

Распределенные системы и технологии.
Обнаружение неполадок и
объединение узлов в группы

Дмитрий Юрьевич Чалый
декан факультета ИВТ,
зав. кафедрой информационных и сетевых технологий



22 мая 2016 г.

Постановка задачи

Дано:

Распределенная система, состоящая из узлов x_1, x_2, \dots, x_n .

Пусть узел x_i вышел из строя.

Задача:

Обнаружить отказ узла x_i и распространить эту информацию по всей распределенной системе.



Актуальность задачи

- Пусть один узел выходит из строя за 10 лет с вероятностью 99,99%;
- найдем p – вероятность отказа за 1 час:

$$1 - (1 - p)^{10 \times 356 \times 24} = 0,9999$$

- откуда $p \approx 0,0002$;
- пусть в нашем датацентре 100 узлов, тогда за один час не произойдет отказа с вероятностью

$$p_{100} = 1 - p$$

- откуда получим вероятность отказа за t часов:

$$1 - (1 - p_{100})^{kt}$$

- например, при $t = 30 \times 24 = 720$ часов вероятность отказа составляет ≈ 0.9996 .



Обнаружение ошибок: варианты

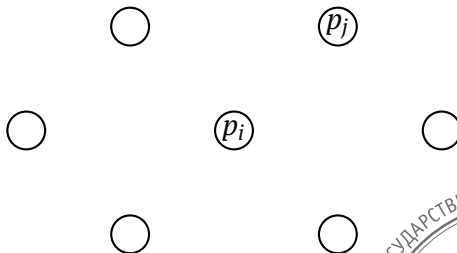
- нанять людей, чтобы они мониторили узлы;
- написать (распределенную) систему мониторинга.



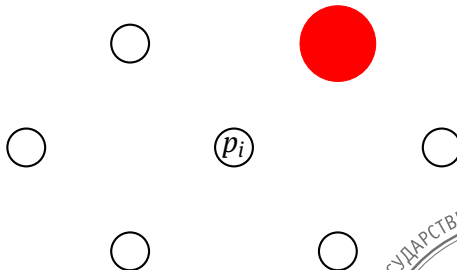
- Элементом является программный процесс p_i ;
- все процессы объединяются в логические группы;
- может случиться фатальный отказ процесса;
- обмен данными между процессами происходит через ненадежную среду передачи;
- важна масштабируемость, процессов может быть много.



Сценарий работы



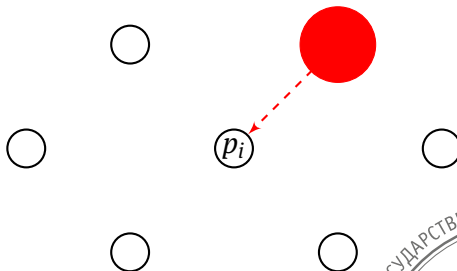
Сценарий работы



1 произошёл сбой и p_j «упал»;



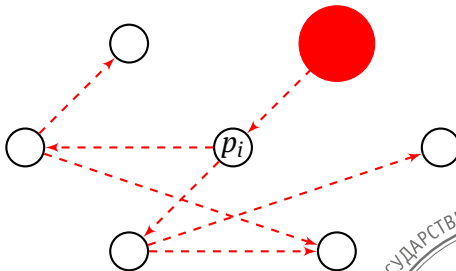
Сценарий работы



- 1 произошел сбой и p_j «упал»;
- 2 другой процесс p_i **обнаружил** этот факт...



Сценарий работы



- 1 произошел сбой и p_i «упал»;
- 2 другой процесс p_i **обнаружил** этот факт...
- 3 ...и **распространил** по всей системе.



Обнаружение отказов: свойства

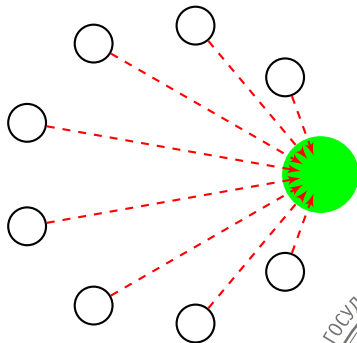
- **Точность:** не происходит ошибок в процессе обнаружения отказов;
- **полнота:** все отказы обнаруживаются;
- **скорость:** насколько быстро происходит обнаружение;
- **масштабируемость:** равномерная нагрузка на все процессы и коммуникационную сеть.

