



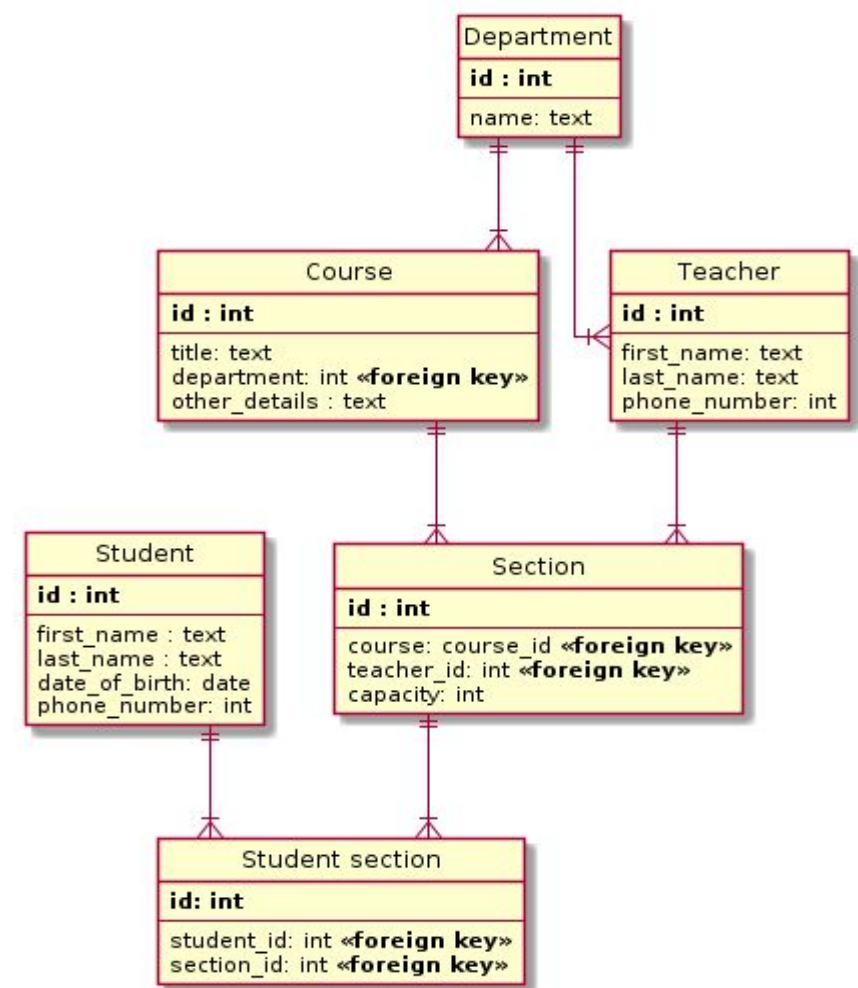
# Виды баз данных

Разновидности, HBase, Cassandra

# ВИДЫ БАЗ ДАННЫХ

# Relational

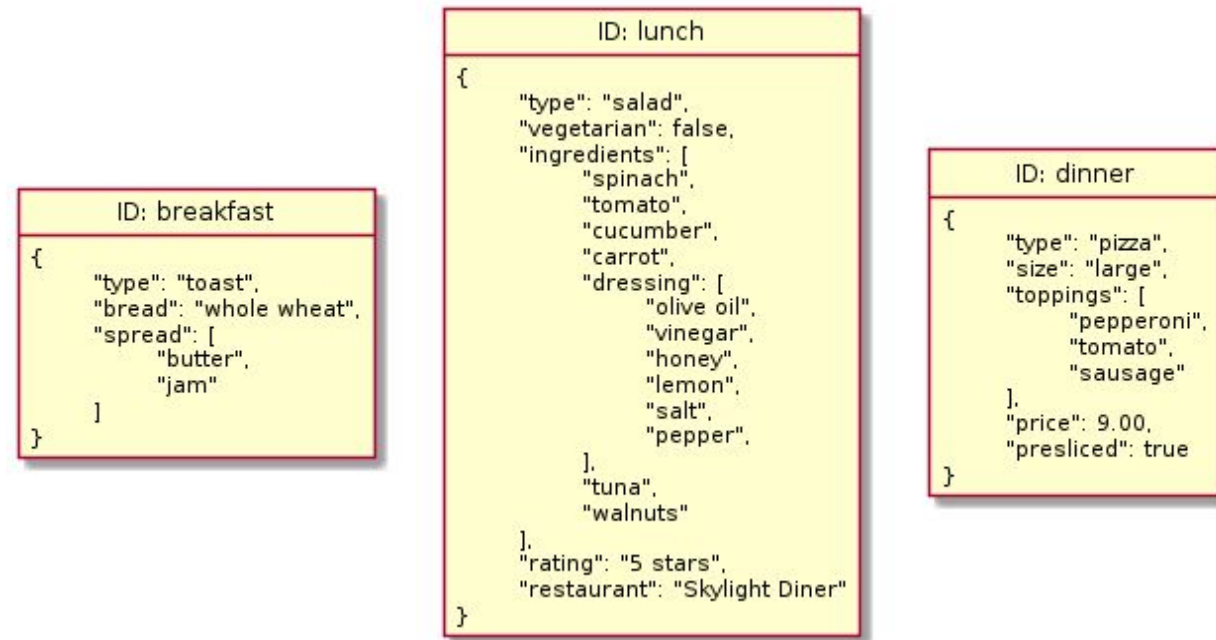
# 01



1. Oracle DB
2. MySQL
3. Postgre
4. Greenplum

# Document-oriented

You know why frontend devs have lunch alone?  
They don't know how to join tables



02



**amazon**  
DynamoDB



mongoDB®

# In memory

03



1. Redis
2. Tarantool
3. Ignite

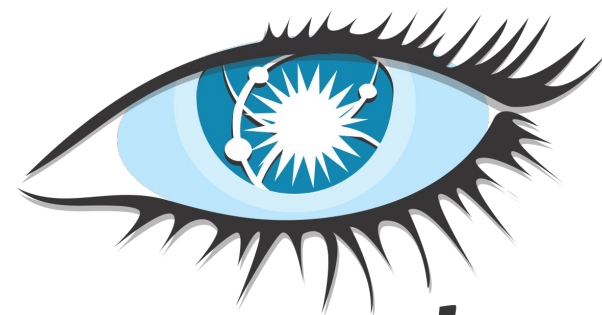
# Wide column

04

- Быстрая запись
- Быстрое чтение
- Линейная масштабируемость



A P A C H E  
**HBASE**



***cassandra***

# Column-oriented

- Быстрые запросы
- Хранить петабайты
- Близко к SQL
- Линейная масштабируемость



# Search engine

Document 1

The bright blue butterfly hangs on the breeze.

Document 2

It's best to forget the great sky and to retire from every wind.

Document 3

Under blue sky, in bright sunlight, one need not search around.

Stopword list

a  
and  
around  
every  
for  
from  
in  
is  
it  
not  
on  
one  
the  
to  
under

Inverted index

ID	Term	Document
1	best	2
2	blue	1, 3
3	bright	1, 3
4	butterfly	1
5	breeze	1
6	forget	2
7	great	2
8	hangs	1
9	need	3
10	retire	2
11	search	3
12	sky	2, 3
13	wind	2



