

# Горизонтальное масштабирование и вертикальное масштабирование веб приложения

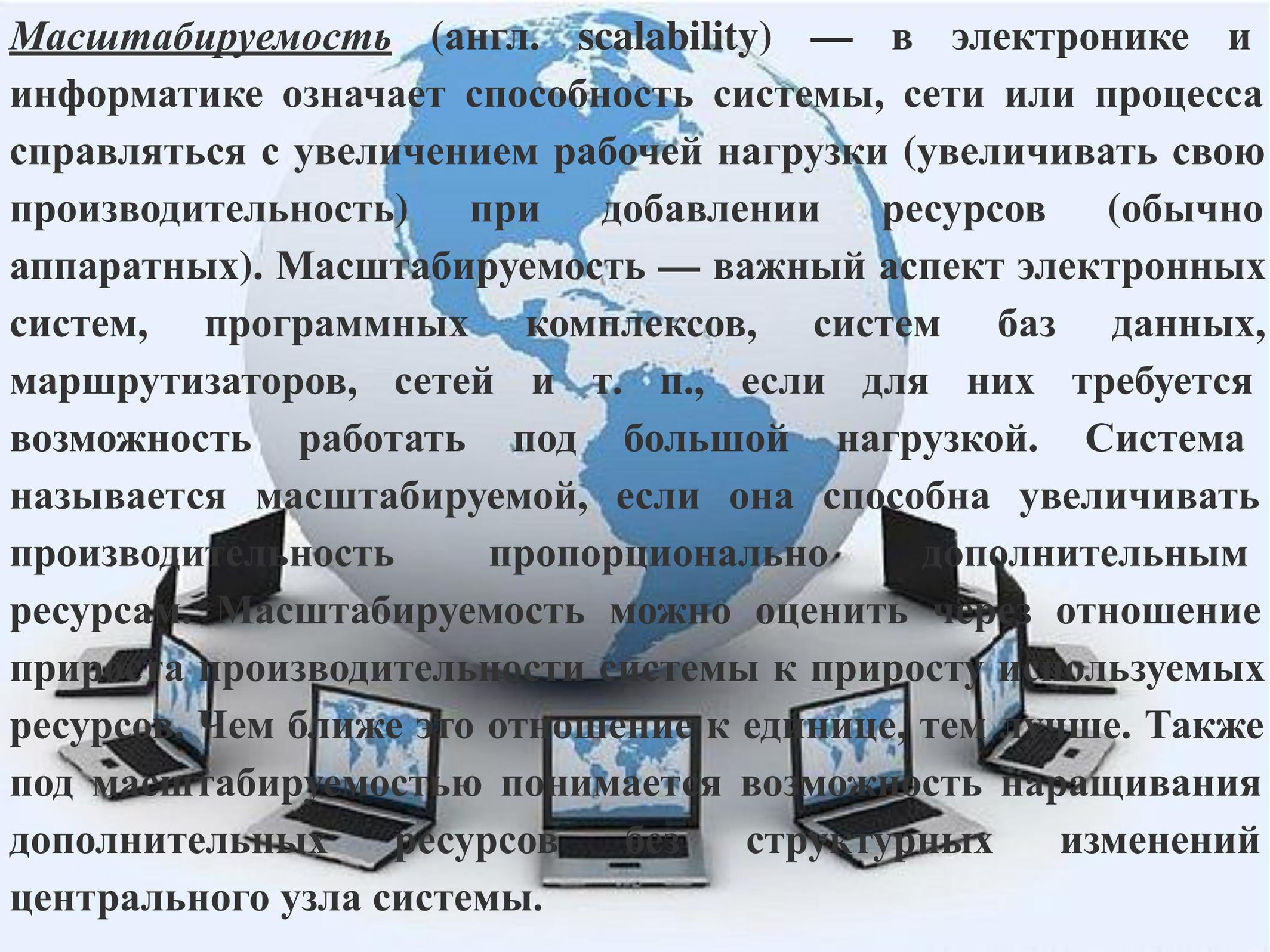


# Веб-приложение

клиент-серверное приложение, в котором клиентом выступает браузер, а сервером — веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети. Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются кроссплатформенными сервисами. Веб-приложения стали широко популярными в конце 1990-х — начале 2000-х годов.



