

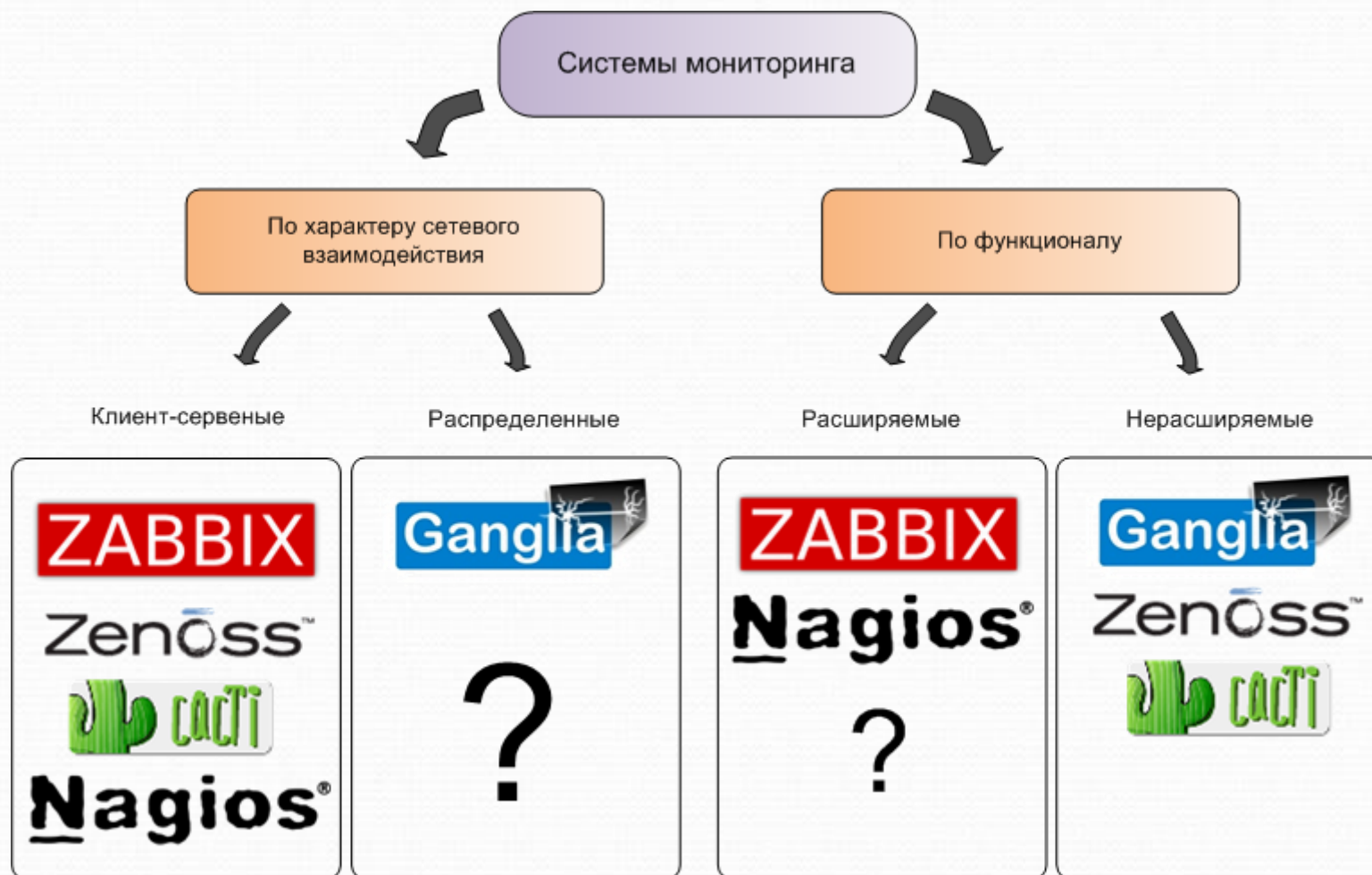
# Распределенная система мониторинга и диспетчеризации процессов гетерогенной среды

студент Костюков В.В.,  
профессор к.ф-м.н Крючкова Е.Н.,  
АлтГТУ / ПОВТ

# Требования



# Классификация



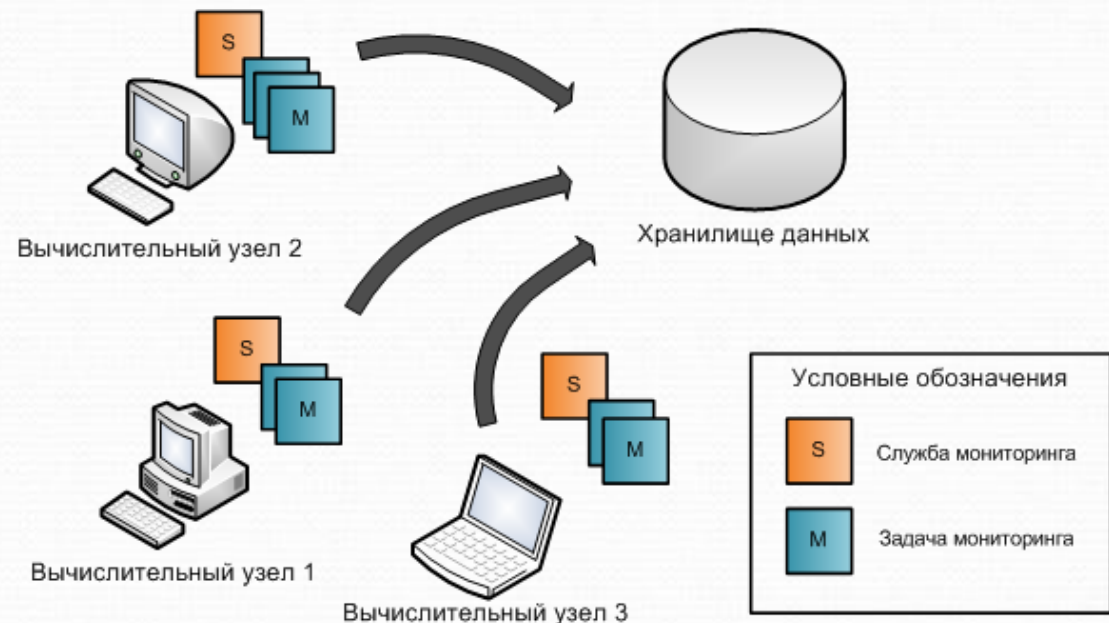
# Базовая терминология

**Узел** - программно-аппаратное устройство, способное исполнять код службы мониторинга.

**Служба** - активная сущность, непрерывно наблюдающая за состоянием узла и сохраняющая сообщения об изменении этого состояния в хранилище данных.