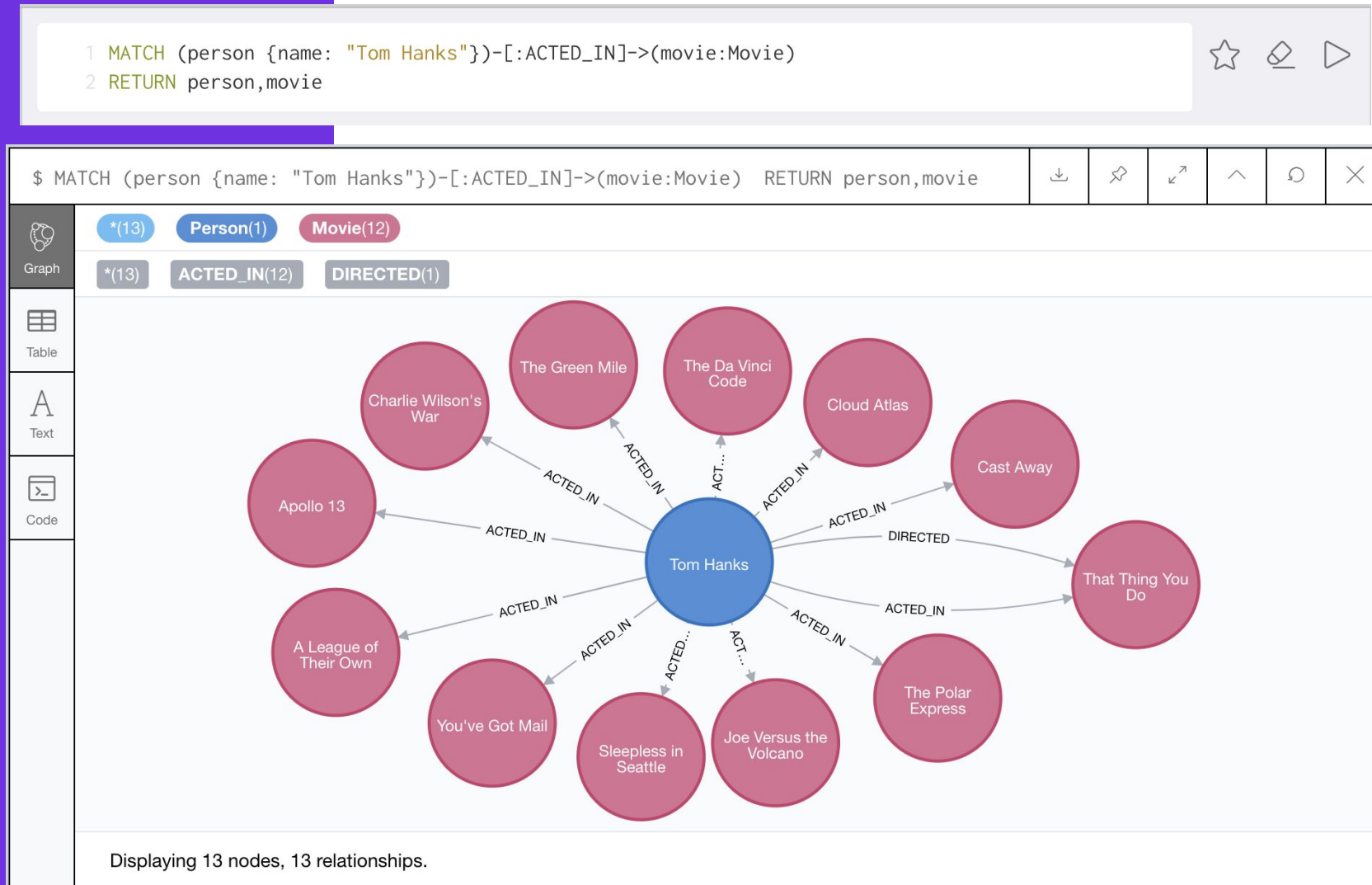
elasticsearch





# Graph

07

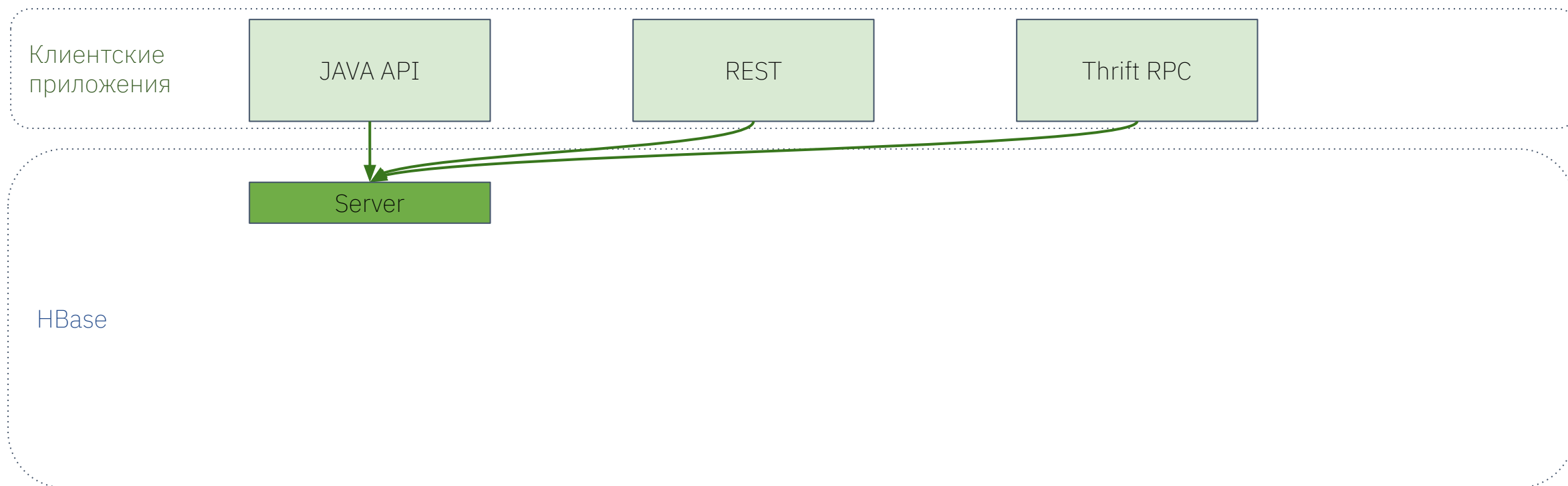


# АРХИТЕКТУРА HBASE

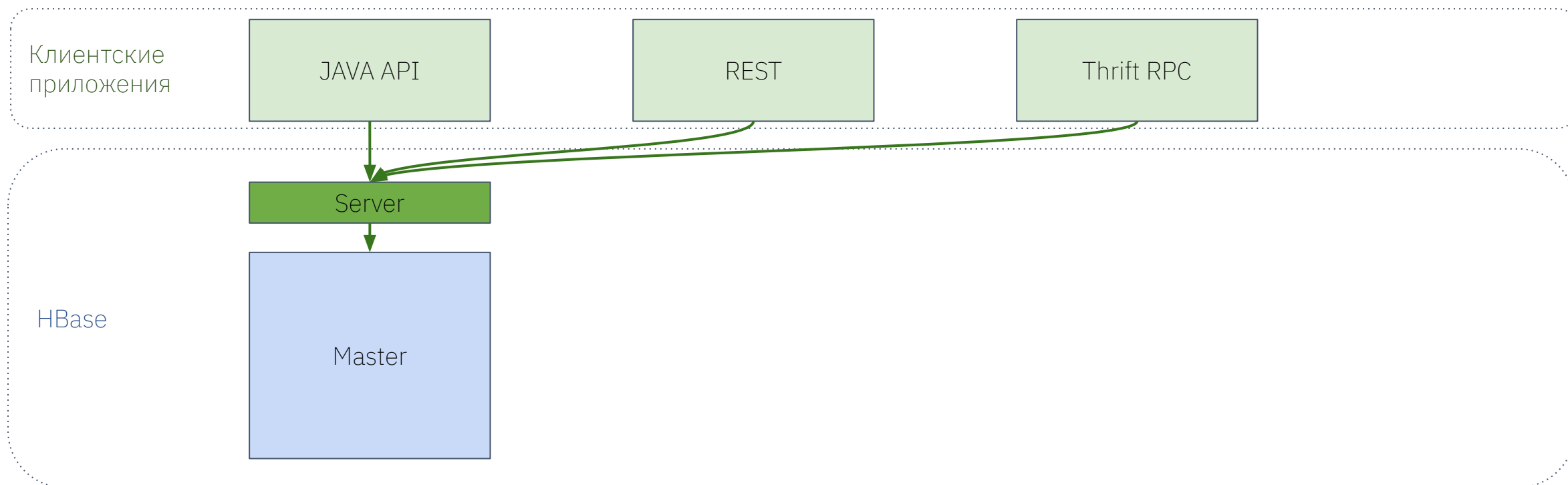
# АРХИТЕКТУРА HBASE



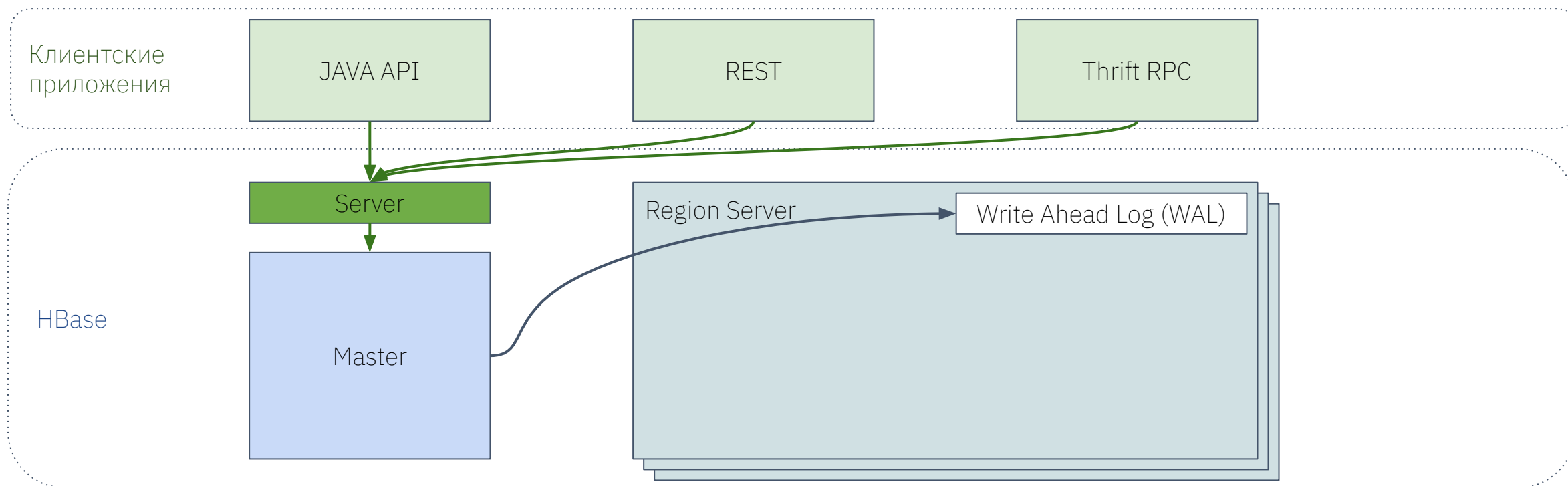
# АРХИТЕКТУРА HBASE



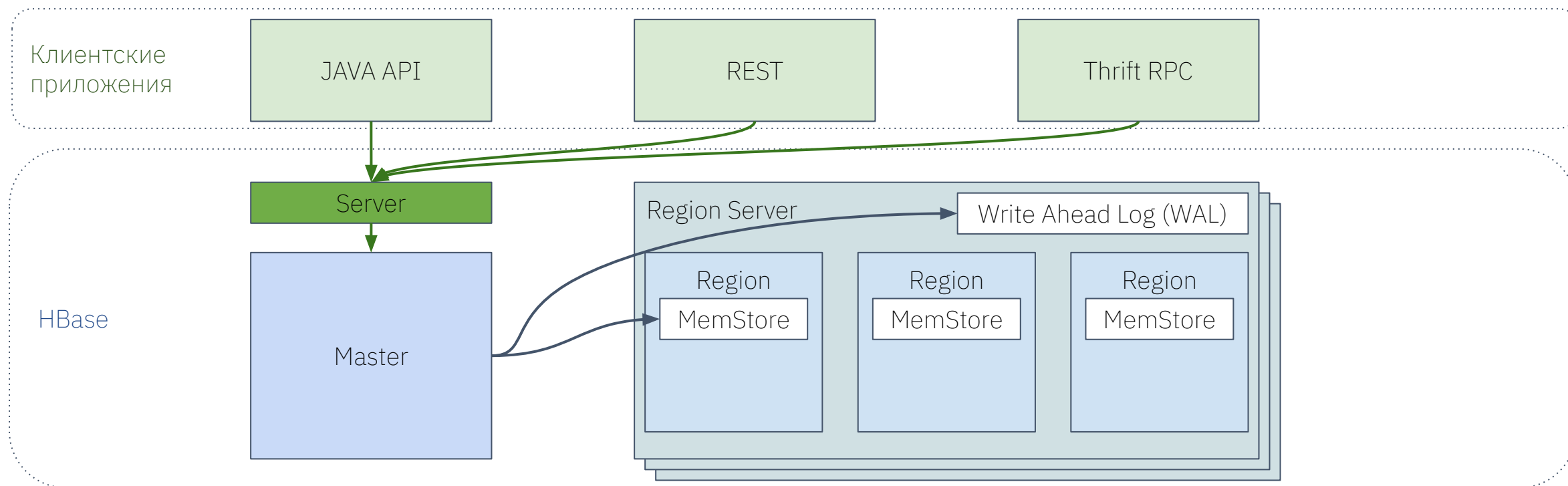
# АРХИТЕКТУРА HBASE



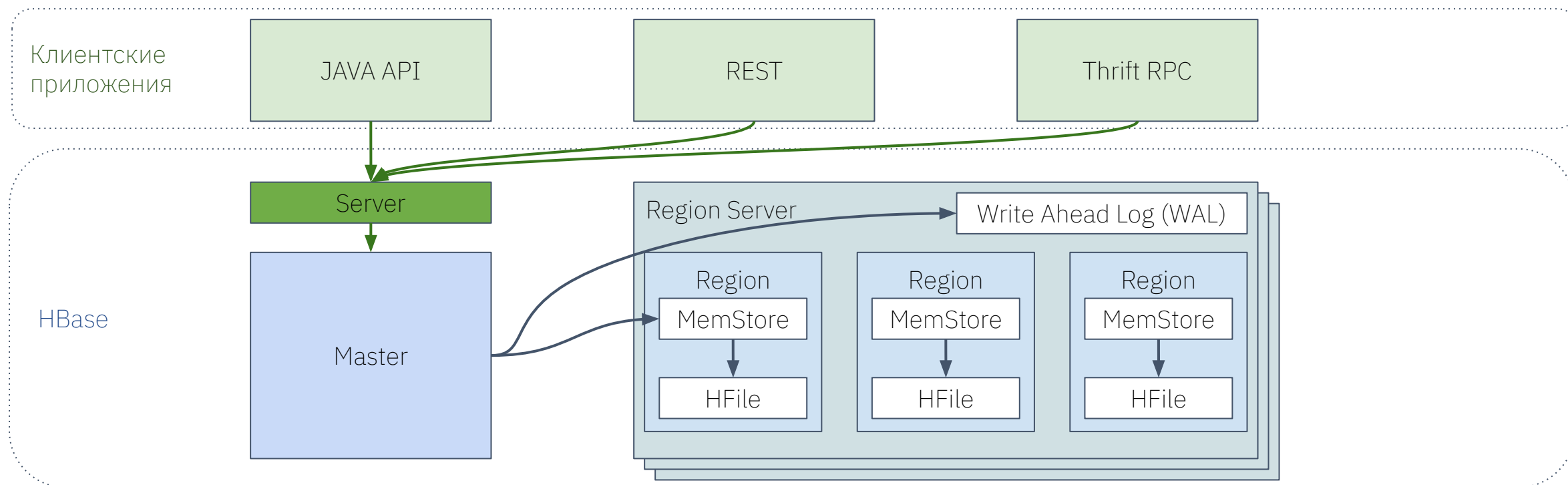
# АРХИТЕКТУРА HBASE



# АРХИТЕКТУРА HBASE

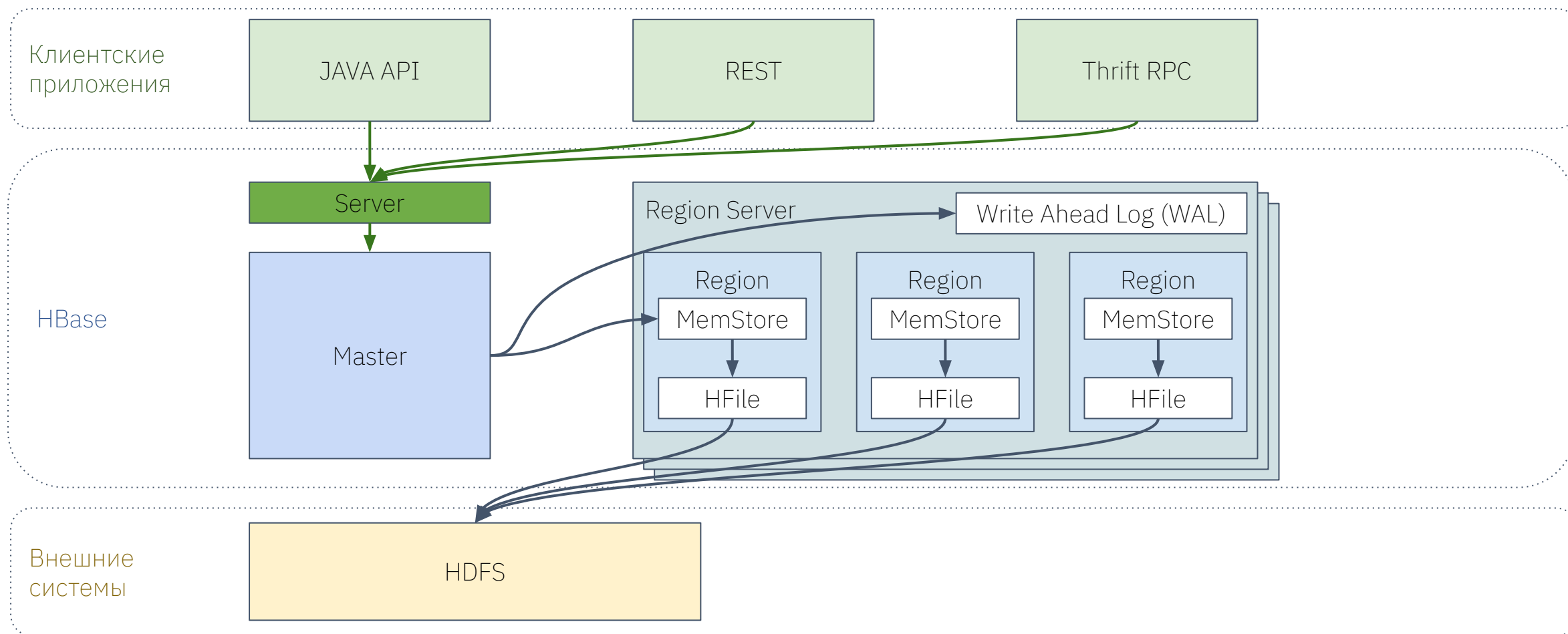


# АРХИТЕКТУРА HBASE

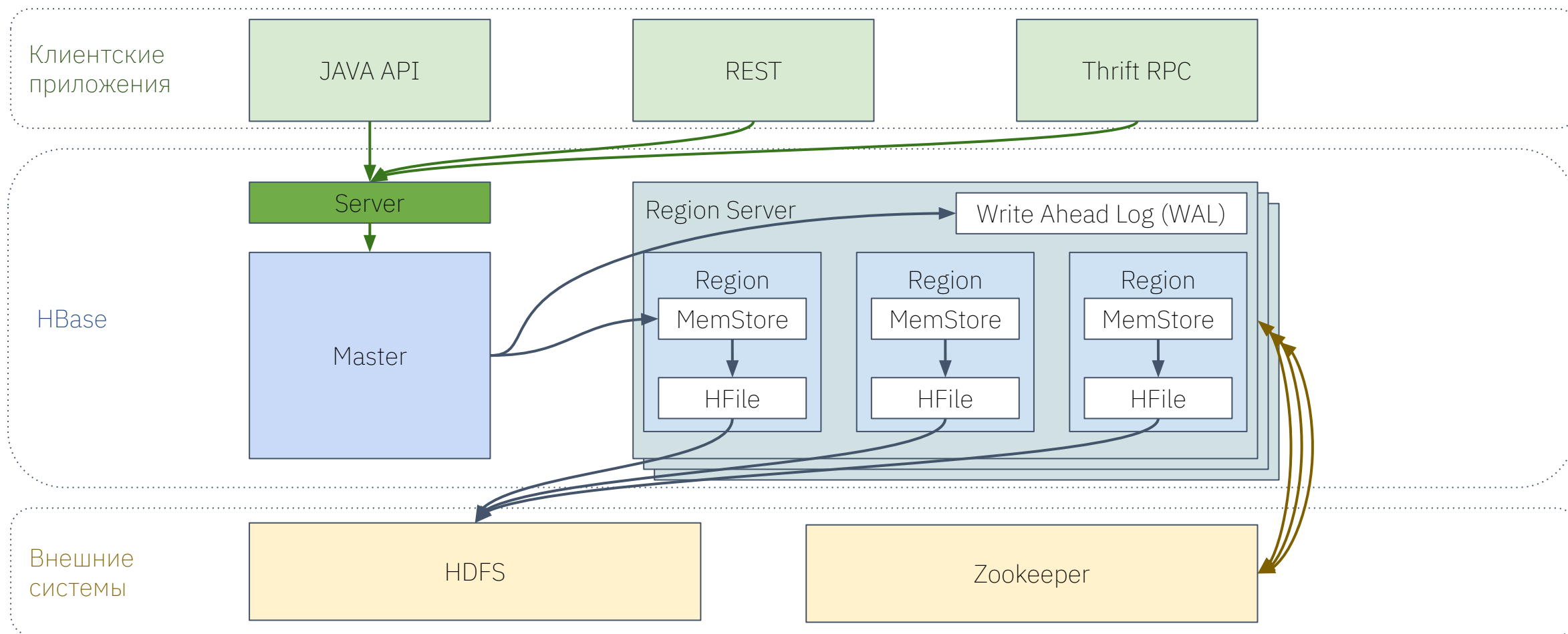




# АРХИТЕКТУРА HBASE



# АРХИТЕКТУРА HBASE



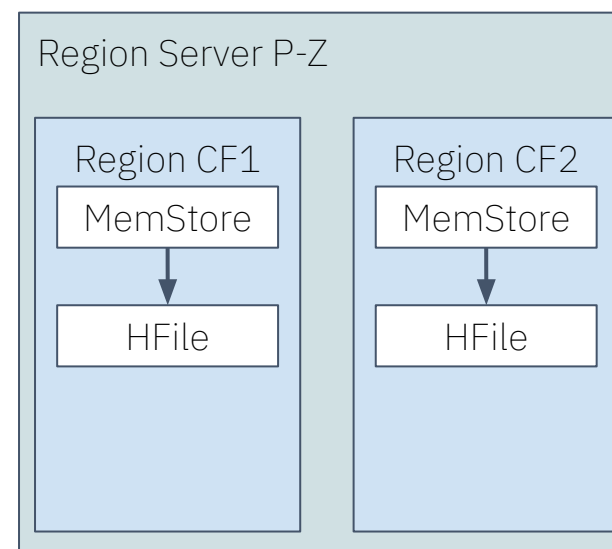