Точность и полнота одновременно не достижимы:

- гарантируем точность;
- вероятностная оценка полноты.



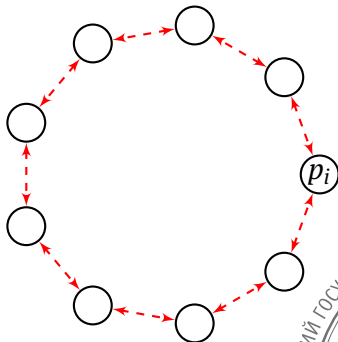
Централизованная архитектура



- каждый узел отправляет тактовый импульс через фиксированный промежуток времени;
- помечаем узлы как неработающие по тайм-ауту;
- достоинства и недостатки?



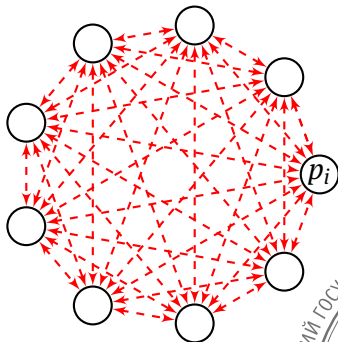
Кольцевая архитектура



- каждый узел отправляет тактовый импульс соседям слева и справа;
- достоинства и недостатки?



Архитектура «каждый-с-каждым»



- каждый узел отправляет тактовые импульсы всем остальным;
- достоинства и недостатки?



Рассылка импульсов по методу Gossip

- Каждый узел хранит таблицу вида:

Адрес узла	Счетчик узла	Локальное время
10.0.0.1	135	120
10.0.0.2	120	124

- периодически происходит рассылка этой таблицы другим членам группы;
- при получении таблицы от другого узла, локальная таблица обновляется.



Пример обновления информации на узлах



Адрес узла	Счетчик узла	Локальное время
10.1.1.12	35	78
10.1.2.23	34	70
10.1.1.13	25	72
10.1.2.18	29	65

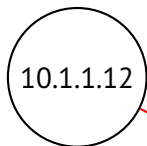
Локальное время – **78**

Адрес узла	Счетчик узла	Локальное время
10.1.1.12	33	120
10.1.2.23	35	117
10.1.1.13	30	125
10.1.2.18	26	110

Локальное время – **125**



Пример обновления информации на узлах



Адрес узла	Счетчик узла	Локальное время
10.1.1.12	35	78
10.1.2.23	34	70
10.1.1.13	25	72
10.1.2.18	29	65

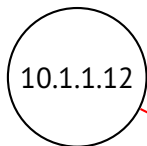
Локальное время – **80**

Адрес узла	Счетчик узла	Локальное время
10.1.1.12	33	120
10.1.2.23	35	117
10.1.1.13	33	140
10.1.2.18	26	110

Локальное время – **140**



Пример обновления информации на узлах



Адрес узла	Счетчик узла	Локальное время
10.1.1.12	35	78
10.1.2.23	34	70
10.1.1.13	25	72
10.1.2.18	29	65

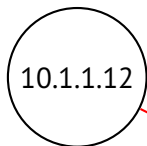
Локальное время – **85**

Адрес узла	Счетчик узла	Локальное время
10.1.1.12	33	120
10.1.2.23	35	117
10.1.1.13	33	140
10.1.2.18	26	110

Локальное время – **140**



Пример обновления информации на узлах



Адрес узла	Счетчик узла	Локальное время
10.1.1.12	35	78
10.1.2.23	34	70
10.1.1.13	25	72
10.1.2.18	29	65

Локальное время – **85**

Адрес узла	Счетчик узла	Локальное время
10.1.1.12	35	140
10.1.2.23	35	117
10.1.1.13	33	140
10.1.2.18	29	140

Локальное время – **140**



Когда помечать узел как отказавший?

По тайм-ауту:

- Если счетчик узла не обновлялся t_{fail} единиц времени, то узел можно пометить как отказавший;
- После $t_{cleanup}$ единиц времени после отказа можно удалить узел из группы.