**Хранилище данных** - пассивная сущность, предоставляющая службам ресурсы для приема сообщений и их последующей обработки и хранения.

**Задача мониторинга** - шаблонная проблема получения и анализа некоторой информации о состоянии удаленного узла.

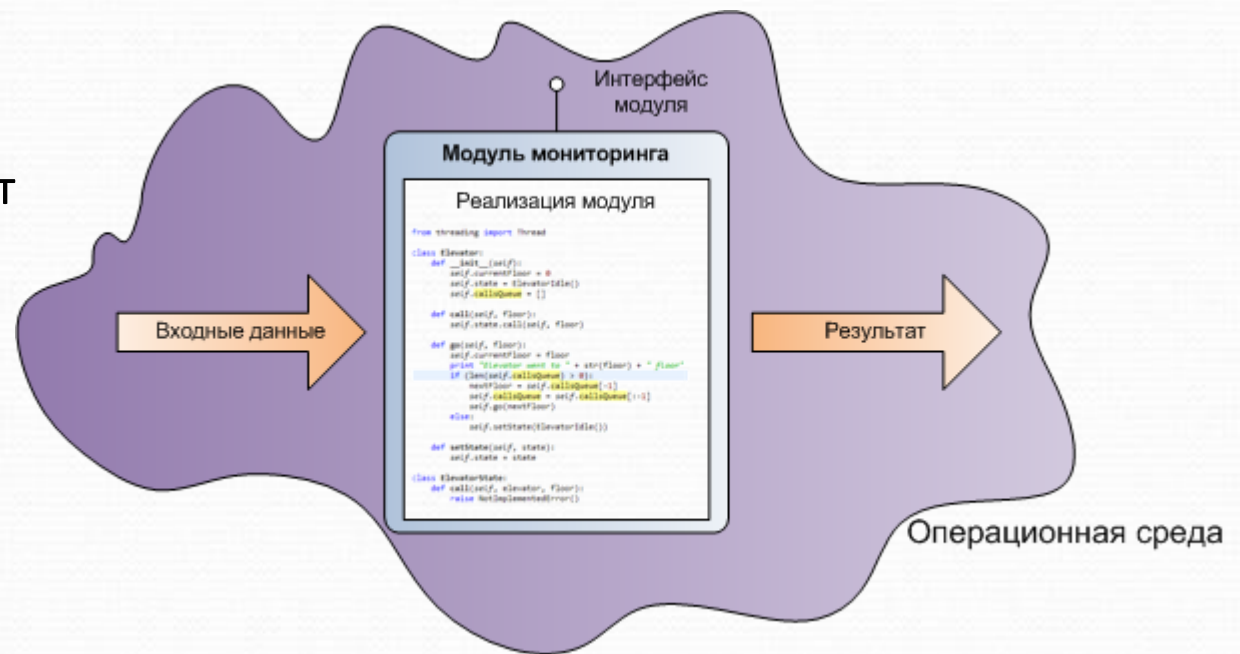




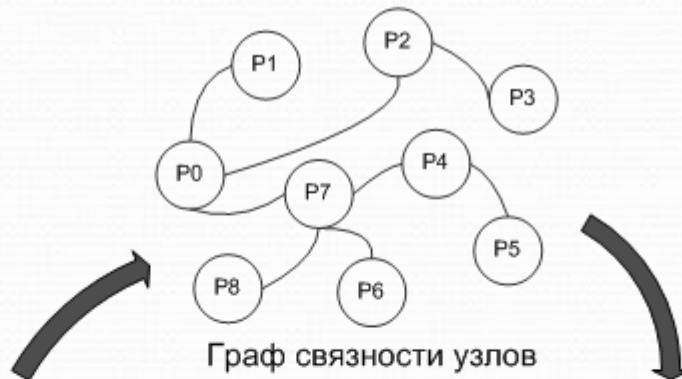
# Абстракция модуля

**Модуль** – это пятерка вида  $M = \{X, Di, Do, I, C\}$ , где

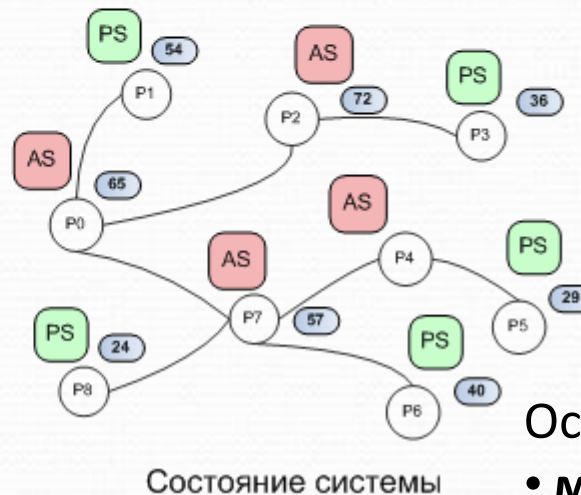
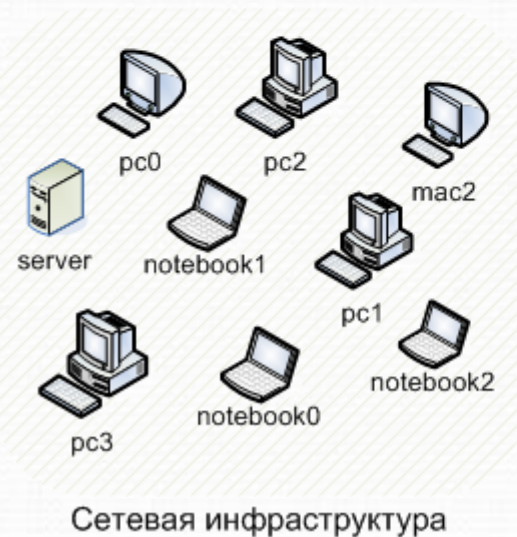
- $X \in \{\text{True}, \text{False}\}$   
(возможность исполнения в ОС),
- $Di$  и  $Do = \{a_0, a_1, \dots, a_n\}$ ,  $a_i$  – объект  
(входные и выходные данные),
- $I = \{b_0, b_1, \dots, b_m\}$ ,  $b_i$  – метод  
(интерфейс модуля),
- $C$  – программный код  
(реализация модуля)



