0:09

okay today

0:15

i am going to uh today i am going to talk on uh

0:22

high performance computing issues in industry

0:28

the topics among topics are scalability load balancing and caching

0:37

i will be talking the general principles and give examples from

0:44

web services and servers

0:49

that are working for companies so it's more application oriented

0:59

so uh the turkish version is uh

1:06

[Music] so unity is actually strength so if you get together

1:14

you can accomplish more things therefore [Music] in the computer

1:20

technology if we can bring different machines together if we can combine the power of

1:26

multiple machines then we can achieve much more [Music]

1:32

interesting and [Music] bigger solutions

1:37

we can implement much bigger solutions

1:43

generally

1:54

there are vertical and horizontal approaches

1:59

uh to represent multi-tire architectures

2:05

when we say multi-tier architecture

2:11

in the 151 and 101 we have seen client server

2:20

architecture in the client server architecture you have the client and you have the server and client and server talks through the

2:26

internet uh everybody is happy but as things become more complicated

2:33

and that's security and performance issues are more complicated

2:38

people introduced multi-tire architecture in the multi-tire architecture there is a front end

2:45

that talks to the client this is client and web servers

2:50

but web server is not that powerful because it is dealing with too many

2:56

requests so some of the logic is distributed to other servers so web server

3:05

communicates with the application server application server actually implements the program logic

3:11

and therefore web server only displays

3:17

displays the result so it is for serving the browsers of the

3:25

clients so it's basically a user interface generation and communication to the

3:30

back back office this application server

3:36

runs on a standalone machine and also it may be connected to a database server

3:45

because database operations are taking too much time so when something is taking too much

3:50

time what you do is hire another person

3:56

too much work

4:02

hire another person

4:08

so multiplier architecture is similar if application server is connected to

4:15

database server in the beginning but it is not performing well

4:22

what they do is separate some concern so that application program application logic

4:29

is different differently handled by another machine database machine

4:35

will be different and database management database server

4:41

is also connected to database management system

4:46

which is another application running on the server

4:52

so in this case as requirements grow and become complicated

4:59

we will have multi-tier architecture this thing can be

5:08

can be represented vertically in some cases so if it is represented

5:13

vertically don't get confused this is from left to right

5:20

sometimes they are from top to bottom but generally

5:25

clients are clients are on upper side or left side

5:31

rather than bottom or right because there is a progress from top to bottom from left to right

5:38

in latin speaking countries therefore

5:44

as things get more deeper and deeper they are usually on the right side

5:50

so data flow is expected from left to right in

5:55

diagrams in general so why do i have multiple

6:03

machines one after another because

6:09

billing necessary as i said higher throughput so instead of one machine

6:15

you may have multiple two machines if one machine is performing performing

6:21

say 60 60

6:29

process per second one machine

6:34

two machines tell me

6:41

more than 16 60. it's more than 60. that's a very good answer it is not

6:48

it is not 120 let's say it is 100

6:57

so as you increase number of machines performance increases not

7:04

linearly but something between

7:10

linear and log relation

7:19

at some point it does not increase because of the complication

7:26

then you stop i mean you you change your architecture you should change your architecture because that kind of

7:33

multiple machine use will do more harm than

7:40

good after that number is achieved

7:45

so one thing is higher throughput second one is lower latency means

7:51

higher better response time

7:59

you go to google you go to hebster brother.com you go to facebook or whatever you do a

8:04

search and things come so there is a response time and that

8:11

response time is very important if you are a commercial website and

8:18

that is provided by using multiple machines to serve

8:23

millions of millions of people sometimes

8:29

[Music] machines get broken

8:35

hard drives crash power supplies go bad

8:41

sometimes memories go bad although less and less likely

8:46

but main boards cause problems

8:51

ethernet drivers ethernet cards can get broken down

8:58

in that case when you have multiple multiple machines serving one client

9:05

then if one of them gets broken the other one

9:10

can continue it's like having two customer representatives in a store

9:16

if you have only one customer representative and customer representative gets sick then

