Московский государственный университет им. М.В. Ломоносова

Параллельные системы баз данных

Лекция 5. Архитектура параллельных систем баз данных

Разработчик:
Л.Б. Соколинский, д.ф.-м.н., профессор
E-mail: sokolinsky@acm.org
Южно-Уральский государственный университет

Направление 010300.68
«Фундаментальная информатика и информационные технологии»

Проект комиссии Президента по модернизации и техническому развитию экономики России
«Создание системы подготовки высококвалифицированных кадров в области суперкомпьютерных технологий и специализированного программного обеспечения»

Проект «Создание системы подготовки высококвалифицированных кадров в области суперкомпьютерных технологий и специализированного программного обеспечения»

Применение потенциала суперкомпьютерных технологий (СКТ) как значимой составляющей инновационного развития страны является задачей государственной важности, относится к приоритетному направлению и находится под постоянным контролем Президента и Правительства России. Одним из сдерживающих факторов развития страны в этом направлении является острая нехватка высококвалифицированных кадров в области СКТ, поскольку подготовка таких специалистов сейчас отсутствует как элемент системы высшего профессионального образования.

Стратегической целью проекта является создание национальной системы подготовки высококвалифицированных кадров в области суперкомпьютерных технологий и специализированного программного обеспечения.

Сайт проекта http://hpc-education.ru.

Содержание курса

1. Введение
2. Формы параллельной обработки транзакций
3. Определение параллельной системы баз данных
4. Классификация многопроцессорных систем
5. Архитектура параллельных систем баз данных
6. Организация межпроцессорных обменов
7. Балансировка загрузки в многопроцессорных иерархиях

Основная литература

1. *Taniar D., Leung C.H.C., Rahayu W., Goel S.* High Performance Parallel Database Processing and Grid Databases. John Wiley & Sons, 2008.
2. *Гарсиа-Молина Г., Ульман Дж., Уидом Дж.* Системы баз данных. Полный курс. М.: Издательский дом "Вильямс", 2004. 1088 с. (Раздел 15.9)
3. *Соколинский Л.Б.* Параллельные машины баз данных // Природа. Естественно-научный журнал Российской академии наук. -2001.№8. -С. 10-17.
4. *Соколинский Л.Б.* Обзор архитектур параллельных систем баз данных // Программирование. -2004.№6. -С. 49-63.
5. *Соколинский Л.Б.* Организация параллельного выполнения запросов в многопроцессорной машине баз данных с иерархической архитектурой // Программирование. -2001.№6. -С. 13-29.

Дополнительная литература

1. *Костенецкий П.С., Лепихов А.В., Соколинский Л.Б.* Технологии параллельных систем баз данных для иерархических многопроцессорных сред // Автоматика и телемеханика. -2007. - Том 68, №5. -С. 847-859
2. *Девитт Д., Грэй Д.* Параллельные системы баз данных: будущее высоко эффективных систем баз данных // СУБД. -1995.№2. -С. 8-31.
3. *Stonebraker M.* The case for shared nothing // Database Engineering Bulletin. -1986. -Vol. 9, No. 1. -P. 4-9.
4. *Graefe G.* Query evaluation techniques for large databases // ACM Computing Surveys. -1993. -Vol. 25, No. 2. -P. 73-169.

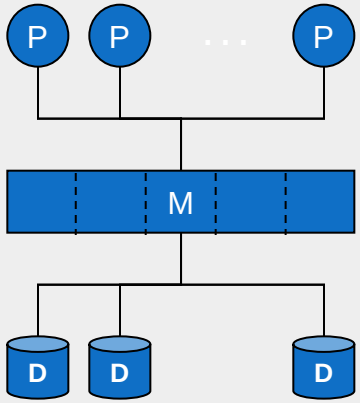
Классификация

- Классификация Стоунбрейкера
- Виртуально-иерархическая классификация

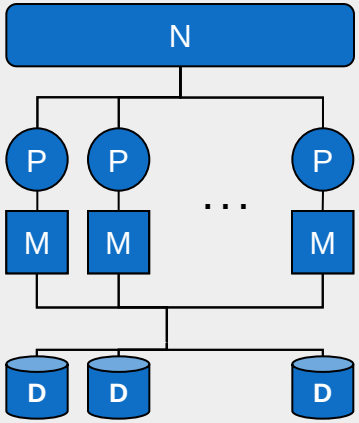
Классификация Стоунбрейкера

- *SE* (*Shared-Everything*) - архитектура с разделяемыми памятью и дисками
- *SD* (*Shared-Disks*) - архитектура с разделяемыми дисками
- *SN* (*Shared-Nothing*) - архитектура без совместного использования ресурсов

