

OTP Sys

Трассировка и журналирование.

Включив флаги трассировки мы теперь можем создавать распечатки системных событий, включая все сообщения и смены состояния

При использовании `sys:log/2` для хранения событий трассировки в цикле сервера количество событий по умолчанию ограничено десятью. Это число можно переопределить, передав флаг `{true, Int}` при включении режима журналирования.

используйте `sys:log_to_file/2`, чтобы сообщения были направлены в текстовый файл.

Системные сообщения

`{in, Сообщение}` — появляется, когда сообщение приходит в `gen_server`
`{out, Сообщение, Кому, Состояние}` — Это сообщение генерируется при ответе клиенту с помощью специального кортежа `{reply, Ответ, НовоеСостояние}`, но это не происходит при использовании `gen_server:reply/2`. Сообщение — это ответ, отправленный клиенту, а Состояние —

идентификатор процесса клиента. Состояние является точно таким же, как и НовоеСостояние в кортеже `reply`.

`term()` — Разрешены системные сообщения в любом формате.

Вы можете реализовать собственную функцию трассировки, которая сработает в связи с системным событием. Вы можете сопоставлять аргумент с разными типами событий и принимать различные действия. Функции трассировки могут генерировать ваши собственные распечатки событий, включать трассировку низкого уровня с помощью `dbg` или встроенных функций трассировки, разрешать журналирование некоторой информации, выполнять диагностические функции или любой другой код, какой вам может понадобиться, а могут не делать и вовсе ничего.

Статистика, статус и состояния

Модуль `sys` также позволяет собирать общую статистику по поведению, а еще получать информацию об их внутреннем состоянии, включая данные цикла

Команды

```
sys:trace(Имя, ФлагТрассировки [, Тайм-аут]) -> ok
sys:log(Имя, ФлагЖурнала [, Тайм-аут]) -> ok | {ok, СписокСобытий}
sys:log_to_file(Имя, ФлагФайла [, Тайм-аут]) -> ok | {error, open_file}
sys:install(Имя, {Функция, СостояниеФункции} [, Тайм-аут]) -> ok
sys:remove(Имя, Функция [, Тайм-аут])
sys:statistics(Имя, Флаг [, Тайм-аут]) -> ok | {ok, Статистика}
sys:get_status(Имя [, Тайм-аут]) -> {status, Pid, {module, Модуль}, Статус}
sys:get_state(Имя [, Тайм-аут]) -> Состояние
sys:replace_state(Имя, ФункцияЗамены [, Тайм-аут]) -> Состояние
sys:suspend(Имя [, Тайм-аут]) -> ok (избегать)
sys:resume(Имя [, Тайм-аут]) -> ok
```