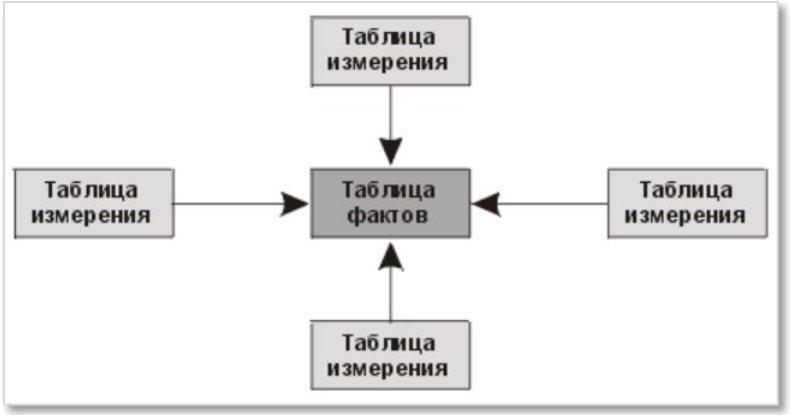
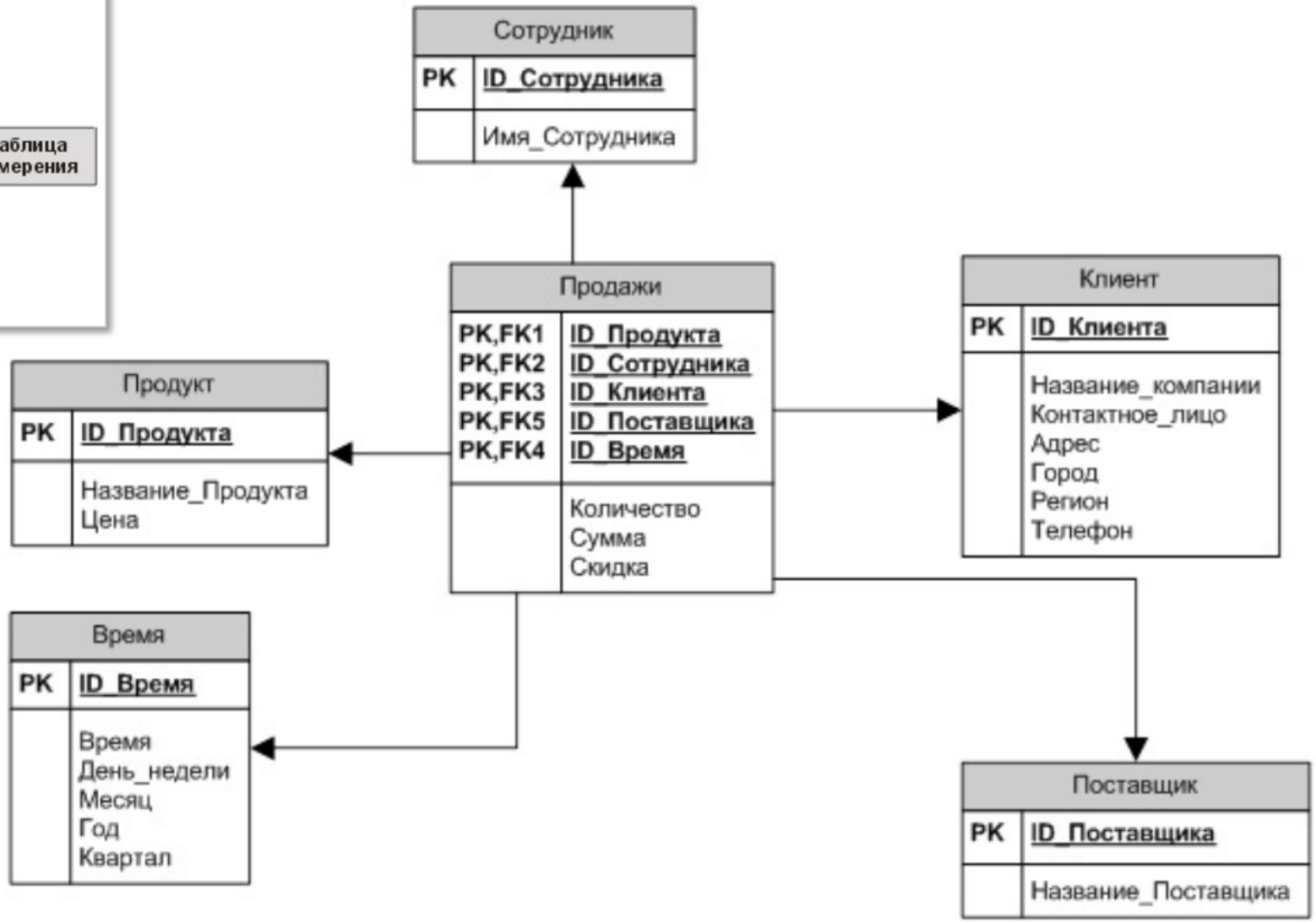
Методы и системы обработки больших данных