**Масштабируемость** (англ. scalability) — в электронике и информатике означает способность системы, сети или процесса справляться с увеличением рабочей нагрузки (увеличивать свою производительность) при добавлении ресурсов (обычно аппаратных). Масштабируемость — важный аспект электронных систем, программных комплексов, систем баз данных, маршрутизаторов, сетей и т. п., если для них требуется возможность работать под большой нагрузкой. Система называется масштабируемой, если она способна увеличивать производительность пропорционально дополнительным ресурсам. Масштабируемость можно оценить через отношение прироста производительности системы к приросту используемых ресурсов. Чем ближе это отношение к единице, тем лучше. Также под масштабируемостью понимается возможность наращивания дополнительных ресурсов без структурных изменений центрального узла системы.

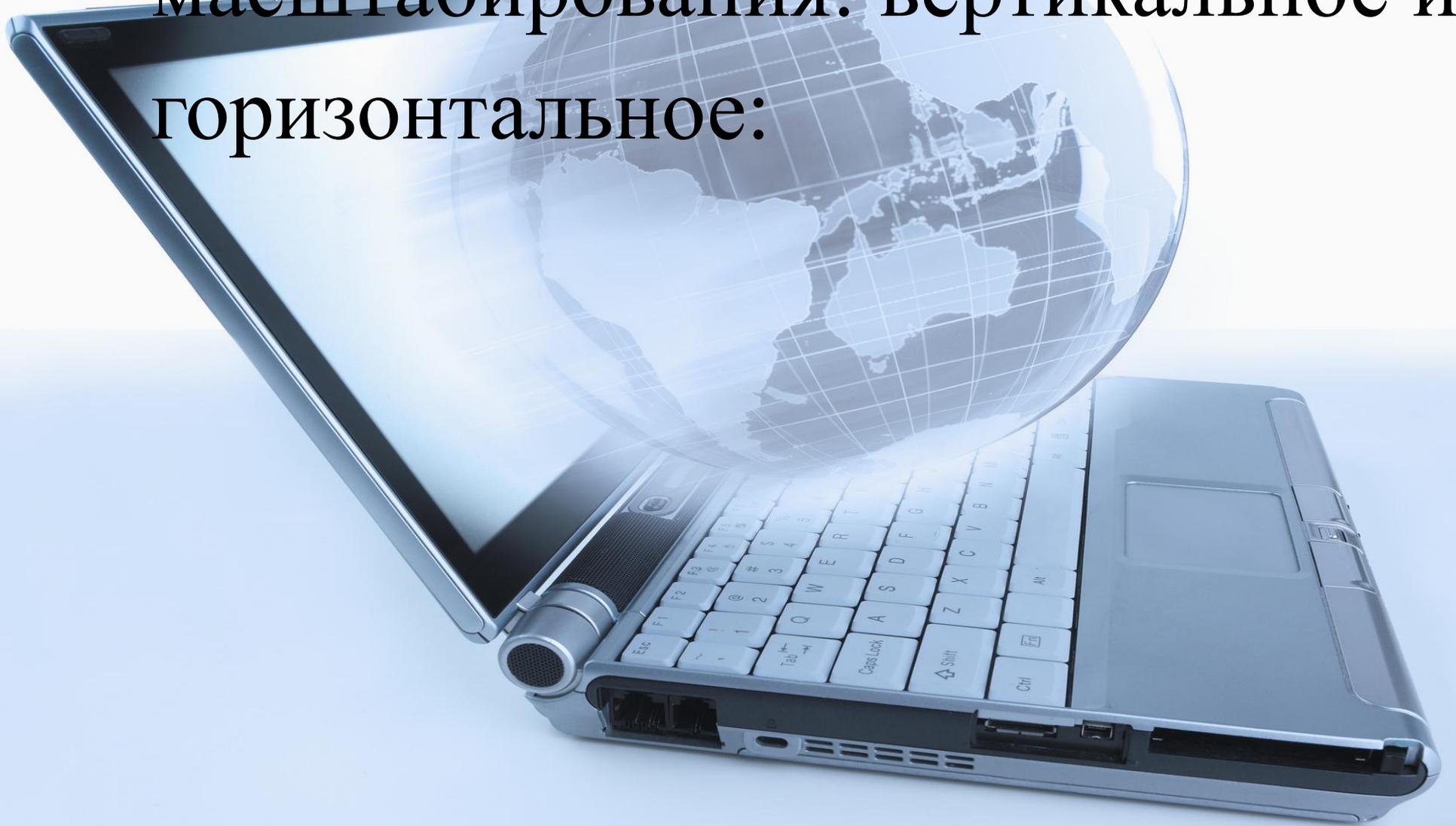


# Специфика масштабирования веб-приложений

Как и многие другие методы оптимизации программ, вертикальное масштабирование опирается на специфику предметной области приложения. В случае веб-приложения можно выделить некоторые существенные для масштабирования особенности:

- приложение ориентировано на выполнение большого числа одновременных запросов (как правило, больше количества вычислительных ядер/процессоров в системе);
- запросы могут исполняться за существенно различное время. В то же время задержка исполнения других запросов из-за одного, время исполнения которого велико, недопустима;
- выполнение запроса может быть приостановлено в связи с ограниченной пропускной способностью сети и/или задержками при чтении/записи данных. Эти ситуации должны обрабатываться во избежание простоя системы;
- возникает необходимость в долгоживущих процессах, например, для поддержания постоянного соединения или передачи значительного объема данных. Обилие подобных процессов при непра-

Различают два способа  
масштабирования: вертикальное и  
горизонтальное:



## *Вертикальное масштабирование*

Данное направление предусматривает увеличение вычислительной мощности среды запуска приложения. В связи с высокой стоимостью высокопроизводительных серверов, техническими и временными ограничениями данный способ имеет ограниченную применимость. Тем не менее он может особенно успешно применяться в облачных средах, где вычислительную мощь среды можно изменить с минимальными временными и финансовыми затратами.