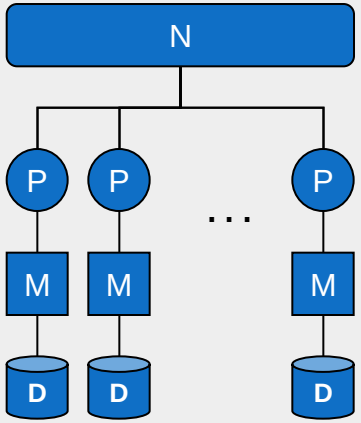
Классификация Стоунбрейкера



(а) SE

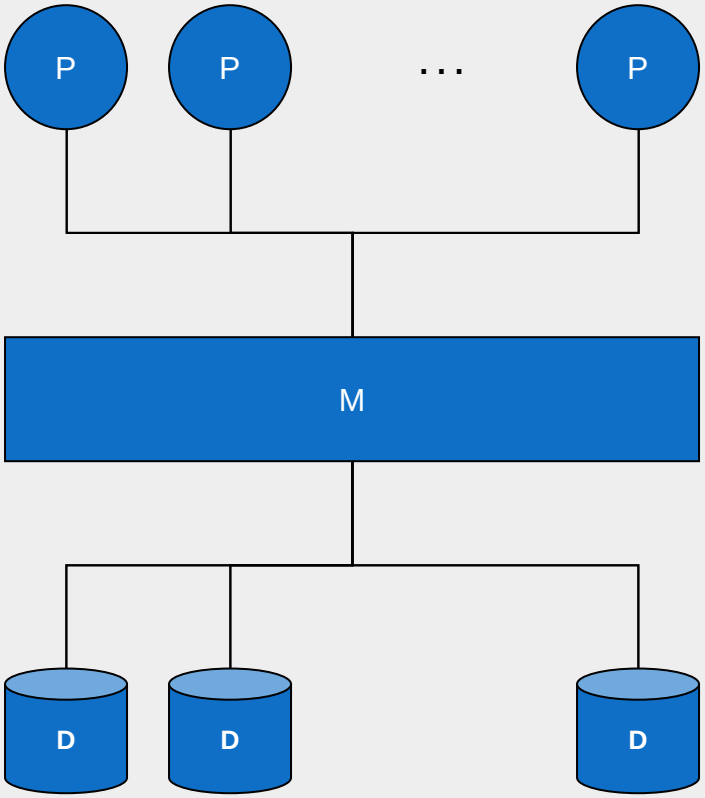


(б) SD



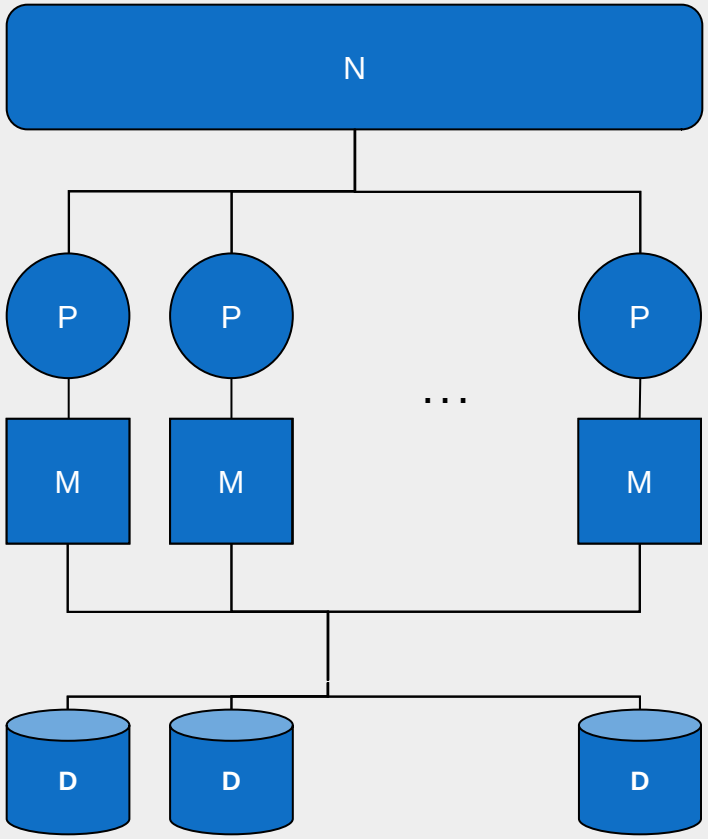
(в) SN

SE (*Shared-Everything*) - архитектура с разделяемыми памятью и дисками



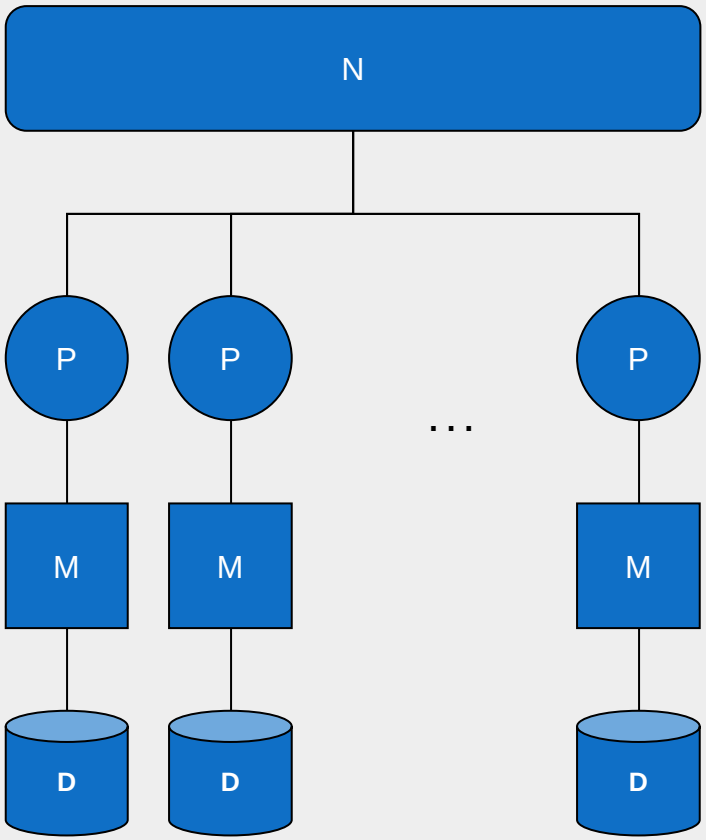
- Все процессоры разделяют общую оперативную память.
- Все диски напрямую доступны всем процессорам с одинаковым временем доступа.
- Межпроцессорные коммуникации осуществляются через общую оперативную память.
- Каждый процессор в SE системе имеет собственную кэш-память.
- Аппаратная платформа: SMP

SD (*Shared-Disks*) - архитектура с разделяемыми дисками



