Fault Tolerant Architectures

Techniques Ensuring System Operation During Periods of High Loading and/or Component Failure

GSAW 2010
03 March 2010

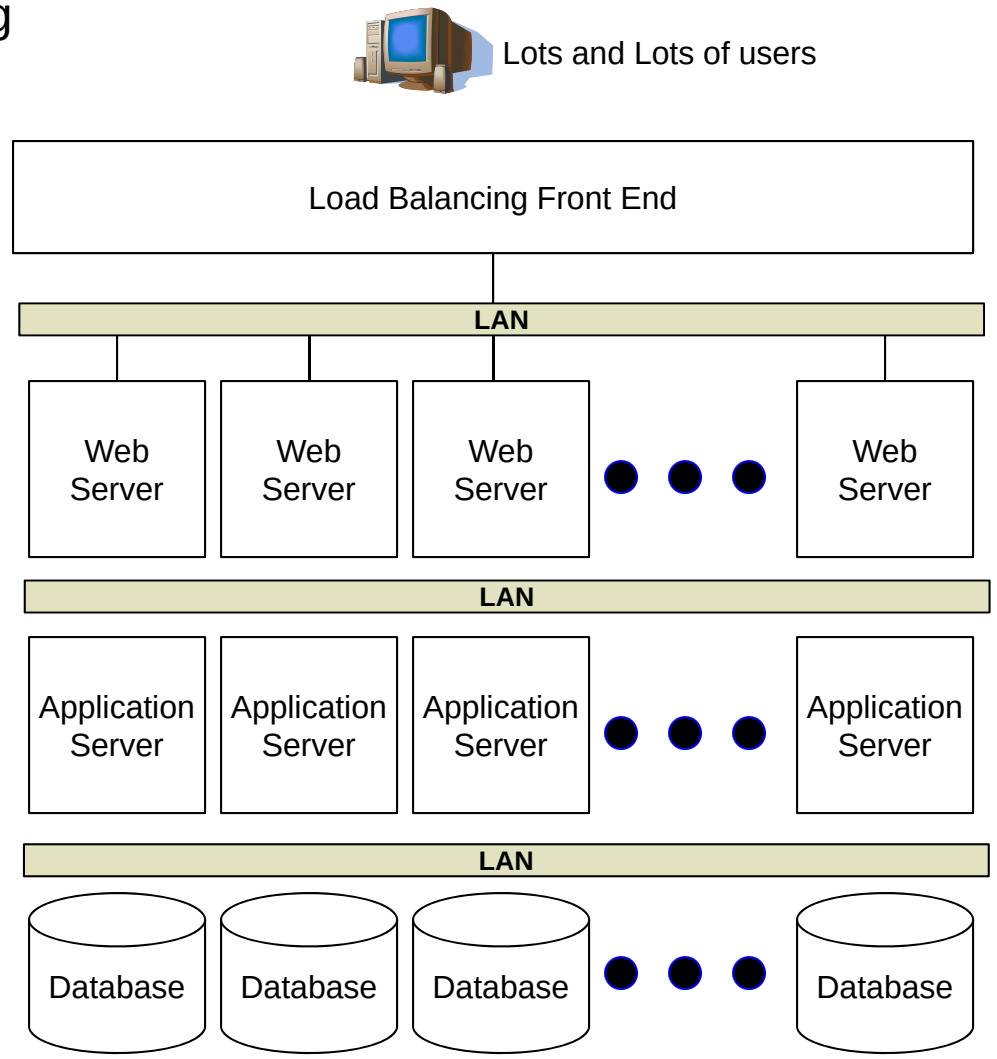
Some Definitions from Wikipedia (where else?)

- Fault Tolerant
  - In engineering, **fault-tolerant design**, also known as **fail-safe design**, is a design that enables a system to continue operation, possibly at a reduced level (also known as graceful degradation), rather than failing completely, when some part of the system fails. The term is most commonly used to describe computer-based systems designed to continue more or less fully operational with, perhaps, a reduction in throughput or an increase in response time in the event of some partial failure. That is, the system as a whole is not stopped due to problems either in the hardware or the software.

- Availability
  - The degree to which a system, subsystem, or equipment is operable and in a committable state at the start of a mission, when the mission is called for at an unknown, *i.e.,* a random, time. Simply put, availability is the proportion of time a system is in a functioning condition.
  - Typical availability objectives are specified either in decimal fractions, such as 0.9998, or sometimes in a logarithmic unit called nines, which corresponds roughly to a number of nines following the decimal point, such as "five nines" for 0.99999 availability.

