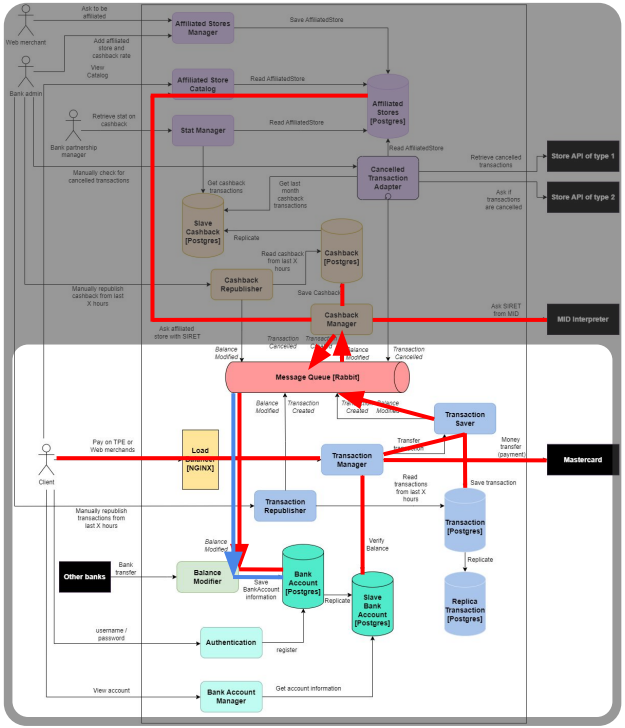
Scénario - Paiement avec Cashback





Scénario - Paiement avec Cashback





Scénario - Paiement avec Cashback

Scénario

NGINX

Failover

RabbitMQ

Failover

Paramètres du test

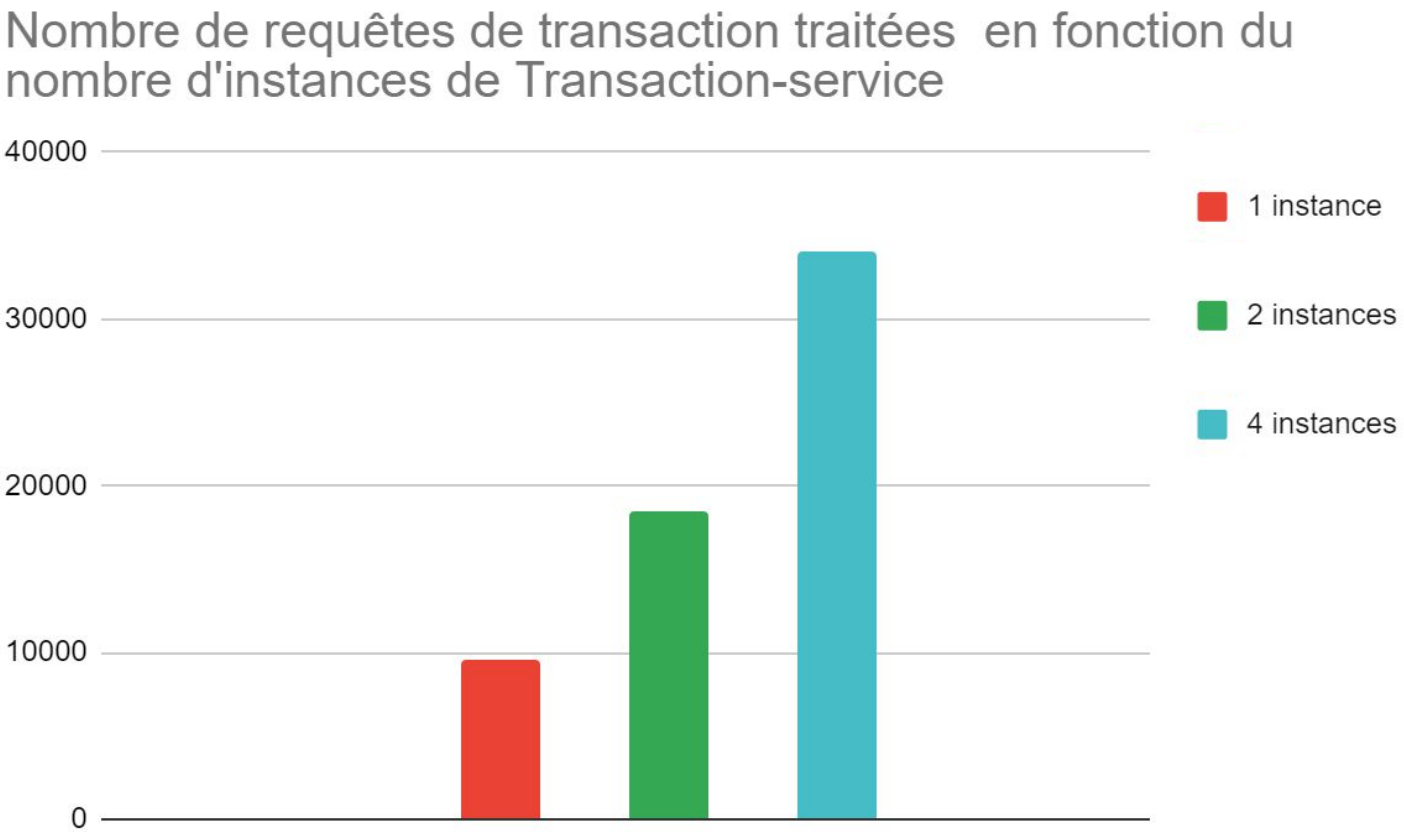
- 30% de puissance CPU par service
- 200 Mo de RAM par service
- 1000 utilisateurs en parallèles.

```
0.71%  ✓ 25          ✗ 3480
```

```
100.00% ✓ 8422       ✗ 0
```