# Состояние системы



Состояние распределенной системы определяется: **графом связности узлов**, расположением запущенных экземпляров **модулей** и **нагрузкой** на узлы;

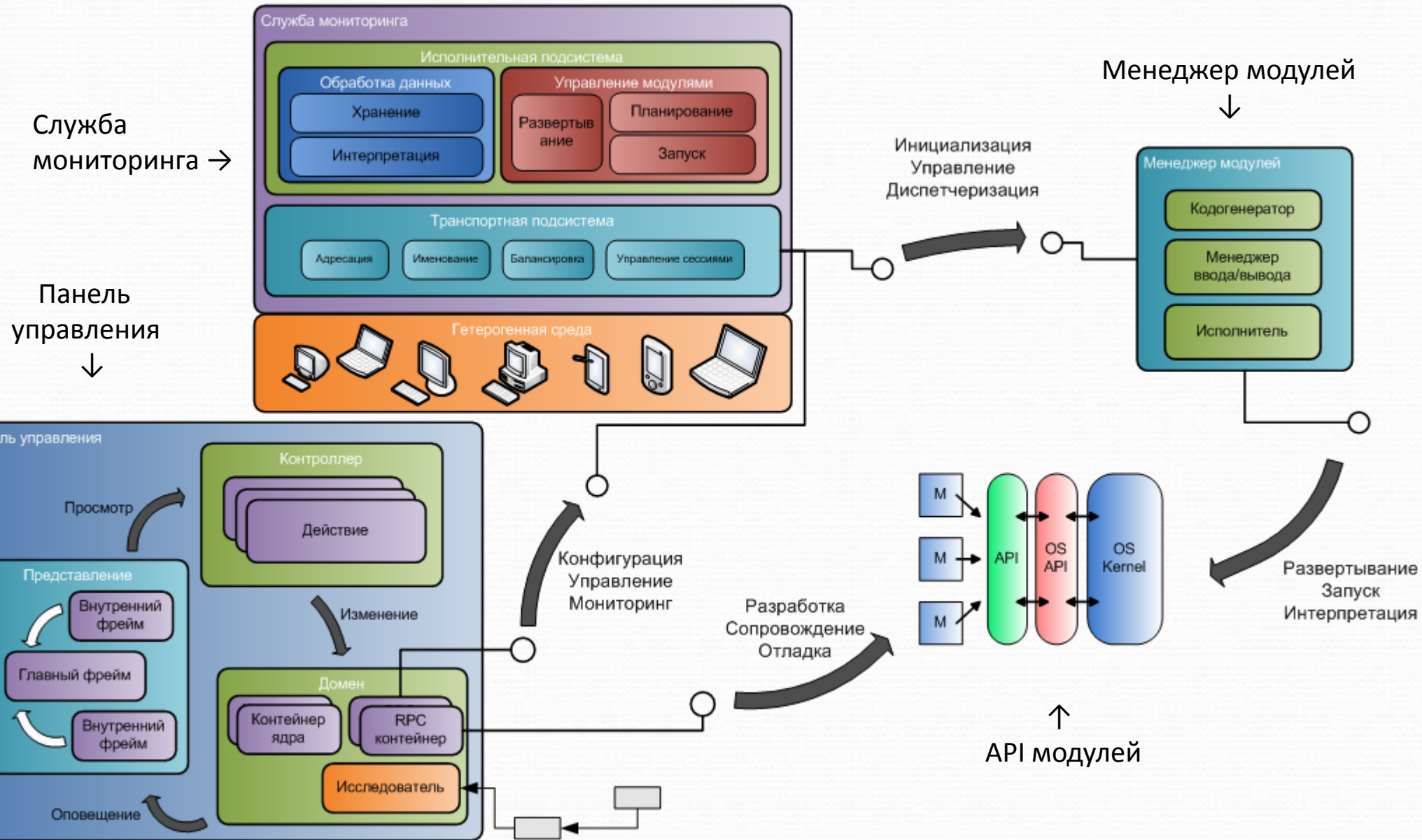


Роль распределенного модуля играет **служба мониторинга**, нагрузки на узел – **индекс производительности**;

Особенности службы мониторинга:

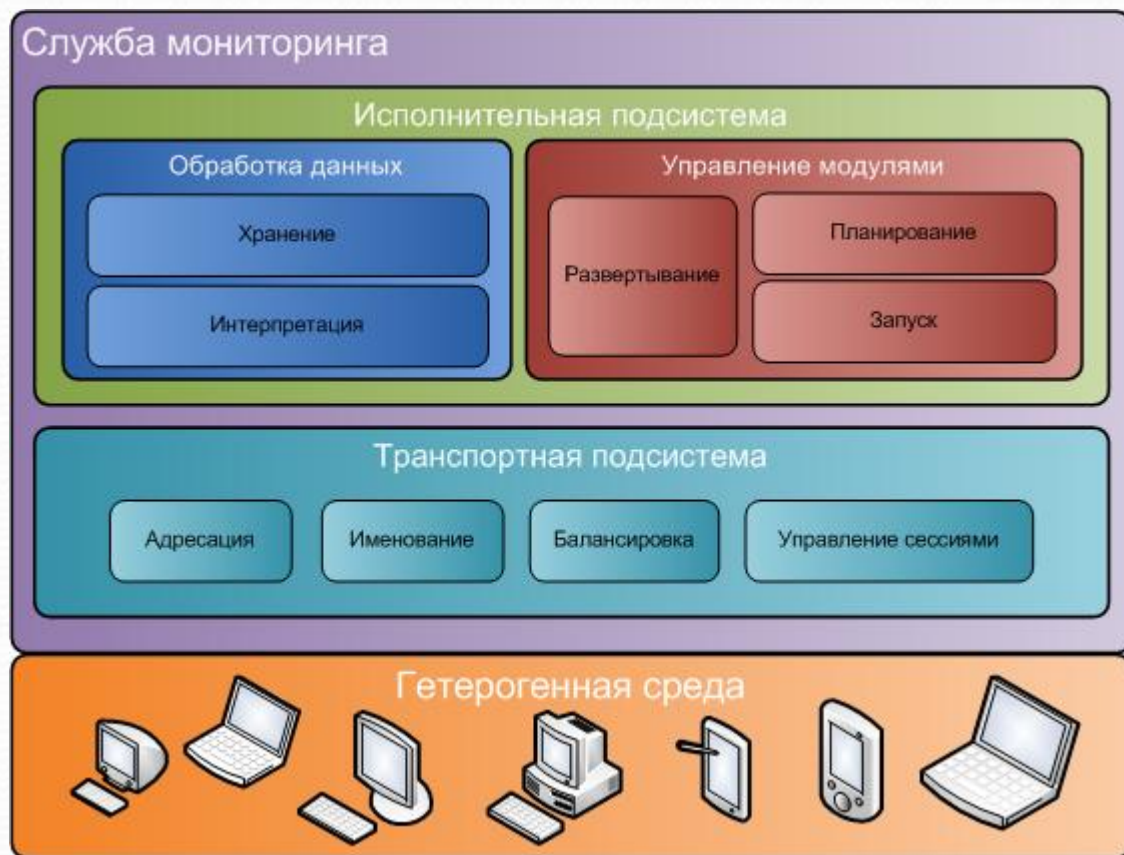
- масштабируемость;
- сериализуемость;
- переносимость;