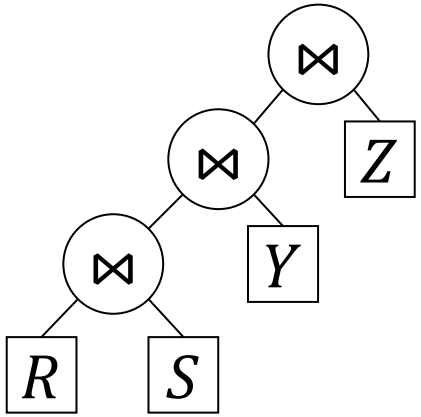
8. Порядок соединений

Схема данных «звезда»



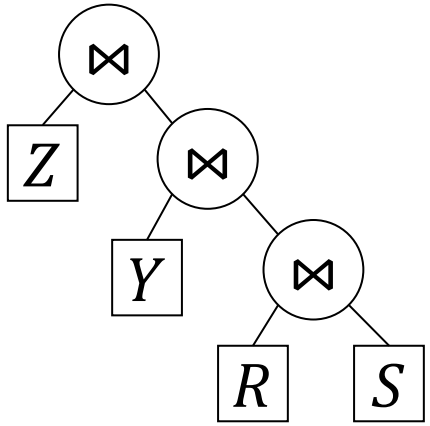


5 способов соединить 4 отношения $R \bowtie S \bowtie Y \bowtie Z$

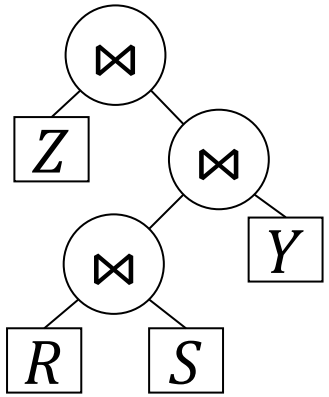


$((R \bowtie S) \bowtie Y) \bowtie Z$

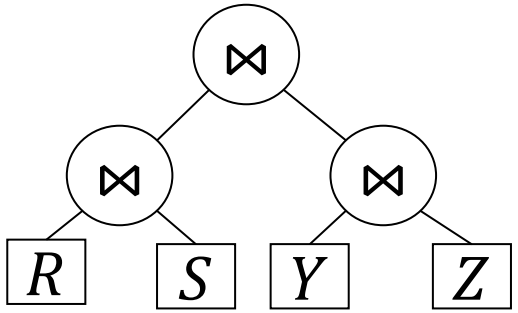
Схема БД:
$R(A, B)$
$S(C, A)$
$Y(D, A)$
$Z(E, A)$