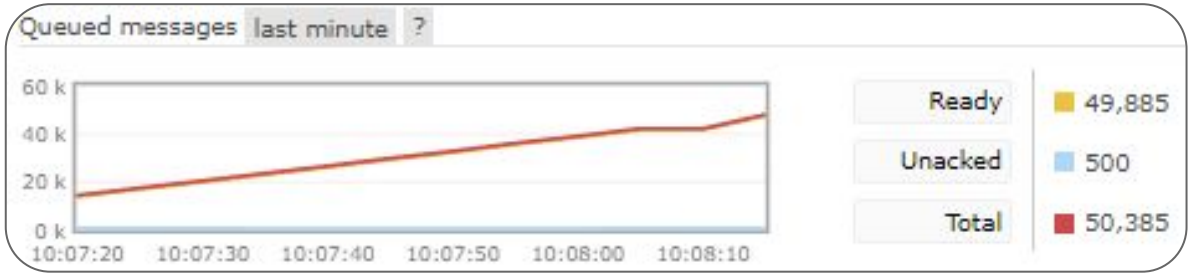
Transaction-service limité à 50% de puissance CPU et 300 Mo de RAM

Tests de charge : Paiements avec Cashback

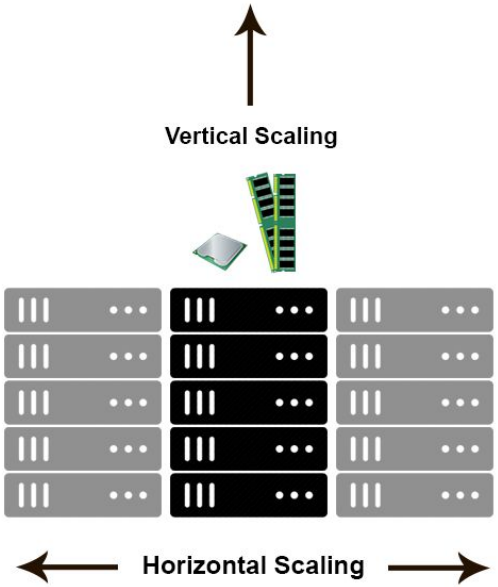


Tests de charge : Paiements avec Cashback

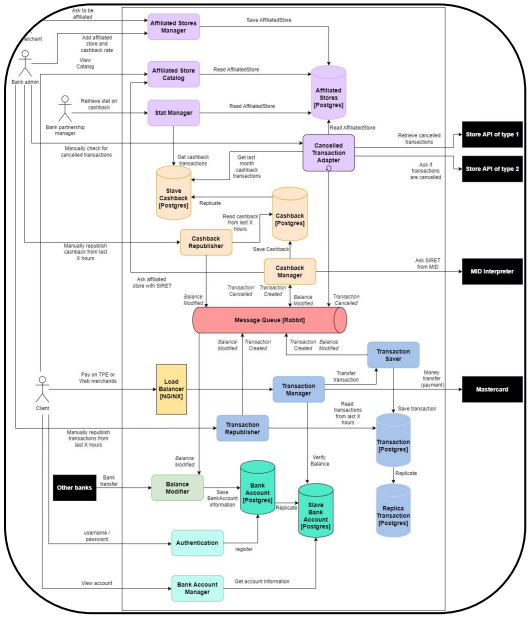
				Prêt	Débit entrant	Débit sortant
balance-queue	classic	D	running	17,891	375/s	40/s

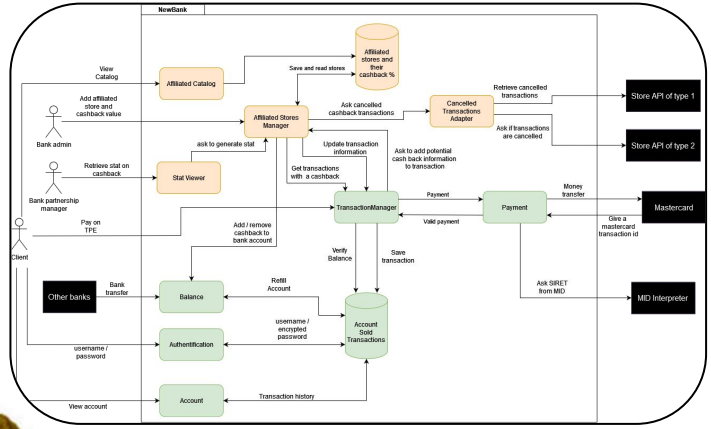


Scaling de Balance service pour suivre la production excessive de message sur la queue



Tests de charge : Impacts sur la Message Queue





- Mise à l’échelle
- Faible couplage
- Résilient

- Latence élevée
- Faible capacité
- Couplage fort

Tests de charge : Conclusion

Organisation de l’équipe

- Séparation des responsabilités et des services de la partie Cashback
- Implémentation du pattern **Master-Slave** de la partie **Cashback**

- Implémentation de **RabbitMQ**
- **Tests de charges K6** et recherche des points bloquants

- Séparation des responsabilités et des services de la partie Banque
- Implémentation du **Load Balancer**

- Séparation des responsabilités et des services de la partie Banque
- Implémentation des patterns **Master-Slave** de la partie **Banque**

Perspectives futures

Migration de la base de donnée “Transaction” PostgreSQL en NoSQL

Automatisation du système de recover après une panne de la queue

Cache dans le service des magasins partenaires