- Каждый процессор имеет свою приватную оперативную память.
- Все диски доступны всем процессорам с одинаковым временем доступа.
- Межпроцессорные коммуникации осуществляются через высокоскоростную соединительную сеть.
- Аппаратная платформа: MPP или кластеры с разделяемыми дисками

SN (*Shared-Nothing*) - архитектура без совместного использования ресурсов

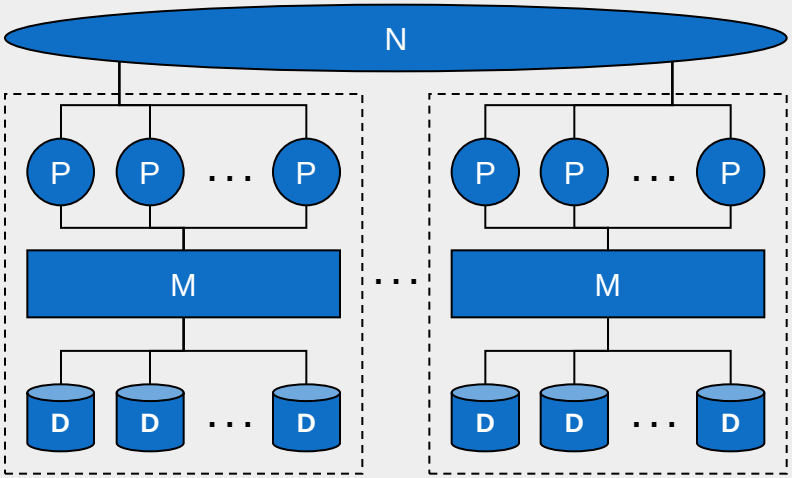


- Каждый процессор имеет свою приватную оперативную память и диск.
- Межпроцессорные коммуникации осуществляются через высокоскоростную соединительную сеть.
- Аппаратная платформа: MPP или кластеры с распределенными дисками

Виртуально-иерархическая классификация (расширение классификации Стоунбрейкера)

