Azure Concept

Janne Kemppi

2015

Server – Client Architecture

* **Server – Client architecture planning has fundamental problem of matching resources to needs.**
* There is vicious circle where increasing number of clients leads to increase in load (traffic) and subsequently resources to handle it.
  * Not increasing resources leads to time-outs…

Server – Client Architecture

* **In practice all server – client architectures are measured with three attributes:**
  * **Reliability** – Does function or request produce right response?
  * **Availability** – Is system up and ready to answer requests? Do you get timeouts instead?
  * **Maintainability** – How easy/difficult it is to restore system if it breaks down?

Server – Client Architecture

* **Architecture must be scaled up and/or scaled out when necessary:**
  * **Scale up** is used when server does not have enough memory or CPU power to complete tasks in time. It is caused by heavy processing tasks: very many objects or heavy algorithms.
    * Solution: Give more processing power to each request.
  * **Scale out** is used when server's external components such as network communication and disk access are the bottleneck.
    * Solution: Run each request in parallel. Use load balancer.

Redundancy

* **Redundancy means parts of the system stays available when it has problems.**
* There are multiple methods to increase redundancy:
  * **Failover** - Have a copy of system (mirror) that takes over if primary system has problems.
  * **Clustering** - Have multiples of hardware to take over if one fails.
  * **Geo-redundancy** - Have multiples of data centers to take over if one fails.
  * **Virtualization** – Hypervisor creates multiples of virtual machine file to deal with spike in requests within your organization.

Azure

* **Microsoft Azure's main selling point is ability to carry scale up, scale out and also scale down as necessary.**
* Azure can be configured practically instantly through Azure Portal to match load.
  * Scaling works also down which isn't easy to do when one owns hardware. Pay only what you use.
  * Autoscaling can be set to increase/decrease resources without human interaction with rules.
  * Existing assets can be virtualized.
* Hybrid solutions are possible (but difficult?).

Azure

* **Think following trade-offs:**
  * Business driven use?
  * Future needs?
  * Total control at hardware level?
  * Dedicated data center?
  * Hybrid use?
  * Security?
  * Law?
  * Large and small companies?
    * 50% of Fortune 500 companies use Azure.

Questions?