

Would setting time in a container update host time as well? If yes, then could it be prevented? If not, how is it limited?

По-умолчанию недоступно, но можно запустить контейнер привилегированно и можно будет менять время изнутри. *cap_sys_time* – работает с системным rtc и может изменять время. *seccomp profile* – дополнительная фильтрация, чтобы сказать, что такие-то вызовы запрещены, т.е. подключаем к приложению эту штуку, но с этой штукой теряем в производительности.

What are problems about running a few processes inside one container (if any)?

1. There's a good chance you'd have to scale APIs and front-ends differently than databases.
2. Separate containers let you version and update versions in isolation.
3. While you may use a container for the database locally, you may want to use a managed service for the database in production. You don't want to ship your database engine with your app then.
4. Running multiple processes will require a process manager (the container only starts one process), which adds complexity to container startup/shutdown.

Хочется также отметить, что при в случае использования большого количества ресурсов система может убить какой-то процесс (контейнер) и тогда несколько приложений, запущенных там будут завершены, что может быть проблемой. Кроме того, процессы так нарушают изоляцию, мешают потреблению ресурсов друг другу.

- I. https://docs.docker.com/get-started/07_multi_container/
- II. https://docs.docker.com/config/containers/multi-service_container/
- III. https://docs.docker.com/config/containers/resource_constraints/

What is better and why: running similar or unsimilar containers on the one physical node?

Думаю, что есть смысл запускать похожие, как минимум потому, что они могут делить образ, что экономит ресурсы на диске.

“So, how do you backup your container, you don’t. Your data doesn’t live in the container, it lives in a named volume that is shared between 1-N containers that you define. You backup the data volume, and forget about the container. Optimally your containers are completely stateless and immutable.” Но! Они могут создавать гонку за ресурсами, если используют доступ к одним ресурсам.

Кроме того, проще будет делать paging, если будет похожая read-only память.

- I. <https://www.docker.com/blog/containers-are-not-vms/>