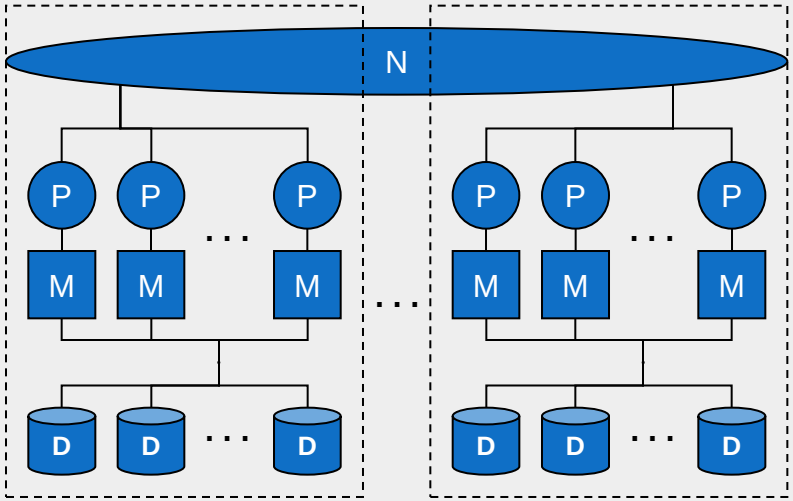
- *CE* (*Clustered-Everything*) – иерархическая архитектура с *SE* кластерами, объединенными по принципу *SN*.
- *CD* (*Clustered-Disk*) - иерархическая архитектура с *SD* кластерами, объединенными по принципу *SN*.
- *CDN* (*Clustered-Disk & Clustered-Nothing*) – иерархическая гибридная архитектура.
- …

CE (*Clustered-Everything*)



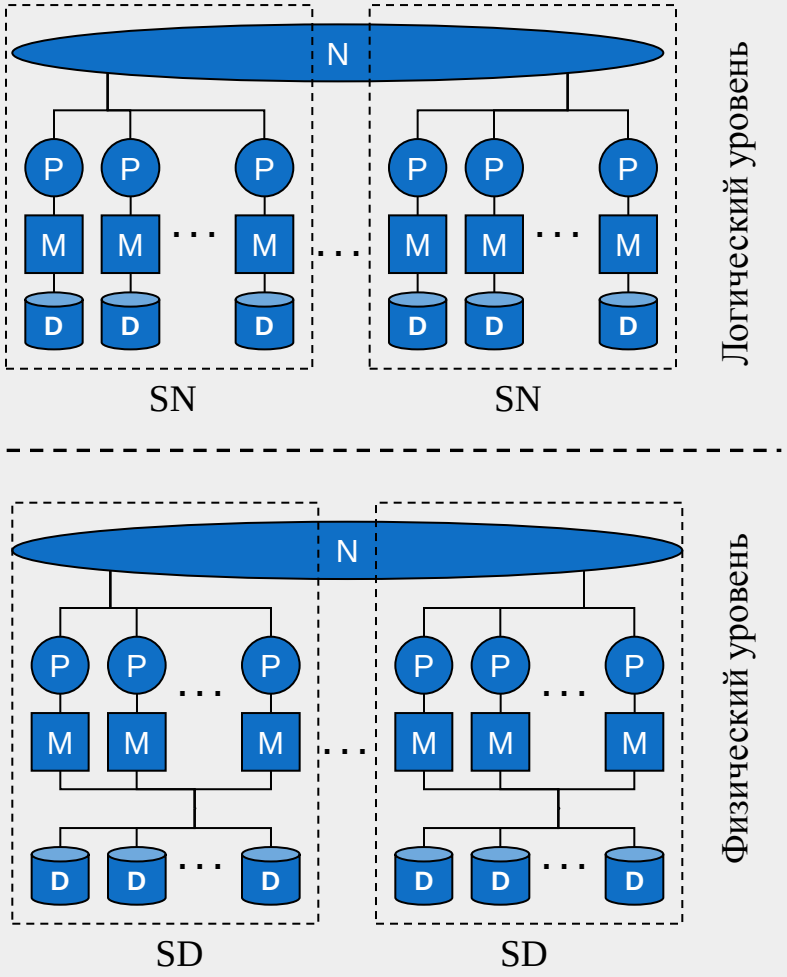
- *SE* кластеры объединены по принципу *SN*
- Межкластерныее коммуникации осуществляются через высокоскоростную соединительную сеть.
- Межпроцессорные коммуникации внутри *SE* кластера осуществляются через общую оперативную память.

CD (*Clustered-Disk*)



- *SD* кластеры объединены по принципу *SN*
- Граница *SD* кластеров распространены на общую (глобальную) соединительную сеть, так как в них может присутствовать собственная (локальная) соединительная сеть.
- Межпроцессорные коммуникации осуществляются через высокоскоростную соединительную сеть.