9:23

[Music] your you should close the store

9:28

but however if you employ two people if one of them gets sick other one can

9:33

others can continue that's for availability

9:42

there are different approaches you can hire two people

9:48

but only one person can work at a time or

9:54

they can share the customers they can work together

10:00

if one of them gets sick otherwise tasks will be

10:05

more difficult but normally they work together

10:10

some other cases second one

10:15

does not work at all he or she only works

10:21

if the first one gets sick why do we have that

10:28

because [Music] task may be too important it may require one person

10:42

and one person is sufficient

10:47

that also happened that also happened in the that happens in the

10:53

banking for example um

11:01

people the word scaling

11:10

scaling means expanding increasing like from

11:15

1 to 60 previous example 2 to 100 3 to

11:27

150 4 to 190

11:39

so this means i'm scaling up

11:45

the system it could be network that could be processing power that

11:51

could be memory hard drive um anything can be scaled depending on

11:59

the requirement examples

12:06

one traditional technique is load balancing meaning that

12:12

there are customers coming

12:20

typical client server architecture this is the shop

12:26

and there are two employees in the shop

12:31

these customers are coming so you have a and

12:36

b assume that these are servers so these customers are coming

12:43

load balancing means you will introduce a mechanism to

12:48

distribute one person here another person there one

12:53

person here another person there so half of the

12:59

or certain portion of the customers are handled by employee b

13:04

other portion is handled by employee a this is how

13:10

load balancing is done

13:15

if you may have you may have three

13:24

customer representatives and bunch of customers coming

13:29

so you have a mechanism called an engine

13:36

like number machine [Music] that is what we call load balancer

13:47

that distributes

13:52

customers to different servers customers are now clients

13:59

web clients and this load balancer could be a machine or

14:04

could be another person sometimes in reality if we are not talking about

14:10

computer operations for example it could be a dispatcher for the taxis

14:16

it could be a person in the hospitals or in any place so

14:23

that acts as a load balancer in any kind of

14:28

application you will need the load balancing code or application or device

14:36

that distribute will distribute the load as if

14:43

as if the customer only sees

14:48

one single point of service without without feeling the difference

14:56

that's what load balancer does second performance improvement in

15:01

industry is named as web caching

15:08

you are watching a film

15:15

in turkey and you're downloading

15:21

that film from youtube from youtube

15:28

you download where is it this is in

15:33

this is in california you're downloading it

15:42

and other people are also downloading

15:52

so by introducing a cache server here

15:58

once it is transferred to here it goes through an intermediary network

16:05

this intermediary network stores the media file or text file or video file

16:12

it could be text media or video or sound

16:17

any kind of file it stores the file temporarily so if another request

16:24

from ankara comes same cache server responds

16:30

locally without this is atlantic

16:36

without giving stress on without creating additional traffic on atlantic

16:42

fiber optic line so reducing the traffic on the network plus reducing the traffic

16:51

on the youtube server

16:56

we call them content delivery networks

17:05

today um [Music]

17:12

but you are in istanbul turkey this guy is in alt ankara

17:19

you watch the video and the guy in ankara also clicked on

17:24

the same link how do we know that the video is not changed

17:33

in between because when the second request comes

17:40

this cache server this this cache server asks

17:48

the main server about

17:53

whether there is a change in the file

17:59

by asking the last modification date if there is no change

18:05

from the previous request this will return no change

18:10

result sending no change is much easier than

18:16

sending the file so as soon as the cache server gets no

18:23

change message it

18:28

immediately sends the actual file

18:33

as if it is coming from california because california said there is no change

18:39

therefore it is crucial to manage those caches correctly

18:46

otherwise otherwise outdated

18:56

information in caches can be present so cash management is critical

19:02

correct cash management is critical and on the web

19:10

let me know if you have questions i have the chat window open in front of me as well

19:21

typical uh names you're all familiar with this but just

19:28

for reminding firewall is a device that restricts traffic

19:38

on rules and restricting traffic based on rules actually restricts unwanted traffic