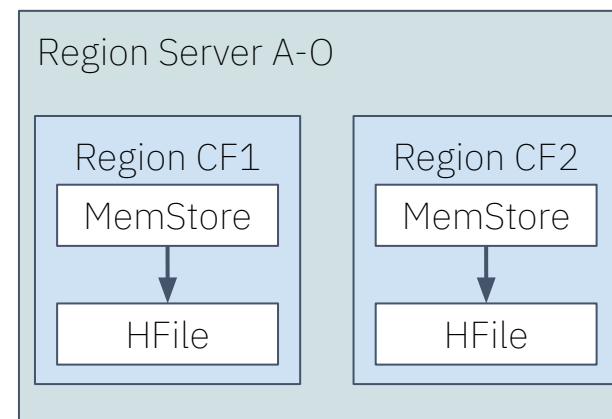
# СТРУКТУРА ТАБЛИЦЫ

# СТРУКТУРА ТАБЛИЦЫ В HBASE

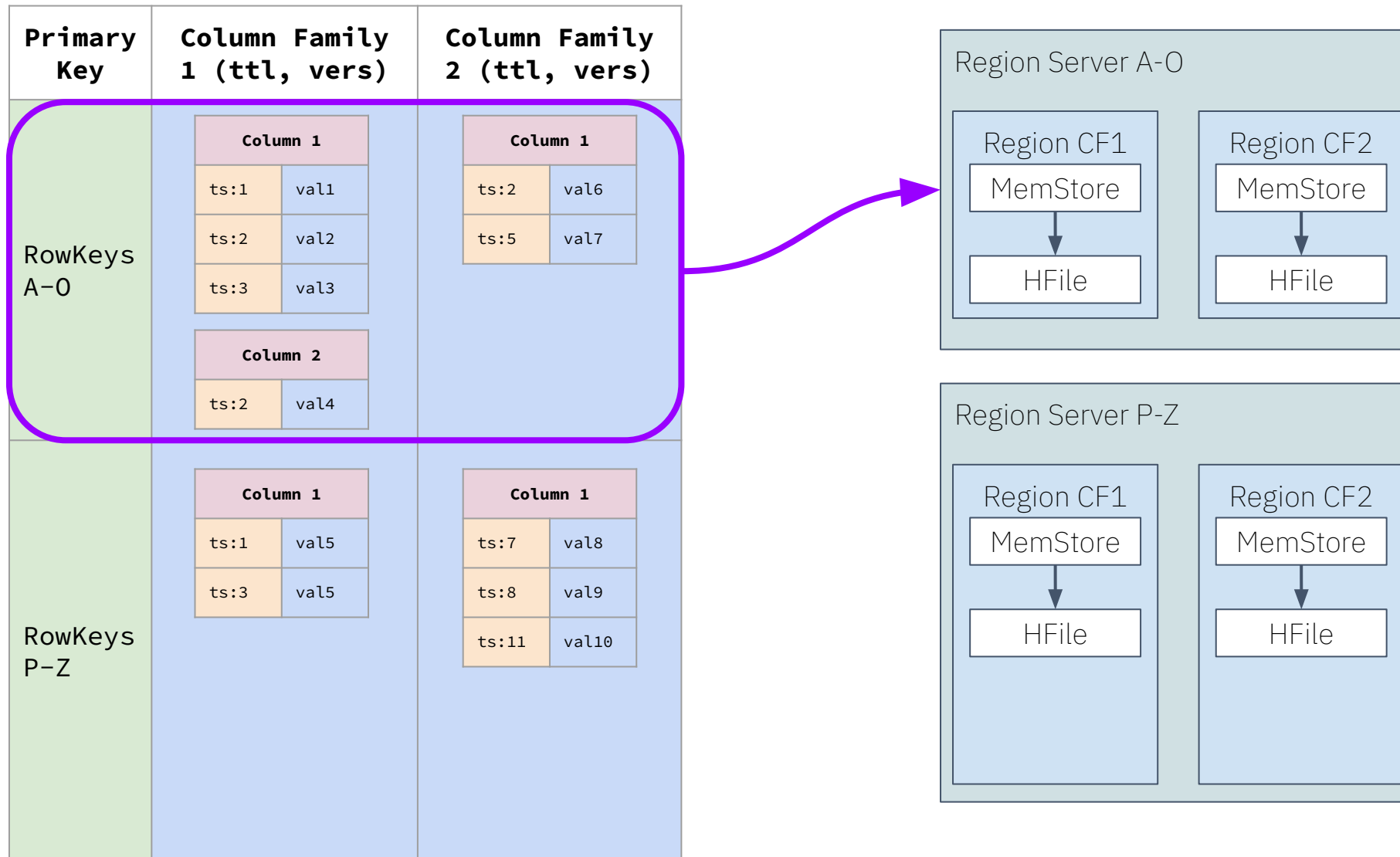
Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)
Row Key 1	<div>Column 1</div> <div>ts:1val1</div> <div>ts:2val2</div> <div>ts:3val3</div> <div>Column 2</div> <div>ts:2val4</div>	<div>Column 1</div> <div>ts:2val6</div> <div>ts:5val7</div>
Row Key 2	<div>Column 1</div> <div>ts:1val5</div> <div>ts:3val5</div>	<div>Column 1</div> <div>ts:7val8</div> <div>ts:8val9</div> <div>ts:11val10</div>

# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ

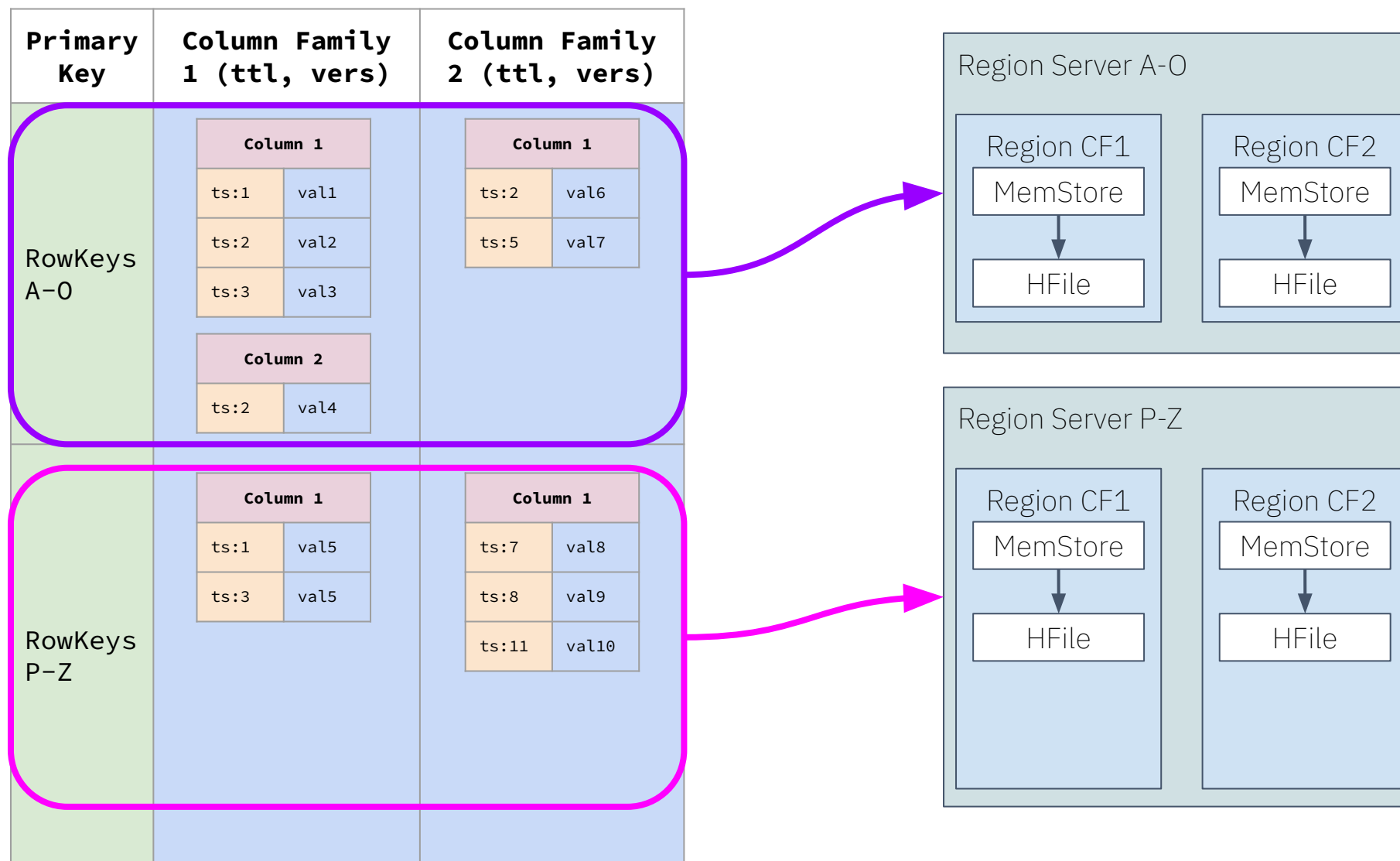
Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)
RowKeys A-O	<div>Column 1</div> <div>ts:1val1</div> <div>ts:2val2</div> <div>ts:3val3</div>	<div>Column 1</div> <div>ts:2val6</div> <div>ts:5val7</div>
	<div>Column 2</div> <div>ts:2val4</div>	
RowKeys P-Z	<div>Column 1</div> <div>ts:1val5</div> <div>ts:3val5</div>	<div>Column 1</div> <div>ts:7val8</div> <div>ts:8val9</div> <div>ts:11val10</div>



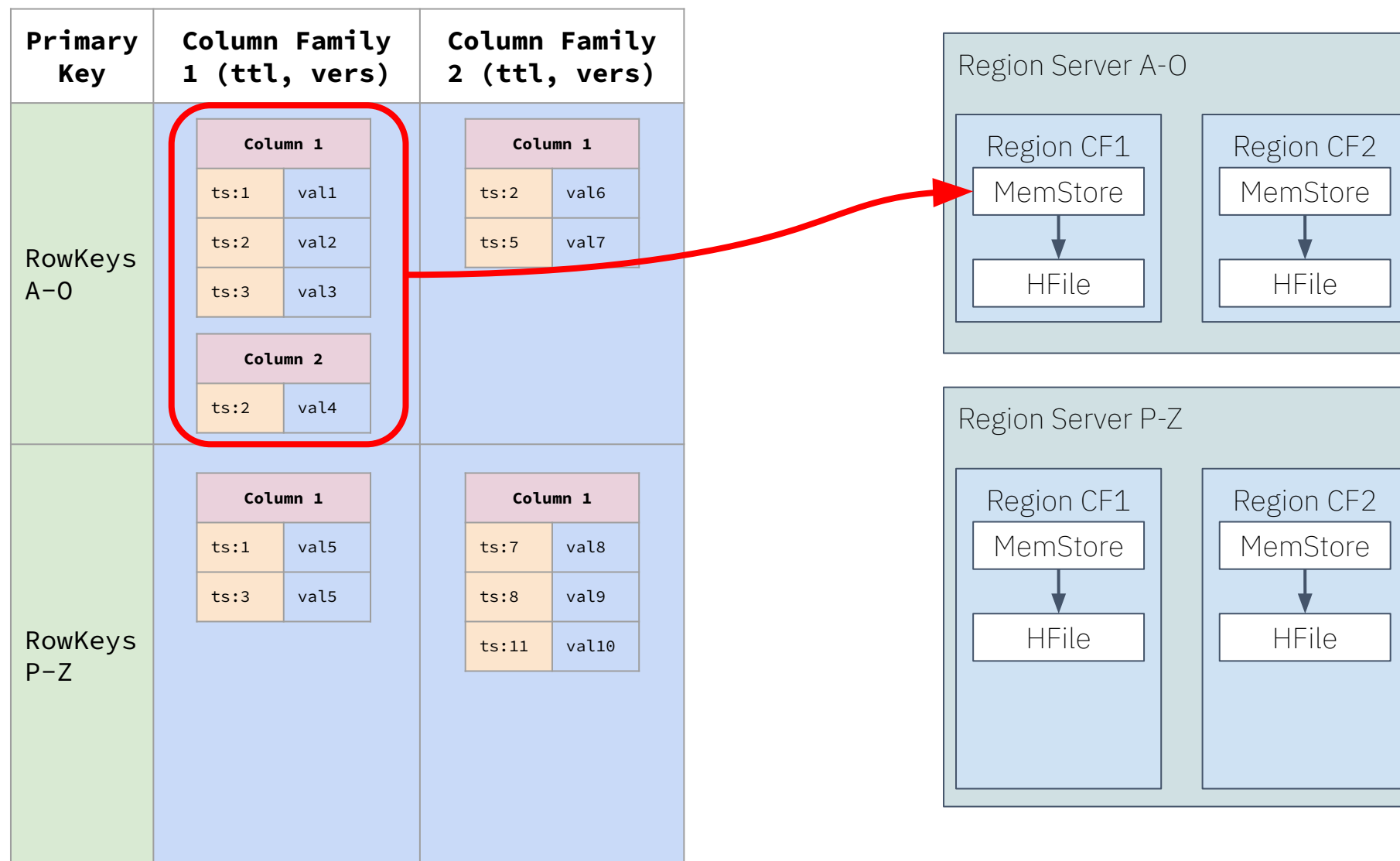
# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ



# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ

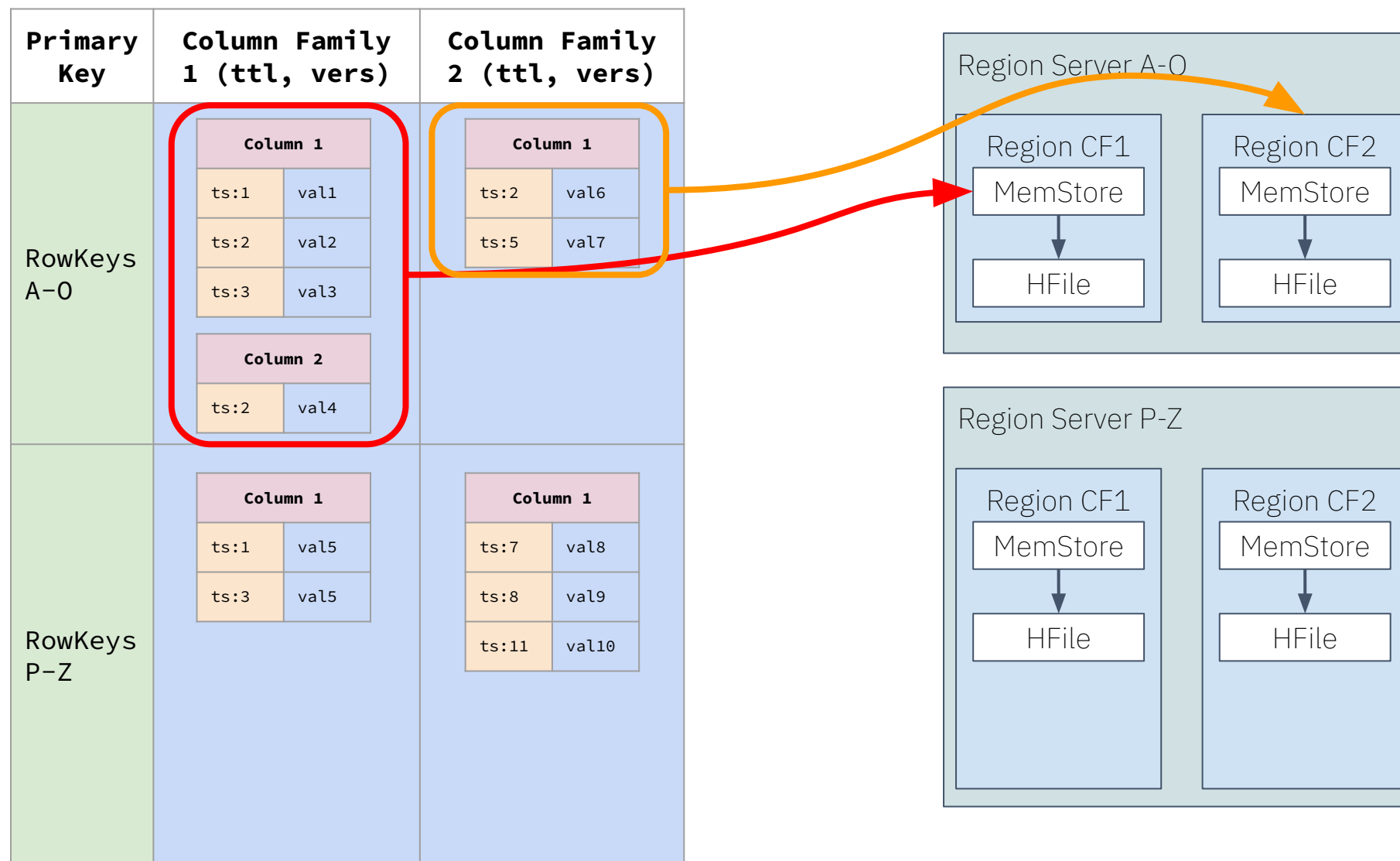


# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ

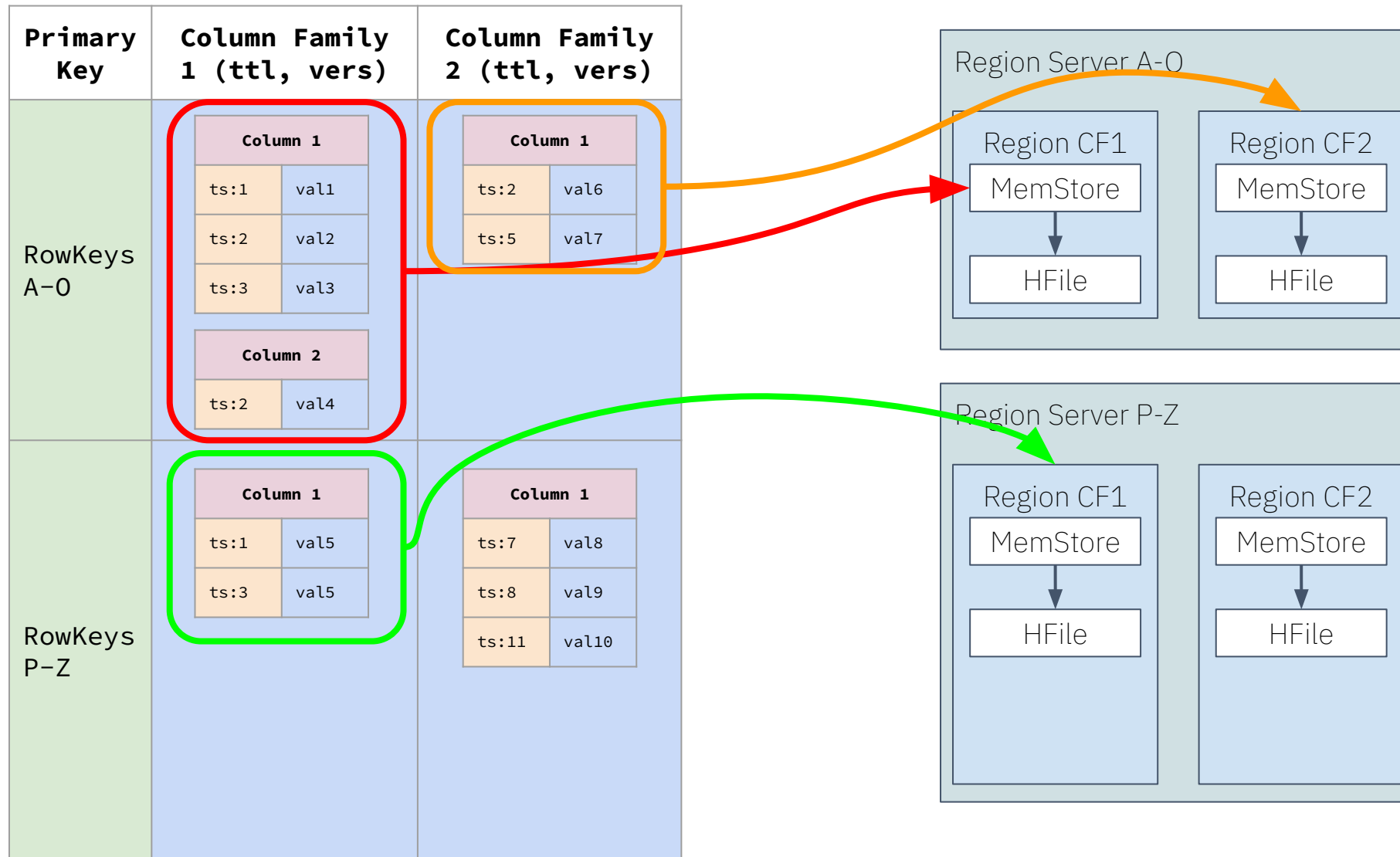




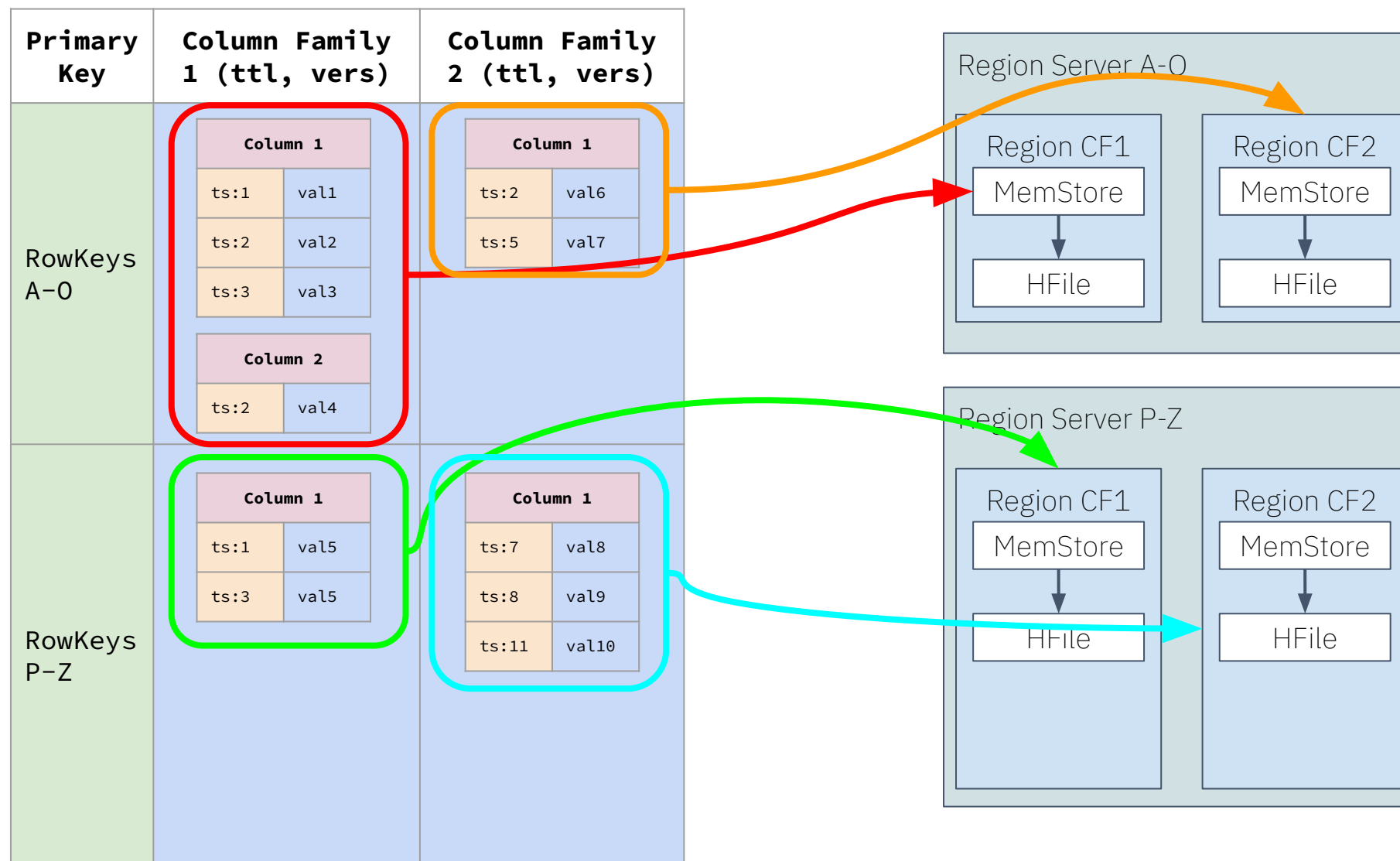
# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ



# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ



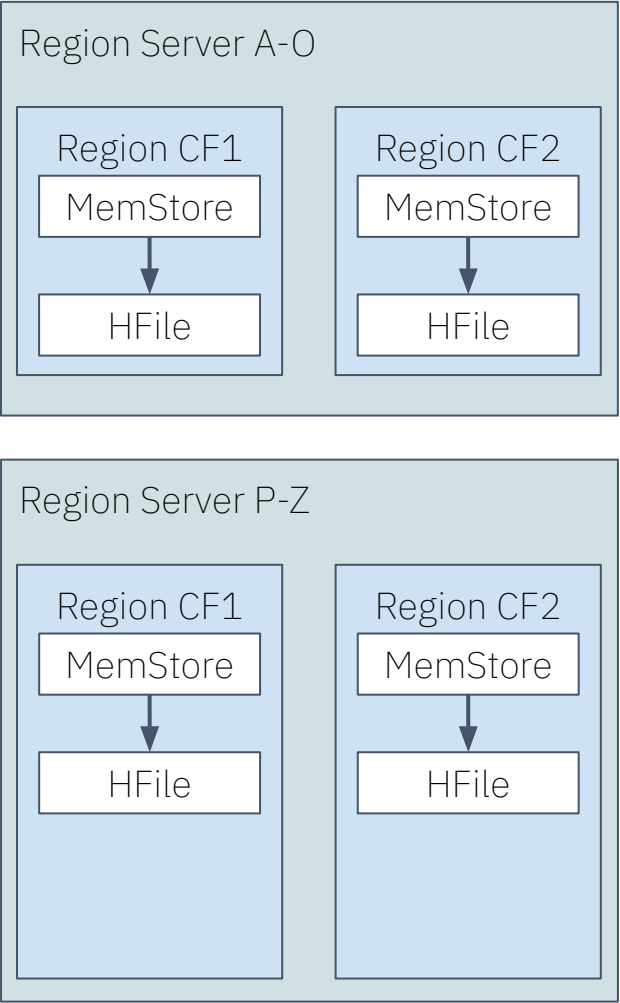
# ФИЗИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ ТАБЛИЦЫ