Гибридная архитектура C_D^N



Требования к параллельной системе баз данных

- высокая масштабируемость
- высокая производительность
- высокая доступность данных

Масштабируемость

- ускорение
- расширяемость

Ускорение

$\mathfrak{A}$ – множество различных конфигураций многопроцессорной системы

$\mathfrak{T}$ – набор тестов на производительность

$A,B \in \mathfrak{A}$

$Q \in \mathfrak{T}$

Коэффициент ускорения при переходе от A к B

$$a_{AB} = \frac{d_A t_{QA}}{d_B t_{QB}}$$

d_A - количество процессоров конфигурации A

d_B - количество процессоров конфигурации B;

t_{QA} - время, затраченное конфигурацией A на выполнение теста Q

t_{QB} - время, затраченное конфигурацией B на выполнение теста Q

Система демонстрирует *линейное ускорение*, если $a_{BC}=1$ для любых конфигураций B и C системы.

Расширяемость

$A_1, A_2, \ldots, A_n \in \mathfrak{A}$
$Q_1, Q_2, \ldots, Q_n \in \mathfrak{T}$
$|A_j|/|A_i| = |Q_j|/|Q_i|$, для всех $i,j = 1,\ldots,n$

Коэффициент расширяемости при переходе от A_k к A_m

$$e_{km} = \frac{t_{Q_k A_k}}{t_{Q_m A_m}}$$

Система демонстрирует *линейную расширяемость*, если $e_{km}=1$ для любых конфигураций системы.

Высокая масштабируемость

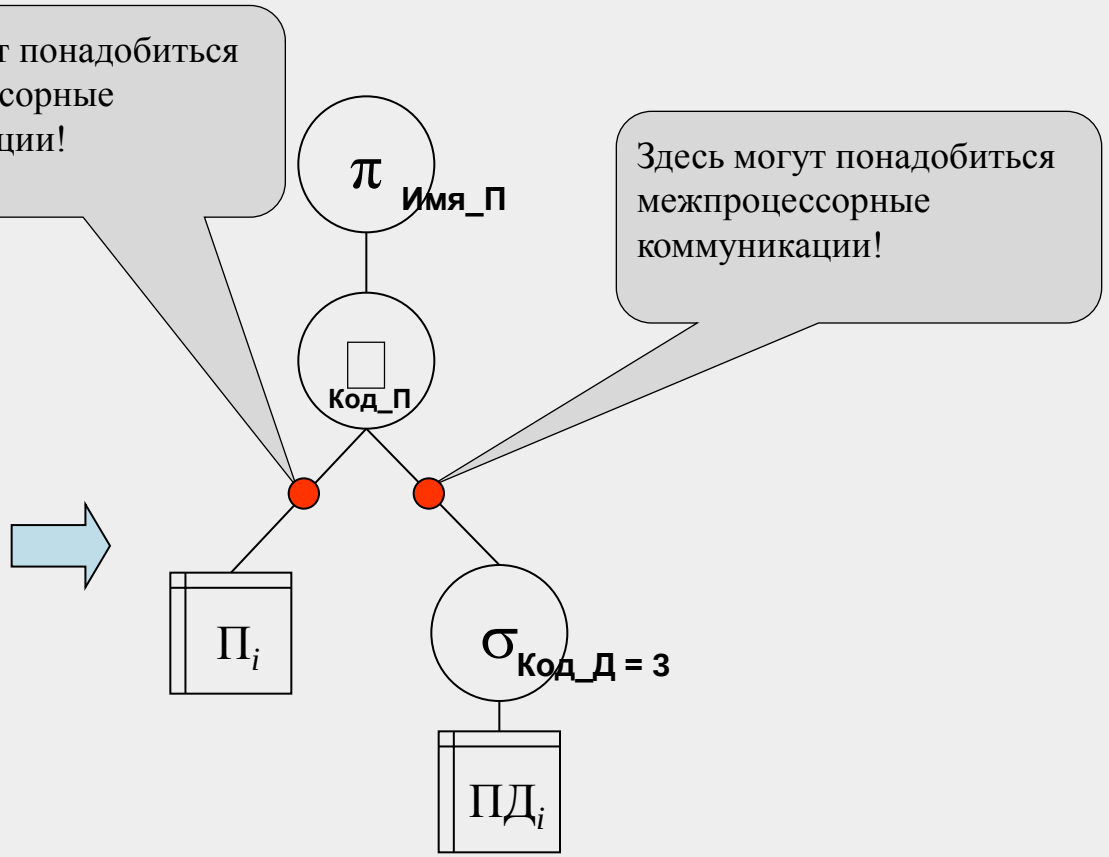
Параллельная система
хорошо масштабируема,
если она демонстрирует ускорение и расширяемость, близкие к линейным