19:46

only passes legitimate traffic legitimate and legal traffic

19:51

therefore it can protect

19:56

servers from outside attacks to a certain degree

20:02

but if you want to attack in a clever way

20:07

you don't directly jump onto a wall

20:13

how do you and how do you attack a building

20:19

you will probably look for open doors so instead of trying to break from the

20:24

wall what people do is what hackers do is

20:30

try to find open doors on the other hand having no wall

20:37

is much worse so it is better to have a wall uh then

20:43

also monitor the doors for the security

20:50

when we say the router uh router is a device that directs traffic to destination based on the best path best

20:58

planned path uh so that it can communicate between subnets

21:03

so router we have said that

21:13

i assume that there is a message here

21:18

okay this message is destiny this is a b c d

21:24

the message m message m goes there and destination is d message goes to d

21:30

[Music] it can go from here to here or

21:36

it can go from here to here or it can go from here to here to here

21:42

or here to here to here so there are four different ways for example i have shown

21:48

um at each stage if there is a router here

21:53

this router has a routing table showing this these two

21:59

[Music] connections and based on the address target address

22:05

looking at d the packet either routed this way or that way

22:11

depending on [Music] current conditions if this line is too busy

22:18

then this will be the first try sometimes routing table we call it

22:24

routing table sometimes routing table is fixed so that it always sends

22:31

the message b unless this thing is down

22:39

so at the end using the using the routers

22:44

your message comes to the destination node

22:51

that is what that is what router and it it is based it is

22:58

used to find destination on the best path sewage

23:05

connects multiple servers in the same subnet what is subnet

23:11

when you have

23:20

when you have this kind of ip number this part is

23:26

dedicated to the network this part is the subnet

23:33

class c subnet so [Music] when you have multiple machines on the

23:39

same subnet the connection between [Music]

23:45

the connection between these servers are done through switch also

23:53

communication between local machines in your house if you are not using wireless is also

24:00

done through switch if you are using wireless it's not switch

24:07

switch means

24:23

something like this so if a is going to connect to d

24:29

under the same subnet switch automatically connects it and passes this message

24:37

for a moment so it stores the information messages so that

24:43

the messages to d so basically it

24:49

makes a direct connection temporarily so it must be fast

24:56

with respect to each device

25:09

so load balancer as the customer employee example it

25:16

takes incoming requests for one virtual server and redirects them to multiple

25:22

real servers so you have one virtual customer representative but actually

25:29

10 real customer representatives so

25:34

load balancer distributes incoming loads in a proper way to different real

25:41

servers as if nothing is happening as if there is only one server

25:54

and consumer retail business there is an example this is typically

26:01

what we see in many companies we have the internet connection we have

26:07

the router and it could be fiber optic or it could be adsl

26:13

so before before the router we have the sewage

26:18

in some applications for example in our homes

26:23

these are single device at home but in company environment

26:30

usually they are different devices and

26:38

the clients

26:44

clients connect to internet and reach through the ip number to the destination

26:50

and this destination is actually pointing to web server so

26:56

this router

27:18

dot com so it's a website

27:25

so let's say port 80

27:30

this port 80 from router this one has

27:49

192 168 110 and 120 this port 80 wweing zoloku.com let's say

28:00

uh from dns

28:15

let's say you bought this dns to this ip number this ip number is actually

28:21

uh directed from public dns servers public data servers

28:27

indicate that this one is the ip number this ip number is found by routers on the internet so that

28:35

it is in istanbul it is in some company etc so that server is achieved

28:42

then once you get there if the request is http [Music]

28:48

then that request is coming from port 80 port 80 is then redirected to here by

28:54

the router

29:00

this web server handles the requests and also there is database server how do they

29:06

talk they talk each other

29:11

but they talk each other through the sewage database server does not talk to the router not necessary