## *Горизонтальное масштабирование*

Этот способ заключается в увеличении числа запущенных экземпляров приложения и распределении нагрузки между ними. Вместе с тем, архитектура приложения значительно усложняется, требуются дополнительные затраты на поддержание работы системы, а также необходимо внедрение дополнительных компонентов приложения, таких как служба балансировки нагрузки. Тем не менее, данный способ является основным методом повышения отказоустойчивости.

- **Вертикальное** масштабирование подразумевает увеличение производительности приложения при добавлении ресурсов (процессора, памяти, диска) в рамках одного узла (хоста). **Горизонтальное** масштабирование характерно для распределённых приложений и подразумевает рост производительности приложения при добавлении ещё одного узла (хоста).
- Понятно, что самым простым способом будет простое обновление железа (процессора, памяти, диска) – то есть вертикальное масштабирование. Кроме того, этот подход не требует никаких доработок приложения. Однако, вертикальное масштабирование очень быстро достигает своего предела, после чего разработчику и администратору ничего не остаётся кроме как перейти к горизонтальному масштабированию приложения.



# Средства языка программирования Go для разработки масштабируемых веб-приложений

Хорошим примером языковых технологий, ориентированных на создание масштабируемых приложений является язык Go, разработанный в 2009г. Ключевой особенностью данного языка является реализация идей языка CSP, предложенного Т. Хоаром в 1978 г., в частности идеи процессов, общающихся между собой при помощи каналов.

В качестве масштабируемого веб-приложения была создана платформа для проведения опросов населения «KtoZa».

Потребность в данного рода приложениях продиктована необходимостью проведения опросов в кратчайшие сроки с привлечением наибольшего числа респондентов. Данное приложение позволяет создать и провести опрос населения с меньшими финансовыми и временными затратами. К данному приложению предъявляются следующие

### ***Требования:***

- возможность управления опросом и статистикой в графическом интерактивном режиме;
- учет числа участников опроса и числа ответов;
- отображение статистики в реальном времени для создателя опроса (наиболее приближенном к реальному);
- поддержка вопросов с одним или несколькими вариантами ответа;
- возможность запуска и остановки опроса по расписанию;