Производительность

- Межпроцессорные коммуникации
- Когерентность КЭШей
- Организация блокировок
- Балансировка загрузки

Межпроцессорные коммуникации

```
/* Имена поставщиков,
поставляющих
деталь с кодом 3 */
SELECT Имя_П
FROM П, ПД
WHERE П.Код_П = ПД.Код_П
AND ПД.Код_Д = 3;
```

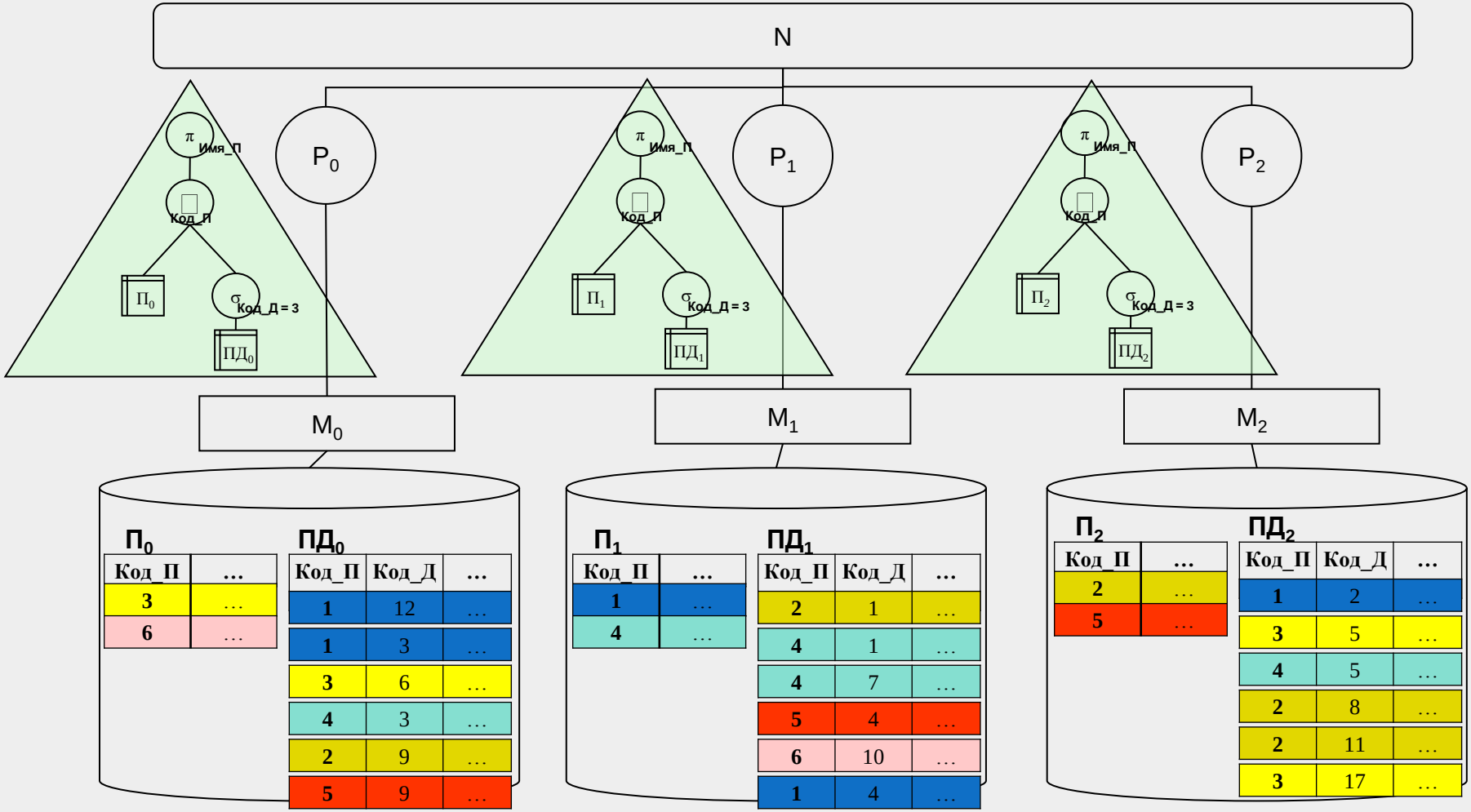


Пример, когда при выполнении запроса необходимы межпроцессорные коммуникации

$n=3$

$\varphi_{П}(x) = x.Код\_П \bmod 3$