# Структура системы





# Служба мониторинга



## Функции службы мониторинга:

- обеспечение работы основных подсистем;
- распределенная коммуникация;
- планирование и запуск модулей;

## Подсистемы службы мониторинга:

- платформа (ядро);
- транспортная;
- исполнительная;



# Ядро службы мониторинга

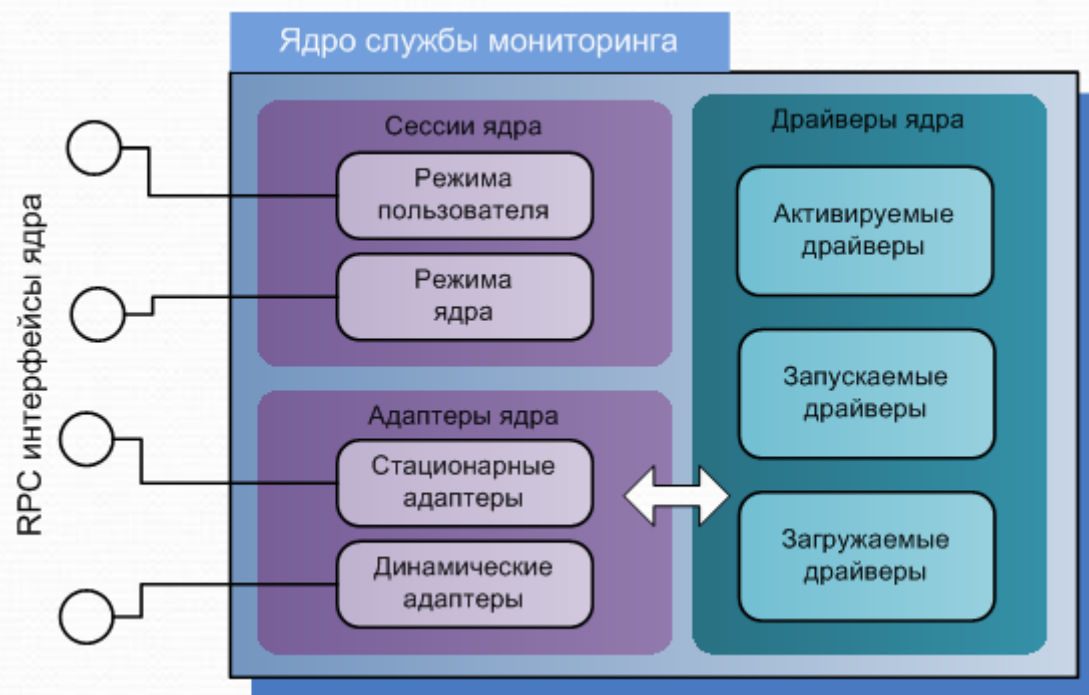
**Ядро** – набор примитивов и механизмов используемых подсистемами службы.

- **управление** драйверами;
- **генерация** событий;
- **управление** адаптерами;
- **инициализация** сессий;
- **синхронизация** потоков;