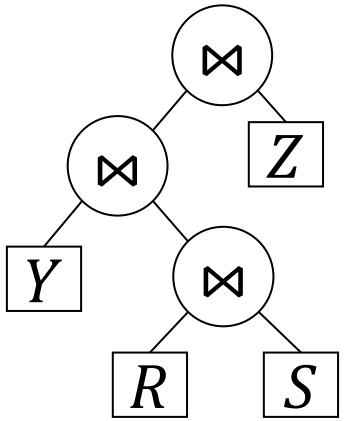
$Z \bowtie (Y \bowtie (R \bowtie S))$



$Z \bowtie ((R \bowtie S) \bowtie Y)$

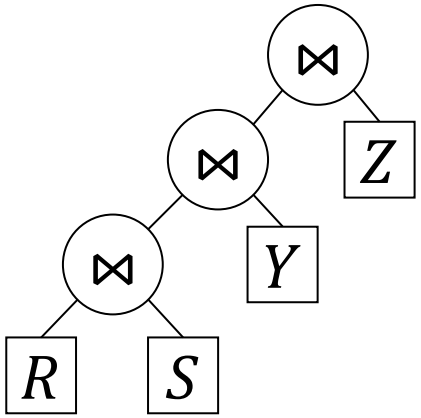


$(R \bowtie S) \bowtie (Y \bowtie Z)$

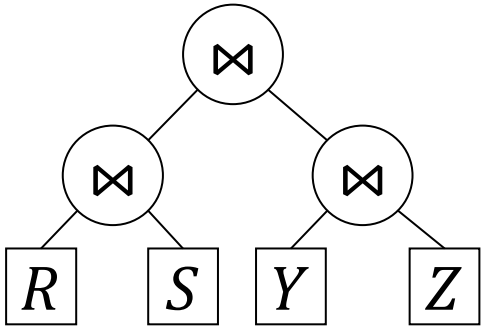


$(Y \bowtie (R \bowtie S)) \bowtie Z$

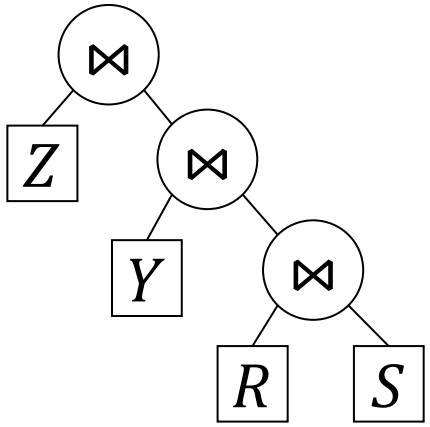
Виды бинарных деревьев



Левостороннее



Ветвящееся, сбалансированное



Правостороннее

- Бинарное дерево называется *левосторонним* (*right-deep*), если все его правые дочерние вершины являются листьями
- Бинарное дерево называется *правосторонним* (*left-deep*), если все его левые дочерние вершины являются листьями
- Бинарное дерево называется *ветвящимся* (*bushy*), если оно не является левосторонним или правосторонним
- Бинарное дерево называется *сбалансированным* (*balanced*), если все пути от корня до листьев имеют одинаковую длину (± 1)

Левосторонние деревья

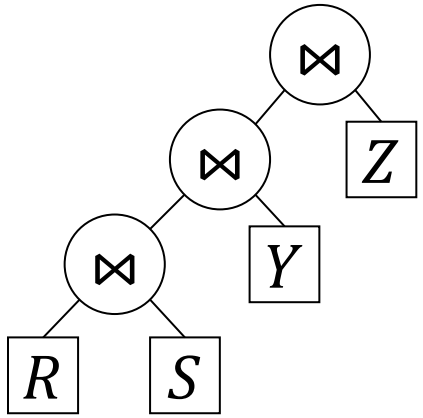


Схема БД:

$R(A^*, B)$

$S(C^*, A^\#)$

$Y(D^*, A^\#)$

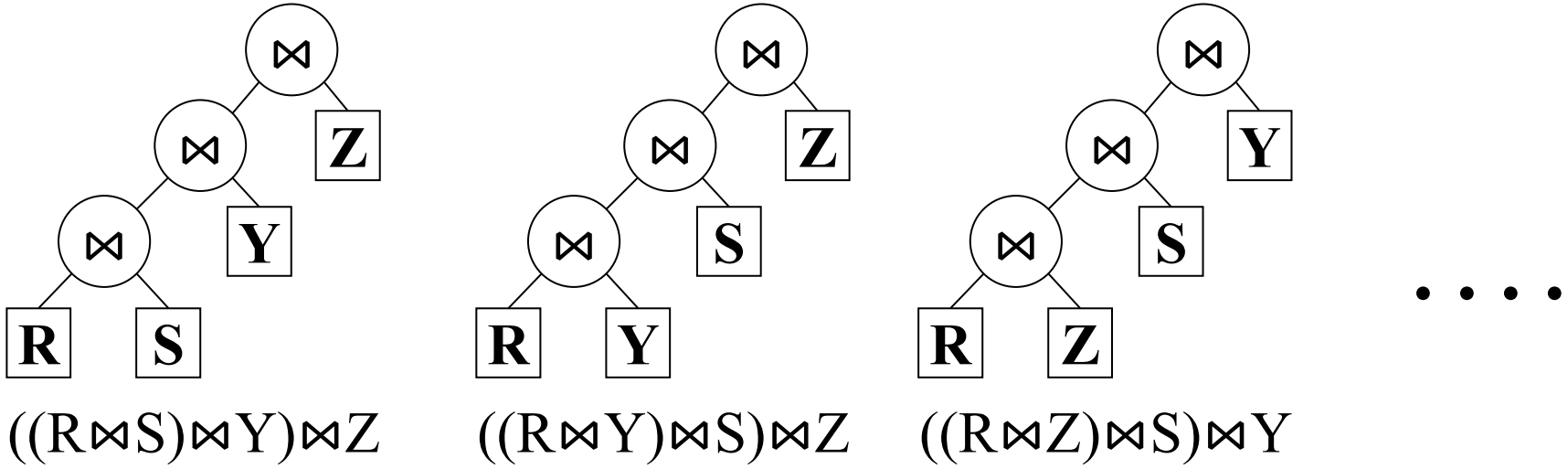
$Z(E^*, A^\#)$

В СУБД используются только левосторонние деревья, так как их структура хорошо согласуется с алгоритмами соединения.

* Первичный ключ

Внешний ключ

Выбор порядка соединения



- Общее количество вариантов для левосторонних деревьев: $4! = 24$
- Какой вариант выбрать?