29:18

because it is not it has no external communication but internally this one

29:24

talks to web server so web server asks certain queries database queries to

29:30

database server and since that kind of curing tasks are

29:36

relatively significant and rough hard tasks so that kind of operation is done by

29:43

other machine so web server can work more

29:48

with more performance with faster response time

29:54

okay any questions

30:07

okay let me explain it

30:17

well i am

30:22

this is machine a this is machine b

30:29

this says

30:38

so i am going to send a message

30:47

how am i going to send it in the tcp

30:53

protocol there is a target number

31:04

usually it's a long

31:09

piece of information it has header

31:17

saying that this is a tcp packet blah blah blah here is the source

31:24

this is my address here is my address here is

31:29

destination address ip number and

31:35

here is the information and here is the end it is like

31:42

telegram message if they're on the same line

31:53

okay if they're on the same line usually the communication starts by putting

31:59

starting starting putting things on the wire

32:06

suddenly you start talking like a prayer like isaan

32:21

so that is the header and then there is a continuation suddenly this listens

32:32

depending on the network okay depending on the network there

32:39

might be other machines if you are in wireless

32:44

if target is not b for example

32:52

if target is 20 not 30 this one hears it

32:58

but no does not respond that

33:05

connection is done in the mac

33:11

layer medium access control and you see there is number called mach number

33:20

mach number is actually registered in switch

33:27

mach number and ip numbers are set in the table on the switch

33:33

so switches actually uses mac numbers

33:38

for faster communication for to able to be able to switch to

33:43

correct places so if you are not interested

33:49

you don't do anything but if you are interested you keep getting it what kind of program

34:00

must be necessary on the b side

34:09

think of azan again prayer if imam starts praying from the mosque

34:20

how do you recognize it

34:27

yeah you should listen so basically a person

34:36

always listens

34:41

and in the

34:49

if you are an old person you can only listen

34:56

the prayer you don't hear other things so the only thing you listen to is

35:03

the prayer from the mosque next to your building other inputs you don't care

35:11

but if you are relatively young normal age person you have a lot of a

35:19

lot of things happening in your life such as your phone rings

35:24

doll rings imam prays email comes right so different things

35:32

happen so you should respond to different things at the same time so what kind of program

35:42

is necessary your programs probably

35:48

have a while loop

35:56

so that if asean if door

36:02

if let me phone

36:10

if

36:24

so uh there will be there will be an uh

36:29

program checking all these

36:34

modern servers if you are listening as an

36:40

as you listen as on your phone rings

36:46

how do you respond

37:01

[Music] there is also a process

37:07

keeps checking other things so that is what we call ports

37:17

you have multiple ports in your brain so that if one port is busy listening to

37:24

something other ports can be used it is basically multiple ifs with

37:30

multiple threads uh depending on the how your program is structured you can do multiple things at

37:36

once it is not exactly simultaneous but it looks like

37:41

simultaneous it acts like simultaneous therefore

37:47

how many ports a typical computer tcp standard you can

37:52

have any any number of ports depending on the depending on the capability of the architecture

37:58

but tcpip standard has 2 to the 16

38:04

different ports so from 0 to 65

38:09

535 ports so as you talk from [Music]

38:16

point a to point b you are sending the message destination source you also indicate

38:24

destination ip source ip and port number

38:30

so if when your message reaches here this b first checks

38:36

port number port number 10 port number is 10 for ezan so

38:42

it goes to religious procedure if it is port 20

38:48

it goes to door opening procedures so different ports are used for different

38:55

purposes even if you have the same ip number okay

39:02

and let me show you the tcp header

39:21

it's very big

40:02

okay this is a nice image

40:10

this is what you say so receiver mac address under mac address number of bytes data