**Драйвер** – сущность расширяющая функционал ядра.

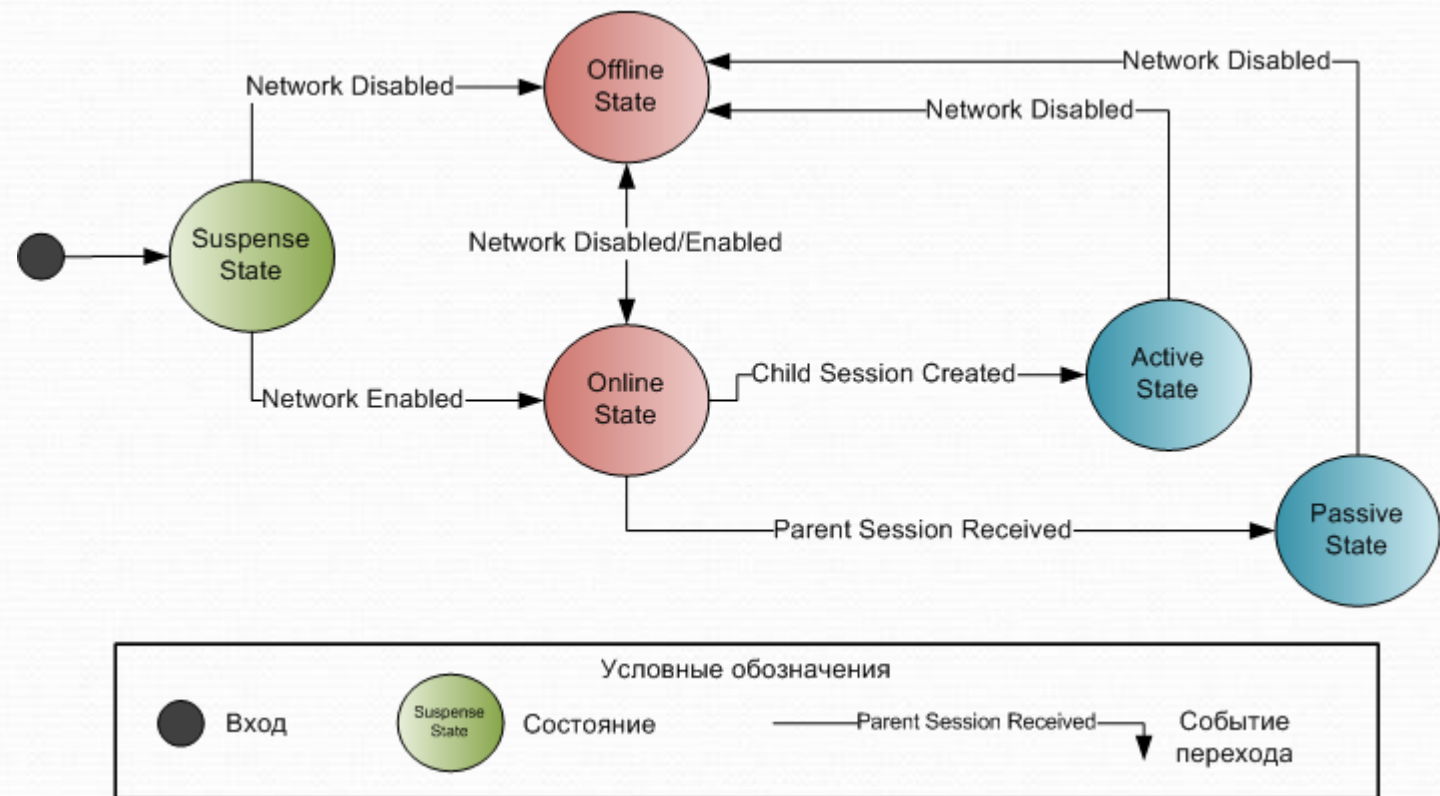
**Событие** –

- унифицированный протокол обмена данными между драйверами;
- механизм изменения состояния ядра;

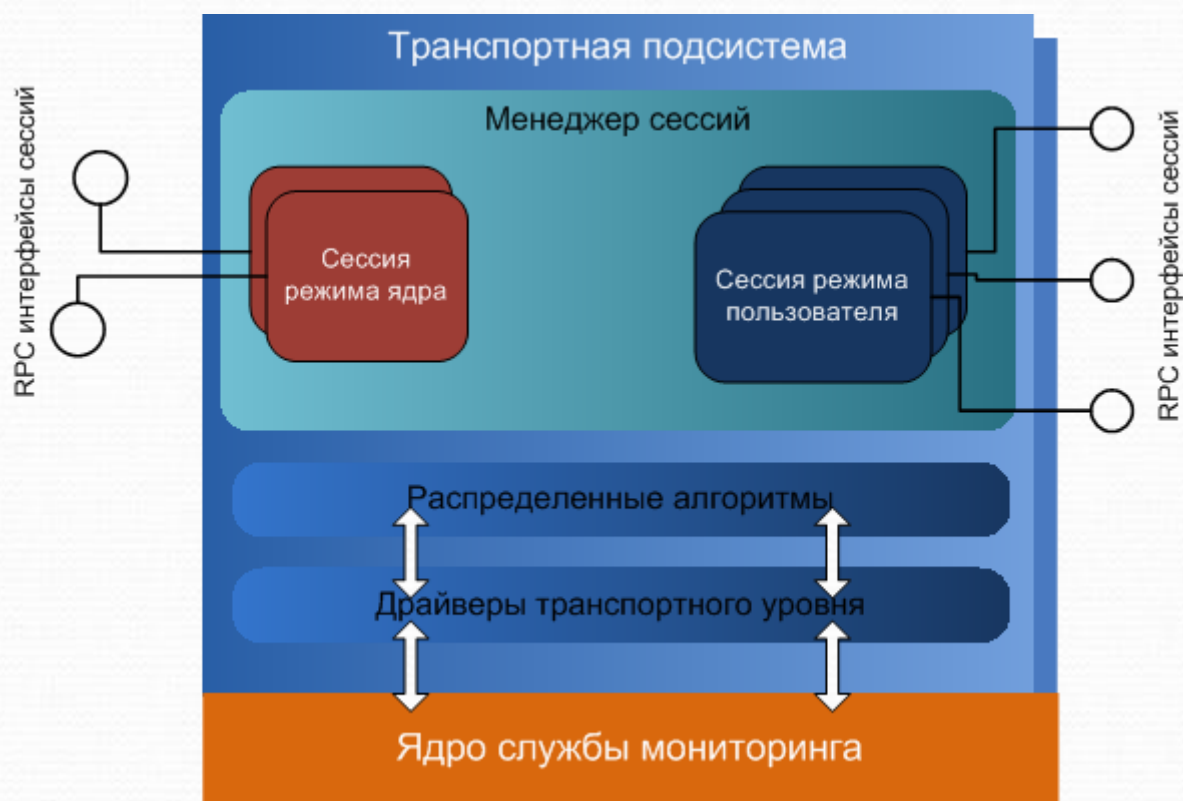


# Состояния ядра

- неопределенное;
- сетевое;
- автономное;
- активное;
- пассивное;