$\varphi_{ПД}(x) = x.Код\_Д \bmod 3$

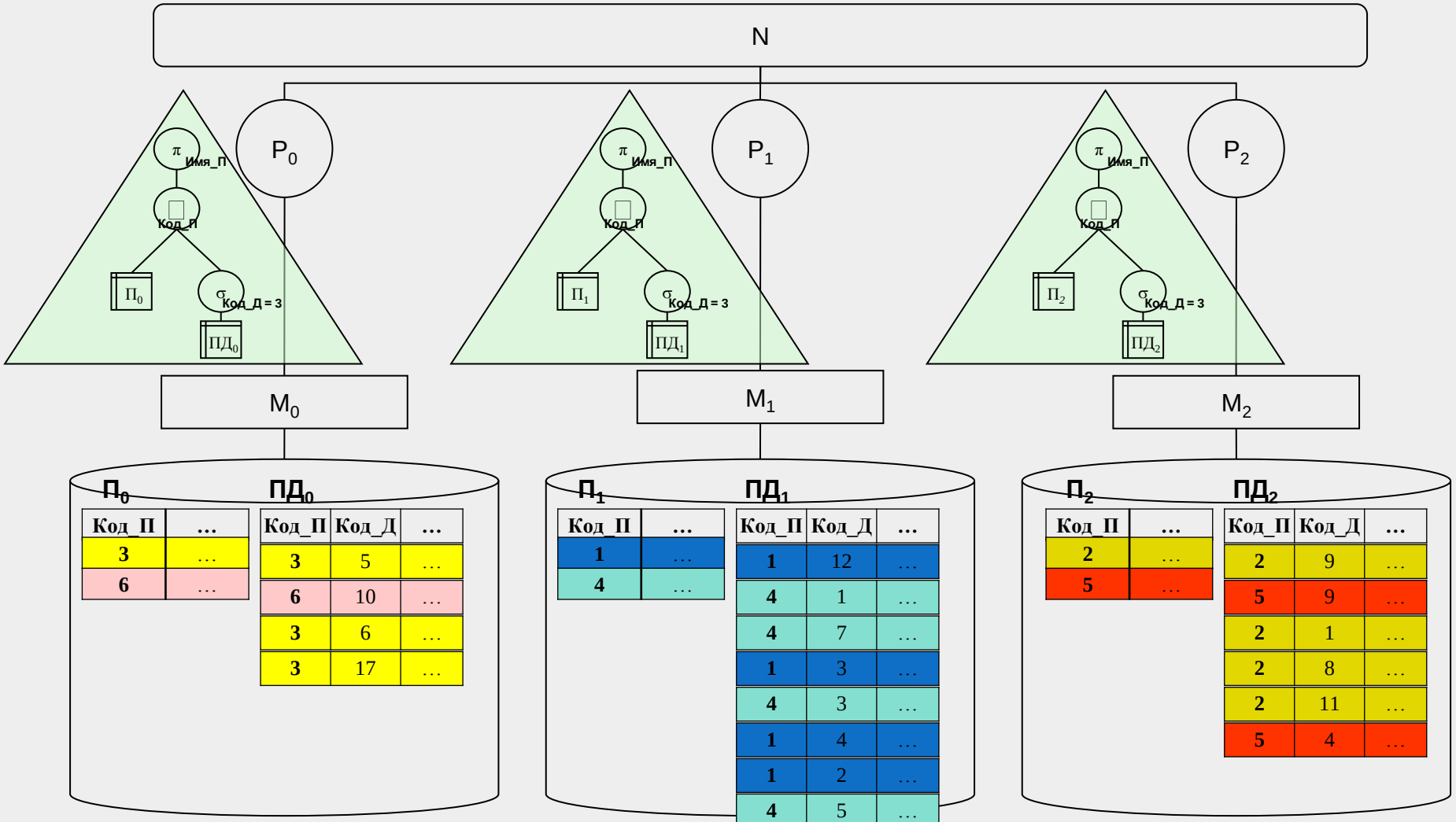


Пример, когда межпроцессорные коммуникации при выполнении запроса не нужны

$n=3$

$\varphi_{П}(x) = x.Код\_П \bmod 3$

$\varphi_{ПД}(x) = x.Код\_П \bmod 3$



Балансировка загрузки

- *Перекосы данных*
- *Перекосы выполнения*

Перекос данных

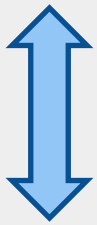
- *На одном узле находится больше обрабатываемых данных, чем на другом*