Выбор порядка соединения

- «Жадный» алгоритм
- Динамическое программирование

«Жадный» алгоритм

Два варианта «жадного» алгоритма:

I. Сначала соединяются два наименьших отношения; промежуточный результат соединяется с наименьшим из оставшихся; и так далее…

II. Первым выполняется соединение, дающее минимальное результирующее отношение; промежуточный результат соединяется с одним из оставшихся отношений так, чтобы результат был минимален; и так далее…

Динамическое программирование

$R(A,B)$	$S(B,C)$	$Y(C,D)$	$Z(D,E)$
$T(R) = 1000$	$T(S) = 1000$	$T(Q) = 1000$	$T(Z) = 1000$
$V(R,A) = 100$			
$V(R,B) = 200$	$V(S,B) = 100$		
	$V(S,C) = 500$	$V(Y,C) = 20$	
		$V(Y,D) = 50$	$V(Z,D) = 1000$

$$T(R)T(S)/\max(V(R,B),V(S,B))$$

Стоимость определяется как сумма размеров **<u>всех</u>** промежуточных отношений.

	$\{R\}$	$\{S\}$	$\{Y\}$	$\{Z\}$
Размер	1000	1000	1000	1000
Стоимость	0	0	0	0
Наилучший план	R	S	Y	Z

	{R,S}	{R,Z}	{R,Y}	{S,Z}	{S,Y}	{Y,Z}
Размер	5000	1000000	1000000	1000000	2000	1000
Стоимость	0	0	0	0	0	0
Наилучший план	$R \bowtie S$	$R \bowtie Z$	$R \bowtie Y$	$S \bowtie Z$	$S \bowtie Y$	$Y \bowtie Z$

Динамическое программирование

$R(A,B)$	$S(B,C)$	$Y(C,D)$	$Z(D,E)$
$T(R) = 1000$	$T(S) = 1000$	$T(Y) = 1000$	$T(Z) = 1000$
$V(R,A) = 100$			
$V(R,B) = 200$	$V(S,B) = 100$		
	$V(S,C) = 500$	$V(Y,C) = 20$	
		$V(Y,D) = 50$	$V(Z,D) = 1000$

	$\{R,S\}$	$\{R,Z\}$	$\{R,Y\}$	$\{S,Z\}$	$\{S,Y\}$	$\{Y,Z\}$
Размер	5000	1000000	1000000	1000000	2000	1000
Стоимость	0	0	0	0	0	0
Наилучший план	$R \bowtie S$	$R \bowtie Z$	$R \bowtie Y$	$S \bowtie Z$	$S \bowtie Y$	$Y \bowtie Z$

$\{R,S,Y\}$			$\{R,S,Z\}$			$\{R,Y,Z\}$			$\{S,Y,Z\}$		
План	Стоимость	Размер	План	Стоимость	Размер	План	Стоимость	Размер	План	Стоимость	Размер
(R⋈S)⋈Y	5000	10000	(R⋈S)⋈Z	5000	5000000	(R⋈Y)⋈Z	1000000	1000000	(S⋈Y)⋈Z	2000	2000
(R⋈Y)⋈S	1000000	10000	(R⋈Z)⋈S	1000000	5000000	(R⋈Z)⋈Y	1000000	1000000	(S⋈Z)⋈Y	1000000	1000000
(S⋈Y)⋈R	2000	10000	(S⋈Z)⋈R	1000000	5000000	(Y⋈Z)⋈R	1000	1000000	(Y⋈Z)⋈S	1000	2000

Динамическое программирование

$\{R,S,Y\}$			$\{R,S,Z\}$			$\{R,Y,Z\}$			$\{S,Y,Z\}$		
План	Стоимость	Размер	План	Стоимость	Размер	План	Стоимость	Размер	План	Стоимость	Размер
(R⋈S)⋈Y	5000	10000	(R⋈S)⋈Z	5000	5000000	(R⋈Y)⋈Z	1000000	1000000	(S⋈Y)⋈Z	2000	2000
(R⋈Y)⋈S	1000000	10000	(R⋈Z)⋈S	1000000	5000000	(R⋈Z)⋈Y	1000000	1000000	(S⋈Z)⋈Y	1000000	1000000
(S⋈Y)⋈R	2000	10000	(S⋈Z)⋈R	1000000	5000000	(Y⋈Z)⋈R	1000	1000000	(Y⋈Z)⋈S	1000	2000

Выражение	Стоимость
$((S \bowtie Y) \bowtie R) \bowtie Z$	12000
$((R \bowtie S) \bowtie Z) \bowtie Y$	5005000
$((Y \bowtie Z) \bowtie R) \bowtie S$	1001000
$((\boldsymbol{Y} \bowtie \boldsymbol{Z}) \bowtie \boldsymbol{S}) \bowtie \boldsymbol{R}$	**3000**

Наилучший план

Конец лекции 8