ВАШИ ВОПРОСЫ

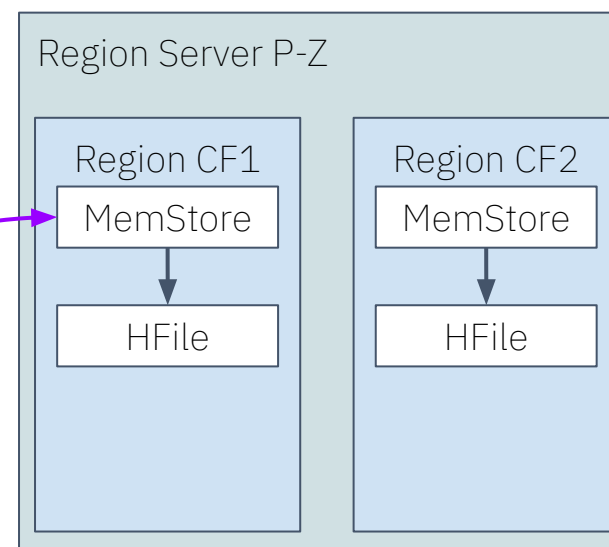
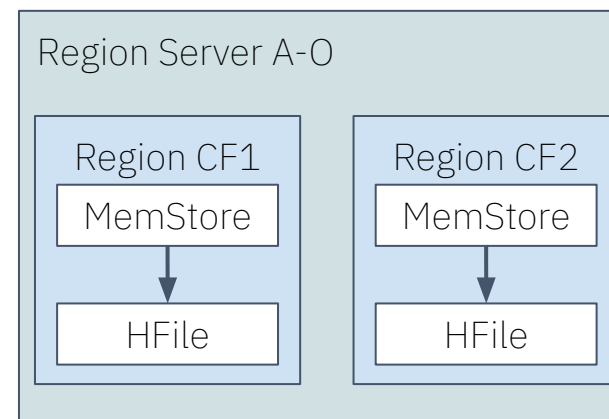
# PUT, UPDATE, DELETE

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)
RowKeys A-O	<div><div>Column 1</div><div><div>ts:1</div><div>val1</div></div><div><div>ts:2</div><div>val2</div></div><div><div>ts:3</div><div>val3</div></div></div> <div><div>Column 2</div><div><div>ts:2</div><div>val4</div></div></div>	<div><div>Column 1</div><div><div>ts:2</div><div>val6</div></div><div><div>ts:5</div><div>val7</div></div></div>
RowKeys P-Z	<div><div>Column 1</div><div><div>ts:1</div><div>val5</div></div><div><div>ts:3</div><div>val5</div></div></div> <div><div>ts:0</div><div>NEW</div></div> <div>put update delete</div>	<div><div>Column 1</div><div><div>ts:7</div><div>val8</div></div><div><div>ts:8</div><div>val9</div></div><div><div>ts:11</div><div>val10</div></div></div>



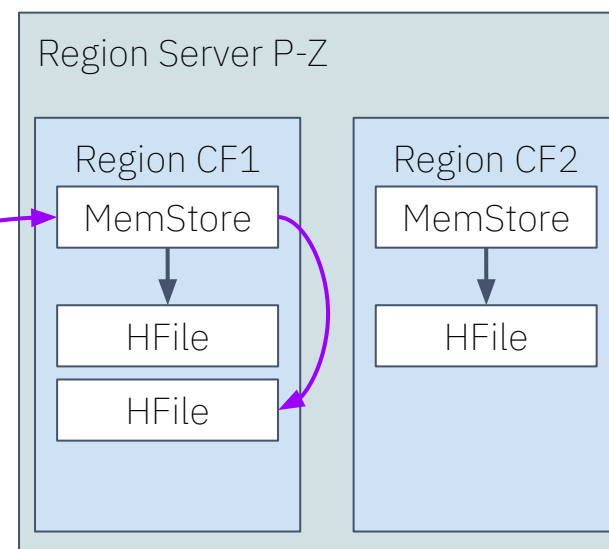
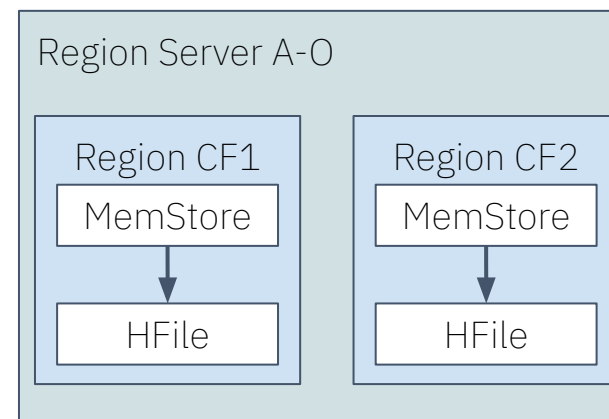
# PUT, UPDATE, DELETE

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)																		
RowKeys A-0	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val1</td></tr><tr><td>ts:2</td><td>val2</td></tr><tr><td>ts:3</td><td>val3</td></tr></table> <table><tr><th colspan="2">Column 2</th></tr><tr><td>ts:2</td><td>val4</td></tr></table>	Column 1		ts:1	val1	ts:2	val2	ts:3	val3	Column 2		ts:2	val4	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:2</td><td>val6</td></tr><tr><td>ts:5</td><td>val7</td></tr></table>	Column 1		ts:2	val6	ts:5	val7
Column 1																				
ts:1	val1																			
ts:2	val2																			
ts:3	val3																			
Column 2																				
ts:2	val4																			
Column 1																				
ts:2	val6																			
ts:5	val7																			
RowKeys P-Z	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val5</td></tr><tr><td>ts:3</td><td>val5</td></tr></table> <table><tr><td>ts:0</td><td>NEW</td></tr></table> <div>put update delete</div>	Column 1		ts:1	val5	ts:3	val5	ts:0	NEW	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:7</td><td>val8</td></tr><tr><td>ts:8</td><td>val9</td></tr><tr><td>ts:11</td><td>val10</td></tr></table>	Column 1		ts:7	val8	ts:8	val9	ts:11	val10		
Column 1																				
ts:1	val5																			
ts:3	val5																			
ts:0	NEW																			
Column 1																				
ts:7	val8																			
ts:8	val9																			
ts:11	val10																			



# PUT, UPDATE, DELETE

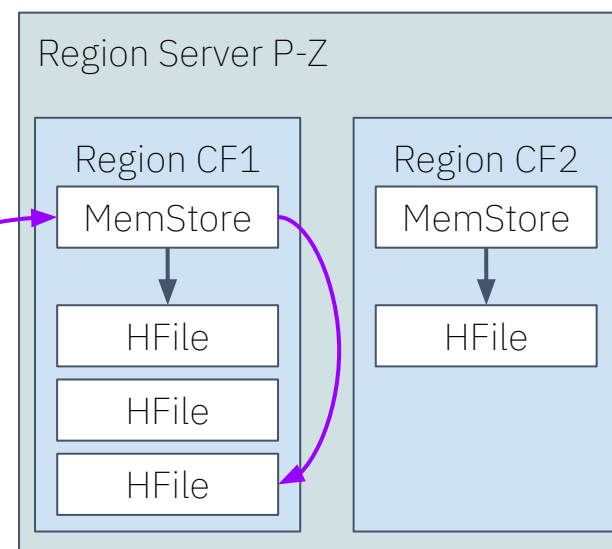
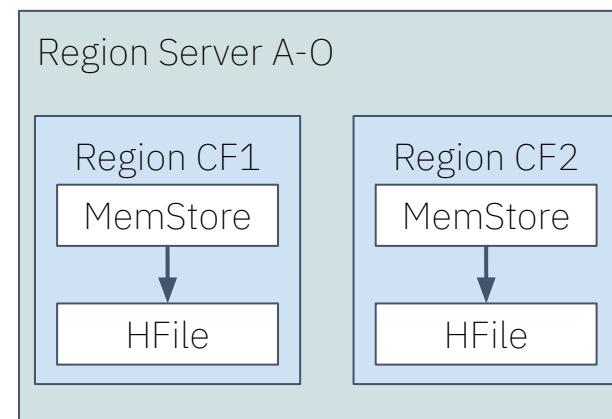
Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)																		
RowKeys A-0	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val1</td></tr><tr><td>ts:2</td><td>val2</td></tr><tr><td>ts:3</td><td>val3</td></tr></table> <table><tr><th colspan="2">Column 2</th></tr><tr><td>ts:2</td><td>val4</td></tr></table>	Column 1		ts:1	val1	ts:2	val2	ts:3	val3	Column 2		ts:2	val4	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:2</td><td>val6</td></tr><tr><td>ts:5</td><td>val7</td></tr></table>	Column 1		ts:2	val6	ts:5	val7
Column 1																				
ts:1	val1																			
ts:2	val2																			
ts:3	val3																			
Column 2																				
ts:2	val4																			
Column 1																				
ts:2	val6																			
ts:5	val7																			
RowKeys P-Z	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val5</td></tr><tr><td>ts:3</td><td>val5</td></tr></table> <table><tr><td>ts:0</td><td>NEW</td></tr></table> <div>put update delete</div>	Column 1		ts:1	val5	ts:3	val5	ts:0	NEW	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:7</td><td>val8</td></tr><tr><td>ts:8</td><td>val9</td></tr><tr><td>ts:11</td><td>val10</td></tr></table>	Column 1		ts:7	val8	ts:8	val9	ts:11	val10		
Column 1																				
ts:1	val5																			
ts:3	val5																			
ts:0	NEW																			
Column 1																				
ts:7	val8																			
ts:8	val9																			
ts:11	val10																			



# PUT, UPDATE, DELETE

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)																		
RowKeys A-0	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val1</td></tr><tr><td>ts:2</td><td>val2</td></tr><tr><td>ts:3</td><td>val3</td></tr></table> <table><tr><th colspan="2">Column 2</th></tr><tr><td>ts:2</td><td>val4</td></tr></table>	Column 1		ts:1	val1	ts:2	val2	ts:3	val3	Column 2		ts:2	val4	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:2</td><td>val6</td></tr><tr><td>ts:5</td><td>val7</td></tr></table>	Column 1		ts:2	val6	ts:5	val7
Column 1																				
ts:1	val1																			
ts:2	val2																			
ts:3	val3																			
Column 2																				
ts:2	val4																			
Column 1																				
ts:2	val6																			
ts:5	val7																			
RowKeys P-Z	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:0</td><td>NEW</td></tr><tr><td>ts:1</td><td>val5</td></tr><tr><td>ts:3</td><td>val5</td></tr></table> <table><tr><td>ts:0</td><td>NEW</td></tr></table> <div>put update delete</div>	Column 1		ts:0	NEW	ts:1	val5	ts:3	val5	ts:0	NEW	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:7</td><td>val8</td></tr><tr><td>ts:8</td><td>val9</td></tr><tr><td>ts:11</td><td>val10</td></tr></table>	Column 1		ts:7	val8	ts:8	val9	ts:11	val10
Column 1																				
ts:0	NEW																			
ts:1	val5																			
ts:3	val5																			
ts:0	NEW																			
Column 1																				
ts:7	val8																			
ts:8	val9																			
ts:11	val10																			