40:16

and some interesting protocol variables here

40:21

then sender ip address receiver ip address this is ip packet this is ethernet

40:28

packet this is tcp packet sender port number receivable port number

40:33

etc each of them are inside each other

40:40

tcp packet is in ipacket

40:45

iep packet is in ethernet packet

40:52

okay they are inside each other the at the

40:58

outside ethernet packet is there so on the local

41:04

sense mac address switching is done at the switch

41:09

in the broader sense ip address determines the routing

41:16

operations and within the machine within the application layer

41:22

whether it is azan or door port number determines it

41:28

so that applications can talk to each other correctly

41:36

um

41:57

okay

42:07

that's it if you want to do more

42:14

power power power so if you are a newspaper

42:20

then this is what is happening typically you have a lot of web server

42:25

database server database server

42:30

so these these web servers talk to the people on the internet and database servers talk to web servers

42:38

internally but who knows this is a b c d

42:46

this is the client client machine

42:52

this client machine talks to for example hey who decides it the router actually

42:58

decides it so here the router is actually

43:03

sometimes called concentrator sometimes called load balancer

43:14

okay sometimes they put

43:20

another box here load balancer

43:38

doing multi-tire approach

43:44

including firewall and router you have load balancer then you have switch

43:51

through virtual lan you have servers and you have additional [Music]

43:57

switch and additional database what what did we do

44:03

we did this

44:08

in the previous case this guy

44:14

is maybe talking to database server one and this is two

44:20

database server one so how are we going to program it

44:26

for example a b c d e f

44:32

so abc talks to this def

44:37

talks to second database and from time to time these databases are synchronized

44:43

so that they contain the same thing

44:48

that can happen but if this guy

44:54

talks to database server one too much with a very difficult problem

45:00

very difficult problem but these have not that difficult problems

45:10

a and c will have difficulty in obtaining response from database

45:17

server one in that case a and c should also talk to

45:24

database server too because one is too busy because this guy is very difficult

45:30

problem a hacker is actually using this web server through the internet for a

45:36

particular time in that case the relation between these and these should be also

45:44

transparent as if they are one so database server and web server

45:51

should see themselves as one ani

46:00

[Music] so we see the ministry of

46:05

education as a single entity although there are multiple people working there

46:10

uh database servers see web servers as single

46:17

web servers see database servers as single not with this configuration but the

46:23

following configuration allows it

46:28

okay that can be an idea

46:37

later more load balancers are

46:44

sometimes required so multiple switches multiple load balancers

46:50

may be required because load balancer may not be good enough also you may have

46:57

different ways of connection to the network you may have for example

47:02

if thinking that you are university

47:09

well not university let's say you are newspaper you are connected from super online

47:15

but if super online goes bad what happens so you also hire a telecom line from

47:23

granted telecom 9 from telecom so if this one is true telecom

47:31

this one is super online if one of them goes other one continues

47:36

to work if both of them are up they can work together

47:45

unfortunately our university has only one line if that line goes out the whole internet

47:51

in the university goes out so that is quite a bad implementation

47:58

depends on money how much money you have if you have critical systems like newspapers or tv stations or

48:06

government services you should have this kind of approach multiple lines because if this line is cut otherwise continues

48:14

to work so you may have multiple switches in first vlan and then load balancers dense

48:21

images different types of servers and different types of database servers so everything

48:27

is now transparent and more redundant and high performance

48:37

they also protect database servers with additional firewalls to protect data further more

48:45

if something happens to web servers it cannot go deeper

48:53

it's like having multiple doors in

48:58

buildings you can open certain doors but you cannot open other doors which have more

49:04

critical content

49:16

can have yes although not common you can have for example administrative a core database

49:23

which is only sometimes accessed but

49:31

what happens is [Music] when you have that kind of critical information

49:38

you should not connect it application in terms of application you should not connect it to

49:45

public servers these are public servers of the

49:51

company what you do is you develop

49:56

you develop additional villain additional we land and under additional

50:02

we land you have critical things uh only certain traffic

50:08

can pass from this vlan to the from this field and to other villain

50:13

and the result there will be also firewall so critical things will be here if it is

50:19

more critical then just don't connect it

50:28

connection means danger

50:38

so

50:44

single point of failure

50:51

here for example if this one is broken

50:57

your operation is gone

51:03

if this point is broken your operation is gone if this is broken

51:08

your operation is gone if this is gone your operation you are discriminated if this is gone you are