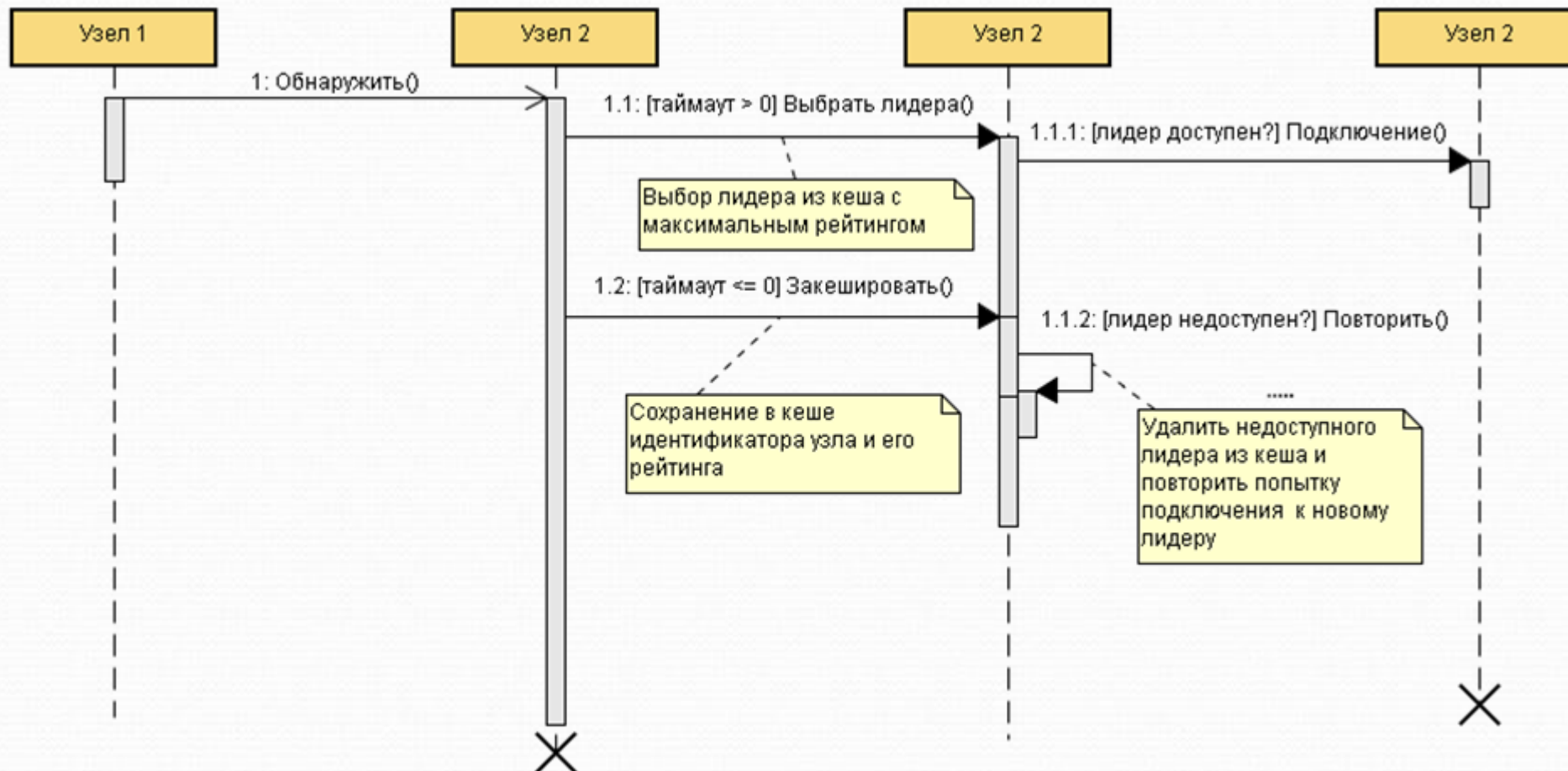
# Транспортная подсистема



- **управление** сессиями;
- **мониторинг** сетевой активности;
- **именование** объектов;
- **адресация**;
- **балансировка** нагрузки;
- **выбор лидеров**;