Classic High Availability n-Tiered Web Architecture

- High Availability achieved through clustering of hardware
  - Even higher availability achieved through backup, fail-over sites
- Money CAN buy you High Availability
  - Buy LOTS and LOTS of Iron
  - How many 9's you get is a function of how much money you got!
  - Design the system such that even as components fail, users are unaware
- However, this only makes sense for high budget projects
- What approach does one take when you can't throw truckloads of cash at the problem?

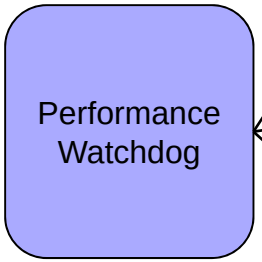


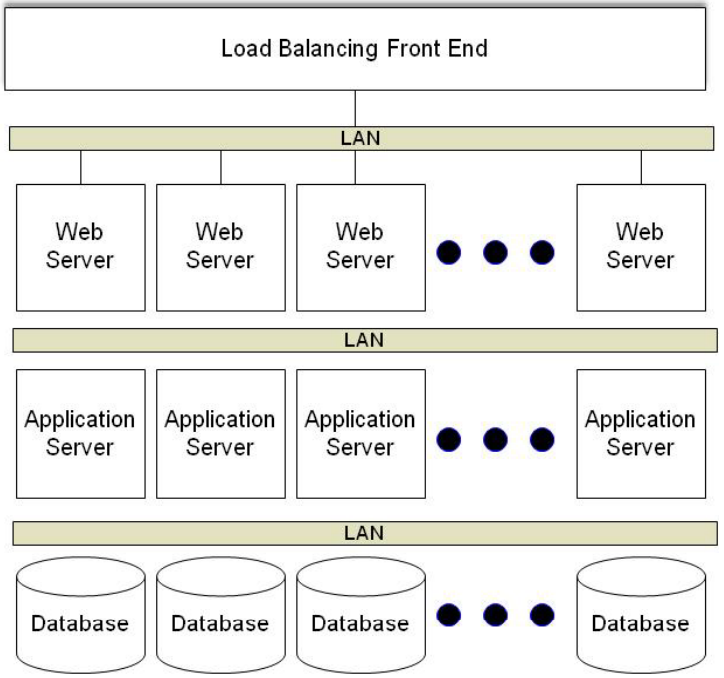
Degraded Operation is the Answer

- Projects on a budget need to accept the fact that as components of the system fail, the performance of the system will degrade
  - Response time is going to go down
- Question becomes “How bad is bad?”
  - What performance hit is bad enough that we need to take some action
- “Actionable” degraded performance is a function of what service is being rendered
  - Off-nominal delay of 5 seconds could be acceptable for a web page to an end-user.
  - Same delay could be completely unacceptable for a web service response
- Use case study of the application needs to be undertaken to determine acceptable degradation
  - Only you can know when the “LAME” bit needs to be set true
- So, how do you determine system is limping along?
- What do you do when your system is in the “Degraded State”?

Performance Watchdog Implementation

- Watchdog monitors critical performance parameters as determined by developers
  - Web page Response time on front end
  - Web Service delay
  - Critical App Server Calls
  - Database Queries
  - Etc……..
- Watchdog logic determines when “bad” threshold has been crossed
- Signals application to enter “Degraded Mode” (DM)
- Once in DM, watchdog monitors to determine if things improve or not





Degraded Mode Processing

- Degraded Mode allows developers to implement as little or as much logic as required/desired
  - Simple response with a different page saying “Heavy Loading be patient”
  - Perhaps some more robust logic providing an expected delay time
  - Web services provide delay response
    - Defined in WSDL
  - Priority Scheme which throttles system
- Admin Console for System
  - System alerts provided to Sys Admin Team
  - JMX is an example in an Java environment
- Watchdog continues to monitor and can bring system out of Degraded Mode
- Expectation Management
  - Users are informed that performance is subpar and kept “In the loop”
  - Users will be more tolerant of a temporary, known degraded state

Failure is not an Option…..

"When bad things happened, we just calmly laid out all the options, and *failure was not one of them*. We never panicked, and we never gave up on finding a solution."

-- Jerry C. Bostick
Flight Dynamics Officer (FDO) Apollo 13

"Success is not final, failure is not fatal: it is the courage to continue that counts."

-- Winston Churchill