51:14

disconnected that means you have single point of failure with this

51:20

structure single point of failure is eliminated

51:27

therefore if one of them is broken the other part can

51:32

still function and keep giving the service

51:39

so that's what we call single point of failure

51:46

with additional

51:53

locations because this is in california if something happens in california

51:59

then your company can be disconnected therefore

52:05

banks and other big companies what they do is they replicate their servers

52:11

in different locations with different scales so that if something goes bad

52:16

other location can continue to work and today for example

52:22

many banks have data centers in izmir or in konya or in ankara in addition to istanbul so

52:30

if something happens in istanbul uh some kind of

52:35

invasion by russia or some earthquake or something like that

52:42

additional data centers

52:47

can still give the same service

52:56

other examples multiple with multiple firewalls with

53:04

switches directory servers search servers web

53:10

servers operations managers active directory servers authentication managers

53:17

then you have business test server these are

53:23

database servers and directory user servers etc so similar image

53:29

as previous but with different blend of it is taken from different place

53:40

basically you have database here connected with

53:45

a firewall or switch firewall or switch multiple application servers and web servers and

53:52

firewall and connected to internet

54:02

sometimes these servers have double

54:09

these servers have double ethernet ports that's a machine

54:14

it has one ethernet port goes up another ethernet port goes up down

54:19

upper part is like okay lower part is

54:34

something like that so totally different ip scheme so

54:39

the happenings in this ip domain upper part cannot go down so this database server

54:47

is protected also physically

55:17

multiple techniques to exist you can have multiple smaller servers

55:24

to get more powerful state or

55:30

you can add few larger servers add more processors more memory and disk

55:37

space for database servers that is the general

55:42

trend for web servers multiple smaller servers is more

55:48

easy to implement that's probably because implementation of database servers is much harder

55:55

than implementation of the web servers therefore what when we want to scale

56:02

database servers it's not that easy so what they do is they use powerful

56:08

database servers if necessary and they increase numbers

56:15

until there is no room for improving the

56:21

machine

56:30

the scalability can be implemented through network through servers

56:35

and also we should make sure that the network has capacity [Music]

56:41

when you should when you add servers because adding servers

56:48

adding these servers require and generate traffic

56:53

in the network so network will be crowded local network performance should be

57:00

good enough for such kind of scaling

57:08

why do we scale we scale because we want better performance better better better better and fast site that goes down

57:16

because of one component is a bad thing so site should be fast and also should be

57:21

reliable so when you design something it should be both reliable and

57:27

fast redundancy and speed

57:34

therefore there are some

57:40

outcomes lessons to be learned one is don't take network is always there

57:46

network is sometimes not present

57:55

it goes down it is congested people may attack so network is

58:00

sometimes not available

58:06

therefore if you are designing an enterprise system you should

58:12

buy multiple internet lines you should incorporate

58:18

multiple routers multiple switches running in parallel

58:23

configured in parallel running multiple web servers running multiple database servers

58:29

running multiple power supplies multiple air conditioning units even the air conditioner in the system

58:36

room must be more than one because if one of them goes down other one keeps working

58:54

the applications on scalability is one way is scaling up

59:00

the other one is scaling out scale up means replace old server with the new server

59:10

get a better machine getting a mesh better machine is called scaling up

59:15

scaling out means adding extra servers adding extra servers in interesting

59:23

among the approaches farming cloning redundant array of computer systems

59:29

partitioning reps and load balancing and web caching

59:36

i'm going to go over those farm

59:43

collection of servers applications and data at a particular size

59:48

site farms have made many special services like directory servers security servers

59:54

web servers mail database collection of servers

1:00:01

it's called farms

1:00:07

okay cloning

1:00:12

in the farms each of them

1:00:18

may be different clone

1:00:28

are replicated nodes containing same software and same data

1:00:35

if one of them is overloaded load balancing system can be used to allocate network

1:00:40

allocate the work among the duplicates so all of them are the same they are copies of each other