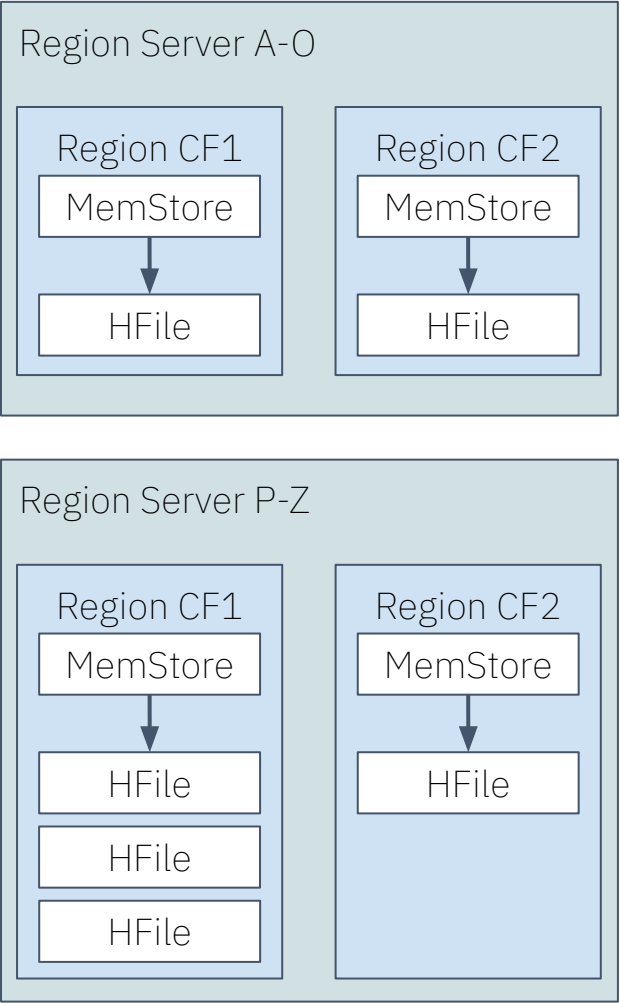
put  
update  
delete





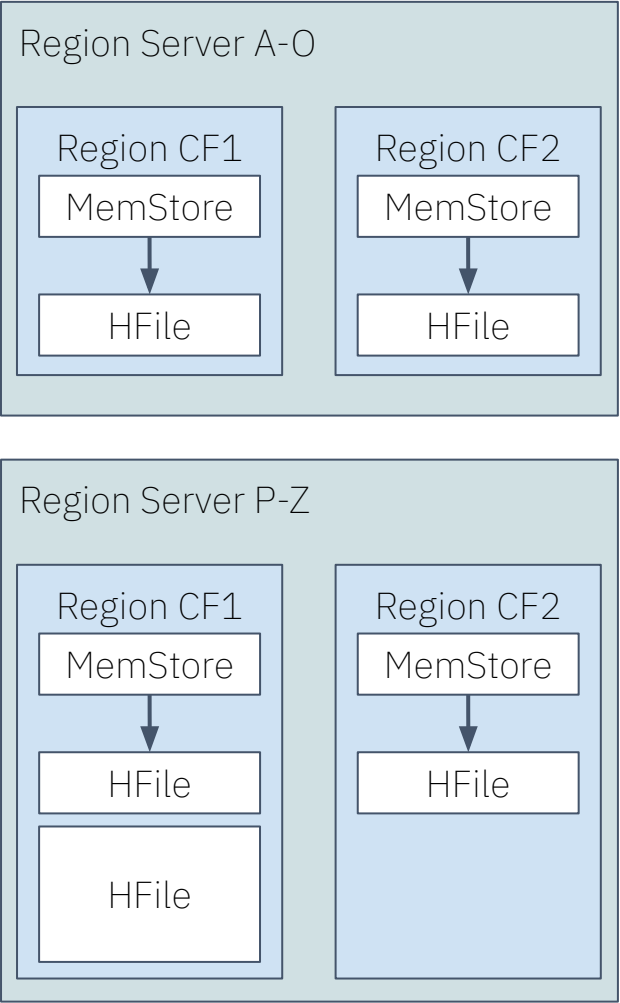
# MINOR COMPACTION

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)																		
RowKeys A-0	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:1</td><td>val1</td></tr><tr><td>ts:2</td><td>val2</td></tr><tr><td>ts:3</td><td>val3</td></tr><tr><th colspan="2">Column 2</th></tr><tr><td>ts:2</td><td>val4</td></tr></table>	Column 1		ts:1	val1	ts:2	val2	ts:3	val3	Column 2		ts:2	val4	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:2</td><td>val6</td></tr><tr><td>ts:5</td><td>val7</td></tr></table>	Column 1		ts:2	val6	ts:5	val7
	Column 1																			
	ts:1	val1																		
	ts:2	val2																		
	ts:3	val3																		
	Column 2																			
ts:2	val4																			
Column 1																				
ts:2	val6																			
ts:5	val7																			
RowKeys P-Z	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:0</td><td>NEW</td></tr><tr><td>ts:0</td><td>NEW</td></tr><tr><td>ts:1</td><td>val5</td></tr><tr><td>ts:3</td><td>val5</td></tr></table>	Column 1		ts:0	NEW	ts:0	NEW	ts:1	val5	ts:3	val5	<table><tr><th colspan="2">Column 1</th></tr><tr><td>ts:7</td><td>val8</td></tr><tr><td>ts:8</td><td>val9</td></tr><tr><td>ts:11</td><td>val10</td></tr></table>	Column 1		ts:7	val8	ts:8	val9	ts:11	val10
	Column 1																			
	ts:0	NEW																		
	ts:0	NEW																		
	ts:1	val5																		
	ts:3	val5																		
Column 1																				
ts:7	val8																			
ts:8	val9																			
ts:11	val10																			



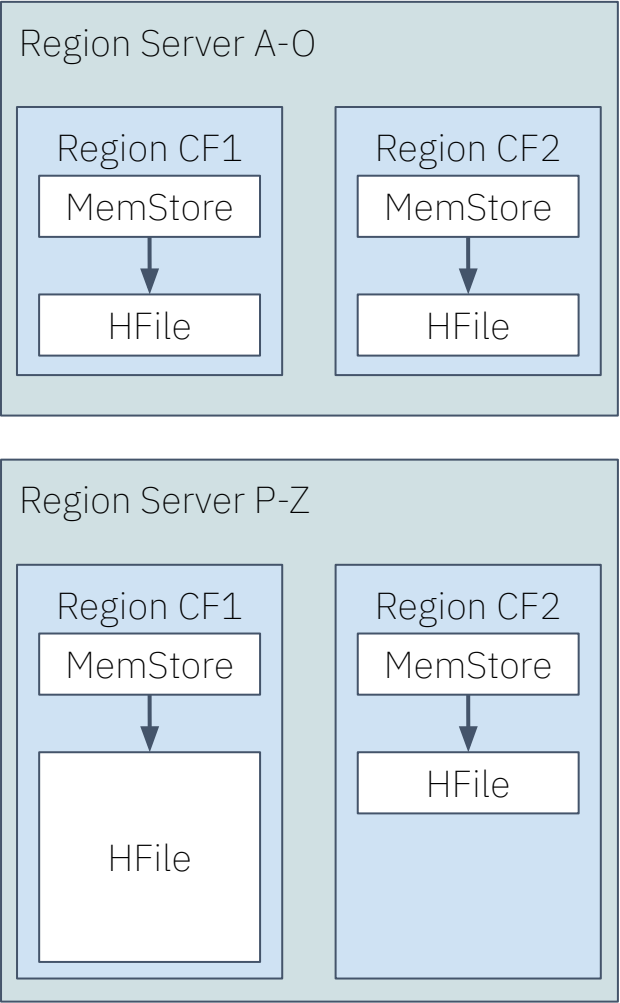
# MINOR COMPACTION

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)
RowKeys A-0	<div>Column 1</div> <div>ts:1val1</div> <div>ts:2val2</div> <div>ts:3val3</div>	<div>Column 1</div> <div>ts:2val6</div> <div>ts:5val7</div>
	<div>Column 2</div> <div>ts:2val4</div>	
RowKeys P-Z	<div>Column 1</div> <div>ts:0NEW</div> <div>ts:0NEW</div> <div>ts:1val5</div> <div>ts:3val5</div>	<div>Column 1</div> <div>ts:7val8</div> <div>ts:8val9</div> <div>ts:11val10</div>



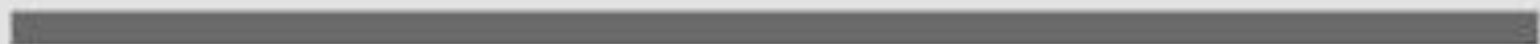
# MAJOR COMPACTION

Primary Key	Column Family 1 (ttl, vers)	Column Family 2 (ttl, vers)
RowKeys A-0	<div>Column 1</div> <div>ts:1val1</div> <div>ts:2val2</div> <div>ts:3val3</div>	<div>Column 1</div> <div>ts:2val6</div> <div>ts:5val7</div>
	<div>Column 2</div> <div>ts:2val4</div>	
RowKeys P-Z	<div>Column 1</div> <div>ts:0NEW</div> <div>ts:0NEW</div> <div>ts:1val5</div> <div>ts:3val5</div>	<div>Column 1</div> <div>ts:7val8</div> <div>ts:8val9</div> <div>ts:11val10</div>



ПЕРЕРЫВ

**10:00**



# РАБОТА С HBASE

# ПОДНИМАЕМ ЛОКАЛЬНЫЙ КЛАСТЕР

```
docker start -i gbhdp
```

# УСТАНОВКА HBASE

Скачиваем и распаковываем дистрибутив:

```
$ wget https://apache-mirror.rbc.ru/pub/apache/hbase/2.3.7/hbase-2.3.7-bin.tar.gz  
$ tar xzf hbase-2.3.7-bin.tar.gz  
$ rm hbase-2.3.7-bin.tar.gz  
$ mv hbase-2.3.7-bin hbase
```

Задаем необходимые переменные окружения:

```
$ cd hbase  
$ export HBASE_HOME=`pwd`  
$ export PATH=$PATH:$HBASE_HOME/bin  
$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

Создаем служебные директории в HDFS:

```
$ hdfs dfs -mkdir -p /user/hbase  
$ hdfs dfs -chmod a+w /user/hbase
```



# НАСТРОЙКА HBASE

Убедимся, что в файле `~/hbase/conf/hbase-site.xml` установлены свойства:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>hdfs://localhost:9000/user/hbase</value>
  </property>
  <property>
    <name>hbase.cluster.distributed</name>
    <value>false</value>
  </property>
</configuration>
```

# ЗАПУСК HBASE

Запускаем сервер и входим в интерактивную оболочку:

```
$ start-hbase.sh  
$ hbase shell
```

Проверяем версию:

```
> version  
2.3.6, r7414579f2620fca6b75146c29ab2726fc4643ac9, Wed Jul 28 22:24:42 UTC 2021
```

Статус базы:

```
> status  
1 active master, 0 backup masters, 1 servers, 0 dead, 2.0000 average load
```

# СОЗДАНИЕ ТАБЛИЦЫ

Как добраться до документации:

```
> help  
> help "create"  
> table_help
```

Создадим таблицу:

```
> create 'tbl', 'cf1', 'cf2'  
Created table tbl
```

Вывести список таблиц:

```
> list  
TABLE  
tbl  
1 row(s)
```

# ОТКЛЮЧЕНИЕ ТАБЛИЦЫ

Отключим таблицу:

```
> disable 'tbl'
```

Проверим, запустив сканирование:

```
> scan 'tbl'
```

```
ERROR: Table tbl is disabled!
```

Проверим через команду:

```
> is_disabled 'tbl'
```

```
true
```

Включим таблицу:

```
> enable 'tbl'
```

# ОПИСАНИЕ ТАБЛИЦЫ

Получим описание таблицы:

```
> describe 'tbl'  
...  
2 row(s)
```

Удалим группу колонок:

```
> alter 'tbl', 'delete' => 'cf2'  
Updating all regions with the new schema...  
1/1 regions updated.  
Done.
```

# ДОБАВЛЕНИЕ ДАННЫХ

Добавим 3 строки:

```
> put 'tbl', 'gb', 'cf1:org', 'geekbrains'  
> put 'tbl', 'mr', 'cf1:org', 'mail.ru'  
> put 'tbl', 'ok', 'cf1:org', 'odnoklassniki'
```