Перекосы выполнения

- *Один узел выполняет более трудоемкую операцию, чем второй*

Доступность данных

$$коэффициент\_доступности\_БД = \frac{реальное\_время\_доступности\_БД}{требуемое\_время\_доступности\_БД}$$



Аппаратная отказоустойчивость

Аппаратная отказоустойчивость

Вероятность отказа массовой компоненты

Вероятность отказа микросхемы памяти (в течении суток) ≈ 0.00001

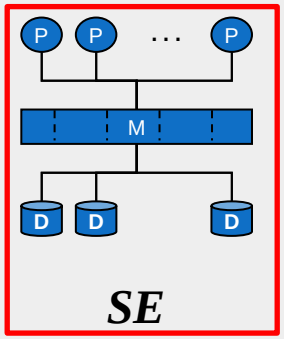
$\Rightarrow$

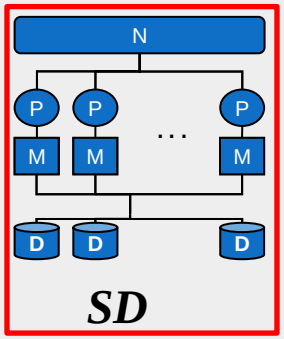
Вероятность отказа микросхемы памяти в системе с 10000 узлами, по 10 микросхем в каждом (в течении суток) ≈ 1

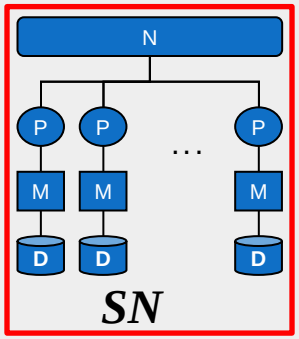
Высокая доступность данных

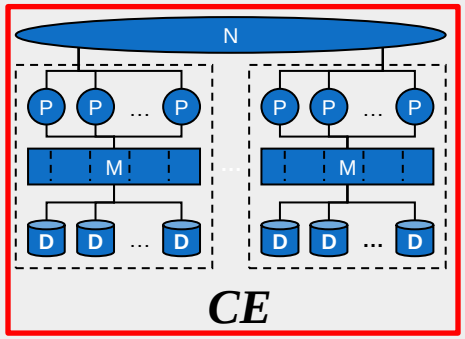
1) аппаратная отказоустойчивость
2) восстановление целостности базы данных после сбоя
3) оперативное восстановление базы данных
4) прозрачность для пользователя процессов восстановления системы

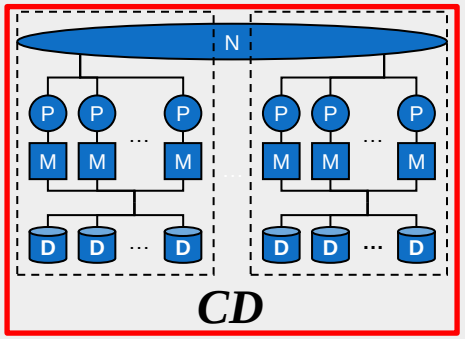
Сравнительный анализ архитектур



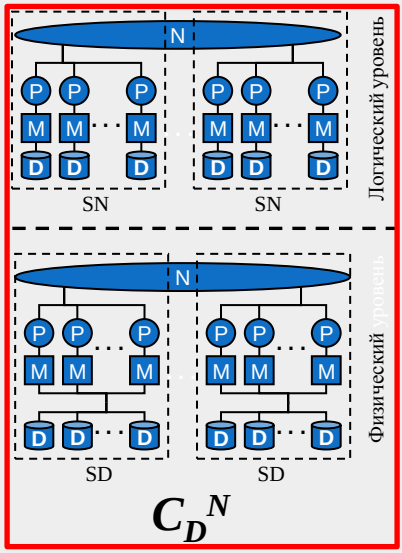








Критерий	Архитектура					
	SE	*SD*	*SN*	*CE*	*CD*	C_D^N
Масштабируемость	0	1	2	3	3	3
Доступность данных	0	1	3	1	2	2
Баланс загрузки	3	3	0	2	2	2
Межпроцессорные коммуникации	3	0	0	2	1	1
Когерентность кэшей	2	0	3	2	0	3
Организация блокировок	2	0	3	2	1	3
Сумма баллов	10	5	11	12	9	14



0 - "плохо“
1 - "удовлетворительно“
2 - "хорошо"
3 - "превосходно"

Наиболее перспективные архитектуры