1:00:46

serving behind the load balancing system so but that's what we call

1:00:51

cloning farming in the farming assume that you are serving for example

1:00:58

videos some part of the videos are on one side some part of the videos are on other side other side so they are

1:01:05

distributed how they are used a concentrating web service

1:01:12

distribute uh the requests based on where the [Music]

1:01:19

where the target location is that is sometimes done by rewriting the http request

1:01:27

for example media files are on one server text files are another server although the website is the same

1:01:34

that can be done [Music] that can be done through apache

1:01:41

plugins

1:01:47

this is an example for clones the these clones share nothing

1:01:55

they are just identical ones uh if you have network good enough

1:02:01

your performance increases sometimes you have computation

1:02:07

computation requirement but not that much disk requirement disk performance is good enough but

1:02:13

computation is different difficult therefore in that case disks may be shared by multiple servers

1:02:20

multiple servers uses the same disk through the disk server through another network

1:02:26

that can be also implemented and this is then later called

1:02:33

storage area network and this will be storage area network controller

1:02:48

if it is not network

1:02:53

it's also network attached storage that's also

1:03:00

name the no differences not much

1:03:05

reliable area of clone services a collection of clones for a particular service it could be database could be

1:03:11

service web service etc it is either shared nothing or shared disks

1:03:21

i have explained this this storage server

1:03:27

oops storage server should be fault tolerance so this also must be

1:03:34

strong enough so that it won't be a single point of failure

1:03:40

or service continuity

1:03:46

clones and redundant area of clone systems

1:03:53

they can be used for usually web applications with low consistency

1:03:59

requirements so if you are only reading things and serving to public

1:04:05

file servers web servers cloning and clone servers are good

1:04:12

replication when you attach a new clone automatic replication is done

1:04:18

automatic request routing to load balanced work is done by the load balancer

1:04:24

if there is a failure in one of the servers automatically it should be excluded from the network

1:04:30

and it should be also configured so that anything goes bad should be

1:04:36

recognized automatically and repaired automatically by the by the operating system

1:04:46

sometimes [Music] objects are

1:04:53

for example this one contain male this monk image

1:04:58

this one video this one this one text

1:05:06

so in that way

1:05:12

there is some scaling but it is through partition

1:05:17

this partition and cloning can be combined together

1:05:22

then you will have more power along with distribution

1:05:38

partitioning means uh

1:05:43

putting things in different positions by duplicating the hardware and software

1:05:49

dividing data among nodes so not same data is stored everywhere

1:05:54

different nodes store different data when you partition but from the upper layers from the customer

1:06:02

point of view you should know you should not realize it you should be

1:06:08

thinking that it's a single application single single server it looks like single server but actually images are

1:06:14

stored in this drive videos are stored in another drive

1:06:19

from the application you shouldn't feel it

1:06:25

in terms of availability and downtime

1:06:31

it does not improve it actually is worse because if one of them goes bad

1:06:37

system goes bad because data is stored in one place but

1:06:42

it can improve the speed it can improve the performance in order to increase the availability

1:06:51

you do this way in addition to partition you also

1:06:56

duplicate it so duplication and partition them together improves

1:07:02

irritability and speed [Music]

1:07:07

so when we do taxonomy when we

1:07:14

when we focus on these

1:07:19

in categorically farming is the top concept

1:07:26

partitioning is one side cloning is on the other side cloning can be done through share

1:07:32

nothing or share through shared disk share disk is something like partitioning

1:07:38

partitioning can be done through packing packing can be done through shared nothing

1:07:44

it could be active active both of them are working together or active passive passive one works if the other one fails

1:07:52

that's also possible

1:08:00

reliable array of partition services is actually both

1:08:06

use partitioning and cloning together when we have update intensive large

1:08:12

database applications we use a reliable array of partition

1:08:18

services because it is bought it has bought high

1:08:24

performance and high availability

1:08:29

and reliability characteristics

1:08:39

uh yeah

1:08:44

for example and [Music] when we talk about mainframe

1:08:52

[Music] these kind of numbers are [Music]