Запустим вывод данных всей таблицы:

```
> scan 'tbl'  
3 row(s)
```

Обновим значение ключа mr:

```
> put 'tbl', 'mr', 'cf1:org', 'mail.ru group'
```

# ЧТЕНИЕ ПО КЛЮЧУ

Получим значение ключа mr:

```
> get 'tbl', 'mr'  
1 row(s)
```

Получим значение ключа mr для определенной колонки:

```
> get 'tbl', 'mr', {COLUMN => 'cf1:org'}  
1 row(s)
```

# УДАЛЕНИЕ ДАННЫХ

Удалим все данные ключа mr:

```
> deleteall 'tbl', 'mr'
```

Удалим значение для определенной колонки:

```
> delete 'tbl', 'ok', 'cf1:org'
```



# ОПЕРАЦИИ С ТАБЛИЦЕЙ

Посчитать количество строк в таблице:

```
> count 'tbl'  
1 row(s)
```

Удалить все данные и пересоздать таблицу:

```
> truncate 'tbl'  
Truncating 'tbl' table (it may take a while):  
Disabling table...  
Truncating table...
```

# УДАЛЕНИЕ ТАБЛИЦЫ

Проверим, что таблица существует:

```
> exists 'tbl'  
Table tbl does exist
```

Отключим её, а затем удалим:

```
> disable 'tbl'  
> drop 'tbl'
```

Проверим, что таблицы больше нет:

```
> exists 'tbl'  
Table tbl does not exist  
> list 'tbl'  
0 row(s)
```

# ОСТАНОВКА HBASE

Выходим из интерактивной оболочки:

```
> exit
```

Останавливаем сервер:

```
$ stop-hbase.sh
```

# ОСТАНОВКА ЛОКАЛЬНОГО КЛАСТЕРА

```
exit
```

# HBASE VS CASSANDRA

# APACHE HBASE VS APACHE CASSANDRA

**Apache HBase**

**Apache Cassandra**

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007

## Apache Cassandra

- Facebook, Java, 2008

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись

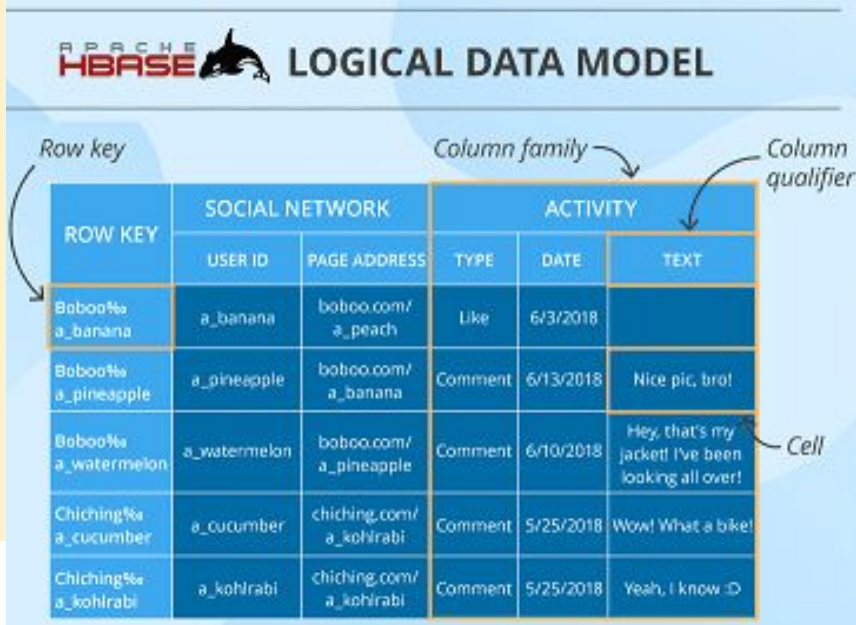


# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение
- Table -> Row Key -> Column Family -> Column Qualifier -> Timestamp -> Value

**APACHE HBASE LOGICAL DATA MODEL**



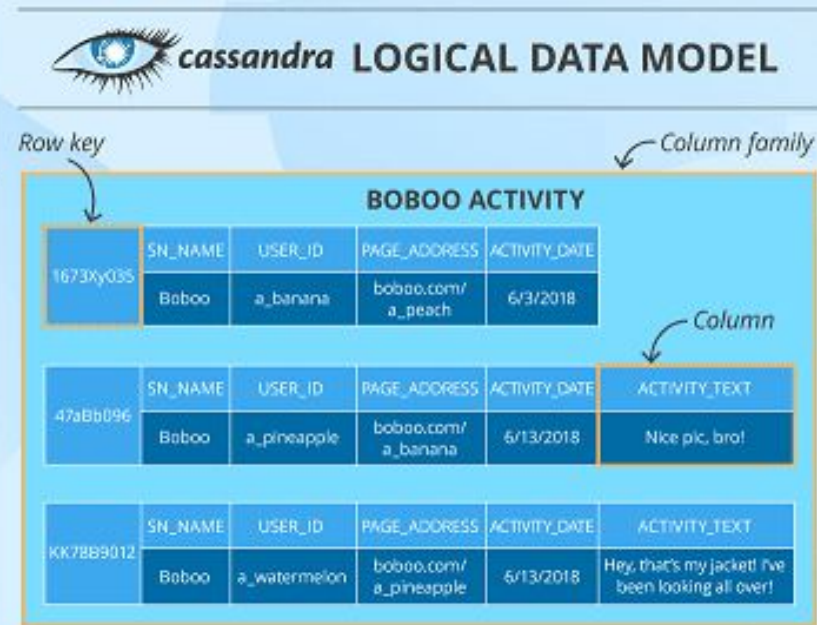
The diagram illustrates the HBase logical data model. It shows a table with a 'ROW KEY' column and several 'COLUMN FAMILIES'. The 'SOCIAL NETWORK' family contains 'USER ID' and 'PAGE ADDRESS'. The 'ACTIVITY' family contains 'TYPE', 'DATE', and 'TEXT'. Each row represents a specific activity, with the row key being a concatenation of the user ID and the page address. The 'CELL' label points to a specific data entry in the 'TEXT' column.

ROW KEY	SOCIAL NETWORK		ACTIVITY		
	USER ID	PAGE ADDRESS	TYPE	DATE	TEXT
Boboo%a_banana	a_banana	boboo.com/a_peach	Like	6/3/2018	
Boboo%a_pineapple	a_pineapple	boboo.com/a_banana	Comment	6/13/2018	Nice pic, bro!
Boboo%a_watermelon	a_watermelon	boboo.com/a_pineapple	Comment	6/10/2018	Hey, that's my jacket! I've been looking all over!
Chiching%a_cucumber	a_cucumber	chiching.com/a_kohlrabi	Comment	5/25/2018	Wow! What a bike!
Chiching%a_kohlrabi	a_kohlrabi	chiching.com/a_kohlrabi	Comment	5/25/2018	Yeah, I know :D

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись
- Table -> Column Family -> Row Key -> Column Qualifier -> Timestamp -> Value

**cassandra LOGICAL DATA MODEL**



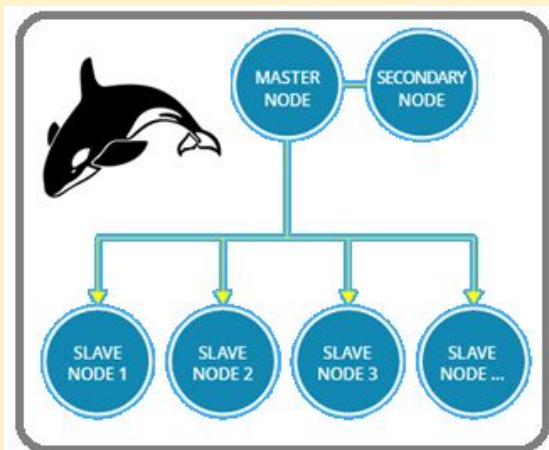
The diagram illustrates the Cassandra logical data model. It shows a table with a 'ROW KEY' column and several 'COLUMN FAMILIES'. The 'BOOBOO ACTIVITY' family contains 'SN\_NAME', 'USER\_ID', 'PAGE\_ADDRESS', 'ACTIVITY\_DATE', and 'ACTIVITY\_TEXT'. Each row represents a specific activity, with the row key being a concatenation of the user ID and the page address. The 'COLUMN' label points to a specific data entry in the 'ACTIVITY\_TEXT' column.

ROW KEY	BOOBOO ACTIVITY				
	SN_NAME	USER_ID	PAGE_ADDRESS	ACTIVITY_DATE	ACTIVITY_TEXT
1673xy035	Boboo	a_banana	boboo.com/a_peach	6/3/2018	
47aBb096	Boboo	a_pineapple	boboo.com/a_banana	6/13/2018	Nice pic, bro!
KK7BB9012	Boboo	a_watermelon	boboo.com/a_pineapple	6/13/2018	Hey, that's my jacket! I've been looking all over!

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

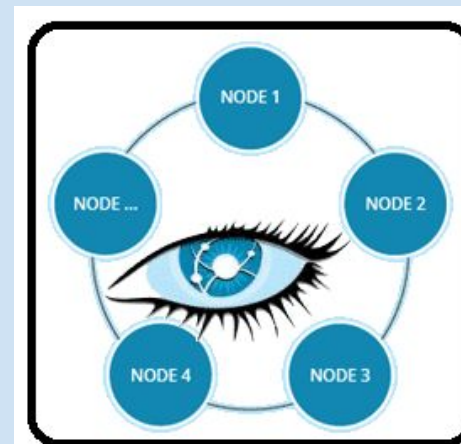
- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение
- Table -> Row Key -> Column Family -> Column Qualifier -> Timestamp -> Value
- Топология “звезда”



APACHE  
**HBASE**

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись
- Table -> Column Family -> Row Key -> Column Qualifier -> Timestamp -> Value
- Топология “кольцо”



*cassandra*

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение
- Table -> Row Key -> Column Family -> Column Qualifier -> Timestamp -> Value
- Топология “звезда”
- Требует Apache Zookeeper и Apache HDFS

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись
- Table -> Column Family -> Row Key -> Column Qualifier -> Timestamp -> Value
- Топология “кольцо”
- Не требует дополнительной инфраструктуры

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение
- Table -> Row Key -> Column Family -> Column Qualifier -> Timestamp -> Value
- Топология “звезда”
- Требует Apache Zookeeper и Apache HDFS
- Только первичный ключ

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись
- Table -> Column Family -> Row Key -> Column Qualifier -> Timestamp -> Value
- Топология “кольцо”
- Не требует дополнительной инфраструктуры
- Поддерживает вторичные индексы

# APACHE HBASE VS APACHE CASSANDRA

## Apache HBase

- Powerset, Java, 2007
- Быстрее на чтение и произвольное чтение
- Table -> Row Key -> Column Family -> Column Qualifier -> Timestamp -> Value
- Топология “звезда”
- Требует Apache Zookeeper и Apache HDFS
- Только первичный ключ

## Apache Cassandra

- Facebook, Java, 2008
- Быстрее на запись
- Table -> Column Family -> Row Key -> Column Qualifier -> Timestamp -> Value
- Топология “кольцо”
- Не требует дополнительной инфраструктуры
- Поддерживает вторичные индексы
- Транзакции на уровне записи

# ПРАКТИЧЕСКОЕ ЗАДАНИЕ



## ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Установите HBase
2. Создайте таблицу
3. Залейте 5 произвольных записей
4. Выполните одну операцию UPDATE
5. Выполните одну операцию DELETE
6. Выполните SCAN по всем записям

Будет плюсом добавлять данные в разные Column Family и Column Qualifier

**Спасибо!**

**Каждый день  
вы становитесь  
лучше :)**