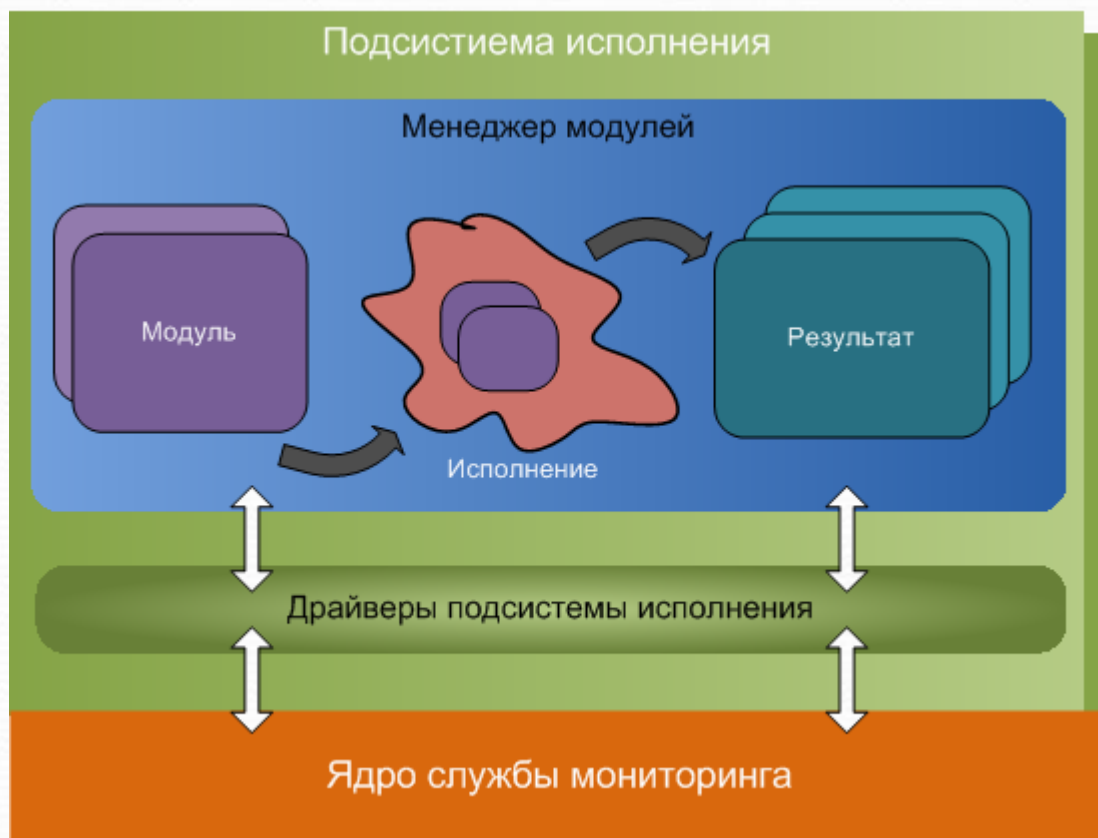


# Алгоритм выбора лидера



# Подсистема исполнения

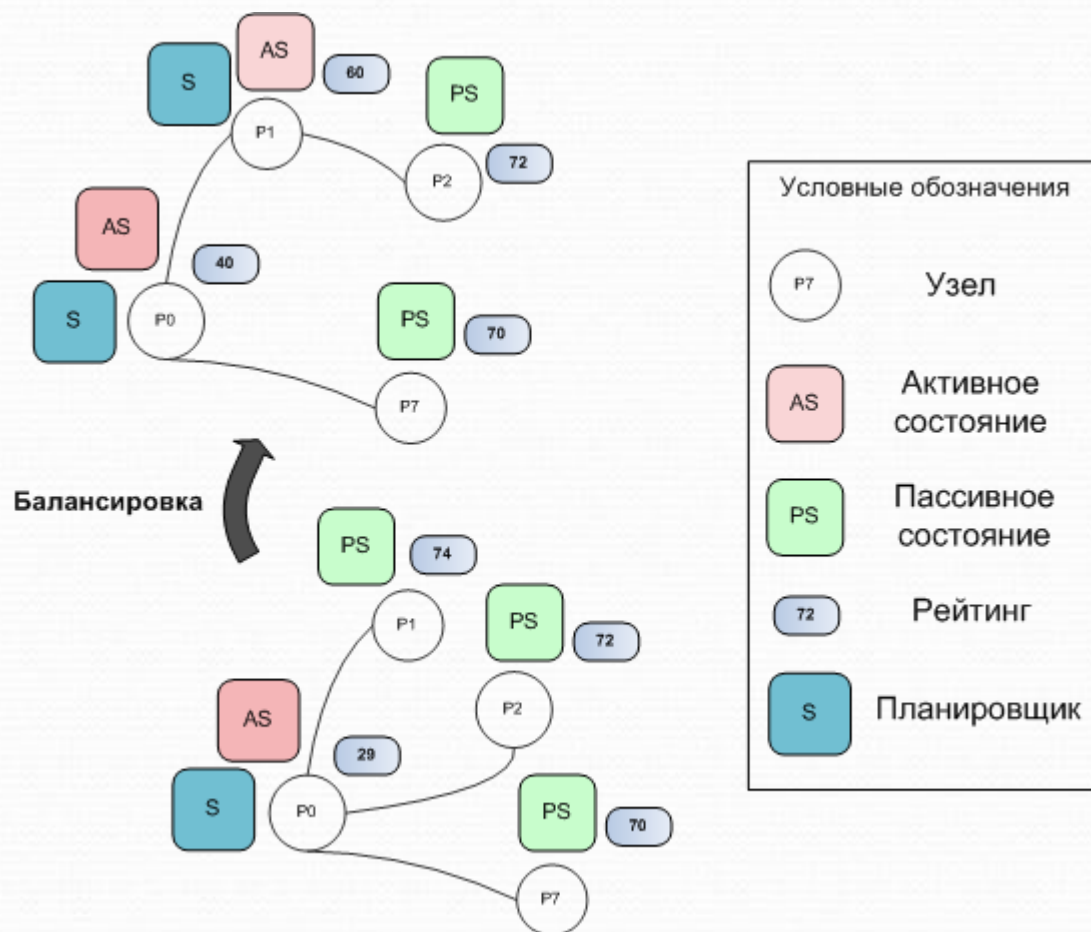
- **планирование** запусков;
- **запуск** модулей мониторига;
- **обработка** результатов;
- **развертывание** модулей;



# Планировщик подсистемы исполнения

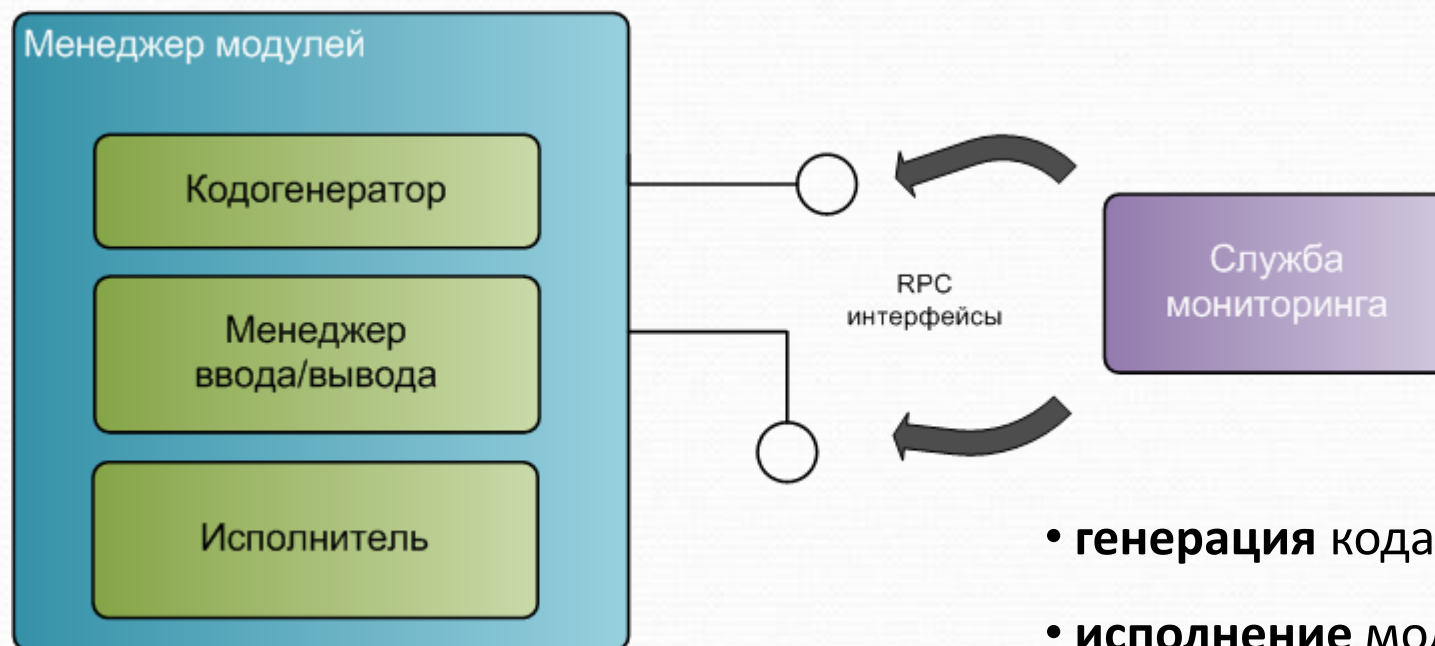
**Планировщик** – запускаемый драйвер ядра, обеспечивающий автономный запуск модулей мониторинга.

