

Актор, запускающий

команду. Может быть

живым человеком с определённой ролью

администратора,

менеджера или клиента,

мобильным

приложением или

умным чайником.

Действие, которое совершает система. Описывается в

настоящем времени и говорит о том, «что» произойдёт, а не «как». Checkout может быть

командой, потому что говорит о том «что», a params validator, produce event и store user

Событие, которое является

действия, которое нельзя отменить. То есть account registered — событие, a create user, user maybe created — нет. Событие говорит как о положительном, так и об отрицательном результате.

Может вызвать другую команду иногда через policy.

Полиси (policy), условие при котором система автоматически запустит команду.

Система автоматически запустит команду. Иногда для вызова команды через событие необходимо соблюдение какого-то условия. Например, для вызова команды отправки заказа в доставку надо проверить, что заказов собралось больше трёх штук. За подобные условия отвечает policy. Иногда под полиси ошибочно подразумевают условие в логике работы команды, например когда нотификация отправляется при успешной оплате заказа. В данном случае это часть бизнес логики, а не условие, при котором система запустит бизнесовую команду как реакцию на событие. Следовательно, такое условие, не может считаться отдельным стикером в ES.

Внешние системы (external system).

Другие системы, с которыми взаимодействует наша, например эквайринг, который как-то списывает деньги. Здесь неважно, из чего состоит эквайринг, важно указать, просто что он есть. Бывают ситуации, когда система ничего не получает на вход. Пример: крон говорит, что нужное время наступило, или гитхаб, который говорит о том, что пользователь отправил коммит в нужный репозиторий (через вебхук). Всегда возвращает событие как результат работы.

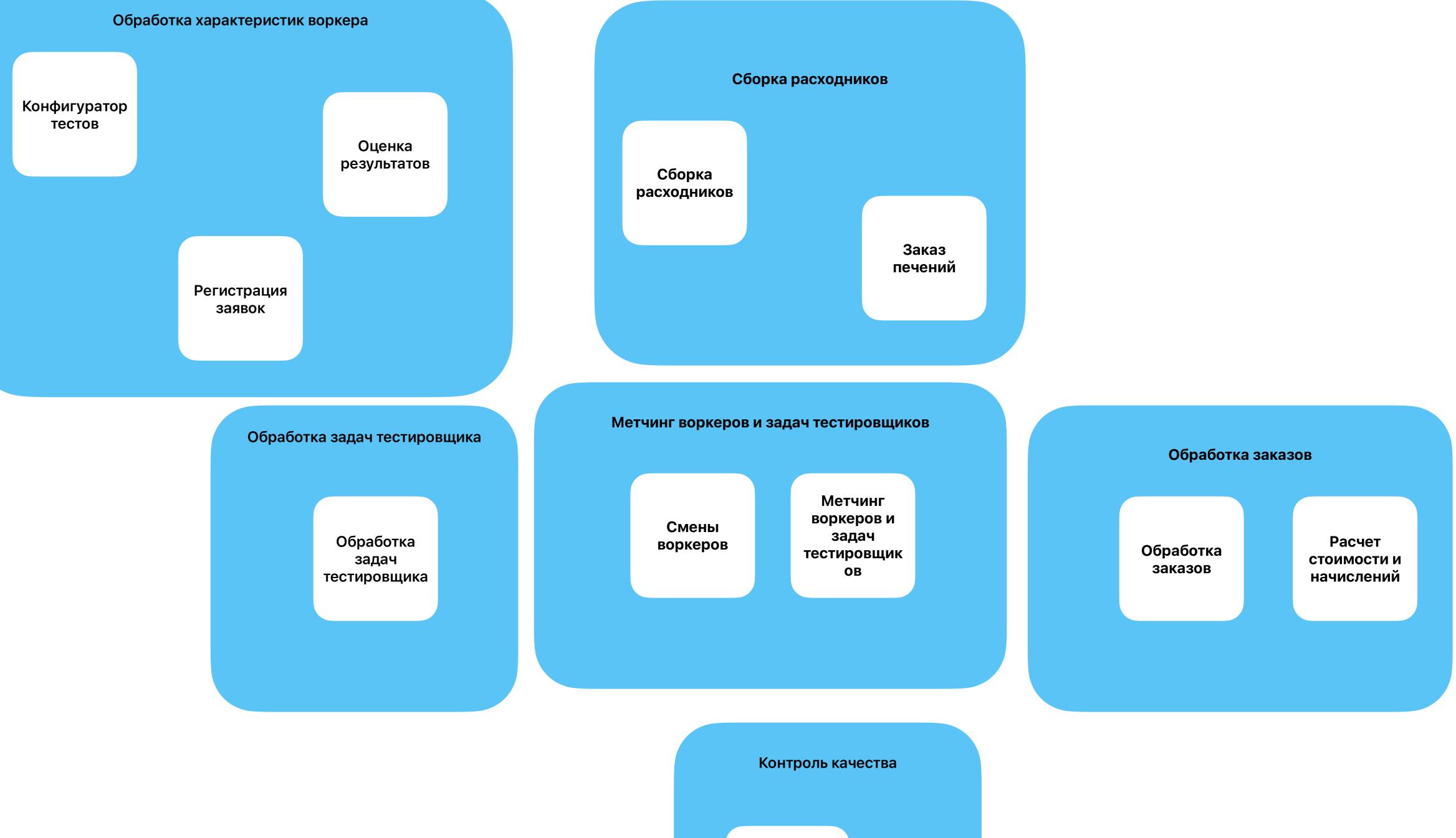
Модель для чтения (read model). Чтобы выполнить

команду, актору надо получить информацию о системе.

Например, чтобы сделать чекаут в интернет-магазине, надо знать, что лежит в корзине. Read model говорит данных, без которых актор не

может принять решение, выполнять команду или нет. Собирается либо из событий, либо из локальных данных.

может быть неживым -



Контроль