- запуск по расписанию;
- принудительный запуск;
- делигирование;
- сериализуемость;



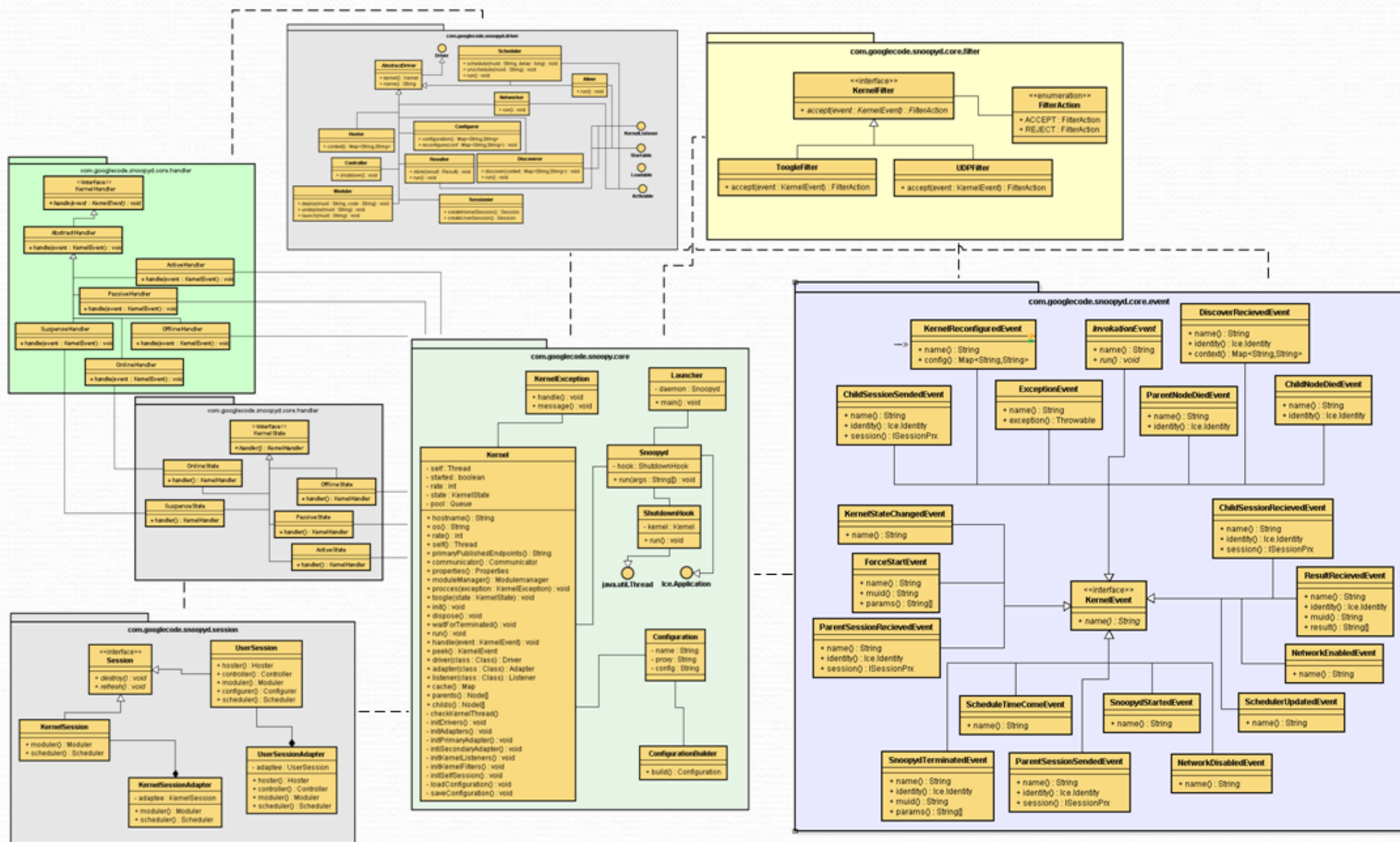


# Менеджер модулей мониторинга



- **генерация** кода каркаса;
- **исполнение** модулей в ОС;
- **выполнение** файловых операций;

# Архитектура службы мониторинга



# Итоги

- разработана **модель**;

Модель распределенной системы с динамически расширяемым функционалом;

- спроектирована **архитектура**;

Архитектура высоконагруженной распределенной системы мониторинга;

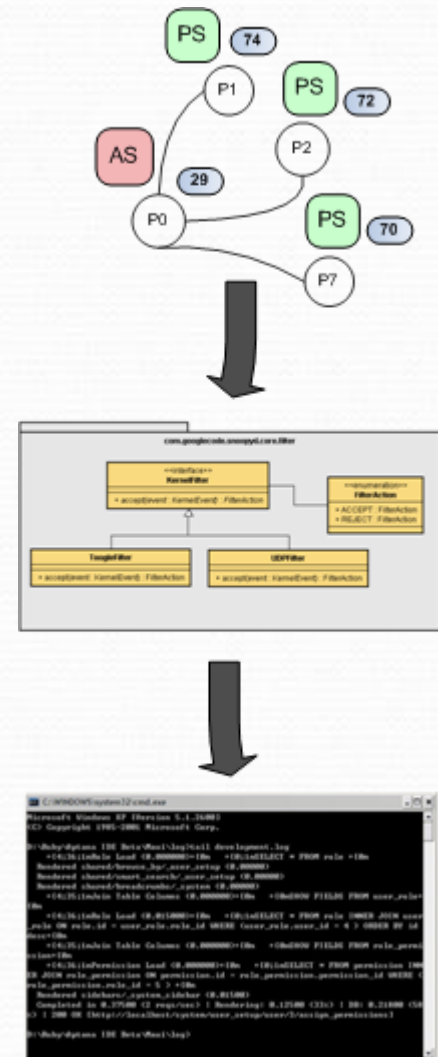
- реализованы **приложения**;

Кроссплатформенное сервисное приложение распределенной **службы** мониторинга состоящее из:

- ядра / платформы;
- подсистемы исполнения;
- транспортной подсистемы;

Встраиваемое приложение **менеджера модулей** мониторинга состоящее из:

- кодогенератора;
- подсистемы ввода/вывода;
- исполнителя;





# Пути развития проекта

- **разработка** шаблонных модулей мониторинга для решения круга повседневных задач (анализ сетевого трафика, загрузка и температура процессора, количество свободной памяти и т.д.);
- **оформление** технической документации и спецификаций программного кода;
- **совершенствование** компонентов и оптимизация алгоритмов базовой платформы;
- полномасштабное **внедрение** и нагрузочное тестирование системы на базе существующей инфраструктуры предприятия, например лаборатории МикроЭВМ АлтГТУ;

Спасибо!  
Вопросы?

<http://snoopy.googlecode.com>