1:08:58

told so 0.99999 uptime that means

1:09:03

one in a 100 000

1:09:10

downtime so one day in 100 000 day is down or one second

1:09:19

for each 100 000 second is down so it is less than five minutes of outage per

1:09:25

year even if that is too much then you should

1:09:32

have two of them

1:09:39

okay load sharing

1:09:45

uh sometimes how do we how do we distribute

1:09:52

the load how do we

1:09:57

say load balancing

1:10:02

how do we load balance one idea is changing dns

1:10:10

having multi having smart switches or routers and load balancers together

1:10:24

it can be done through hierarchical architectures i am going to show or

1:10:30

sometimes distribution is done through where things are for example images are on

1:10:36

this server so if the content contains image it is taken from image server so that is

1:10:42

locality everywhere distribution i'm going to [Music]

1:10:48

first explain what dns rotation is you have the

1:10:53

[Music] uh

1:10:58

server cluster on the internet you have servers running up

1:11:05

and people are looking for www

1:11:11

let's say [Music]

1:11:17

fifa orc okay

1:11:22

fifa hawk has let's say four servers on the dns record in the dns servers

1:11:30

fifa when you say fifa org it should map to an ip number right it

1:11:35

is called as a record

1:11:40

archive a record means which ip number is

1:11:45

connected to the domain name

1:11:51

when you have dns rotation class dns rotation

1:11:59

clustering this dns server returns sometimes this number sometimes

1:12:07

this number sometimes this number for each request so

1:12:13

the world knows that fifawork is sometimes

1:12:18

this server sometimes this server sometimes the other server so depending on who you are

1:12:26

different clients browsing fifa work website some of them are using

1:12:32

this server others are using the second server others are using the third server so if you change this dns

1:12:40

lookups very frequently at each time you can do

1:12:45

very nice dns rotation

1:12:50

load balancing done is that nice

1:12:56

it's kind of easy

1:13:02

it's a simple clustering strategy client-side ip caching loading balance

1:13:08

failover management is not present but still

1:13:16

still okay

1:13:24

if it goes down how do you solve the problem you can have hot stand by machine

1:13:29

[Music] so you can have a second machine if the first machine goes down second machine

1:13:35

automatically goes up that is expensive because you have to pay twice

1:13:41

some banks do that cisco

1:13:48

or huawei or other networks have switching products

1:13:55

so load balancing switches are also present so even if you have

1:14:04

single ip number smart switch

1:14:12

distributes same request to four servers in randomly

1:14:21

it also checks whether these are connected or not

1:14:26

by so by checking certain uh requirements if the if they are not connected

1:14:36

it eliminates the connection so that it does not send packets anymore so monitoring and switching is done

1:14:42

together in the load balancing

1:14:51

switch base cluster is as follows so this switch changes the requirements dns

1:14:59

returns only one result and clients

1:15:07

always have the same ip number right but this switch distributes it

1:15:14

so it is more intelligent and this switch also can sense if the server goes down if the server

1:15:21

goes down system still can function

1:15:27

all of them [Music] all of the clients can get the service

1:15:35

and you have flat architecture with switch based as well as we have seen

1:15:43

we cluster the servers through one ip number distribute the workload through switch

1:15:49

typically around the level you can have failure detection uh the problem is

1:15:54

for the dynamic content changing content it's it may not be sufficient because

1:16:01

when the content is changed all these must be updated and also

1:16:06

if the content is dynamic for example if the client is changing this content

1:16:12

such as such as filling a form it is filling the form here

1:16:20

but not here so the data for a certain amount of time

1:16:26

will be different than data on the other server so data synchronization

1:16:32

must be implemented which is also a problem

1:16:38

so in order to separate that generally database is also separated

1:16:45

as we have seen while ago since database is separated

1:16:51

active data is stored in different database servers as if

1:16:57

there is only one database server so these are only working as a

1:17:03

front end for the customer actual data is stored transparently in single data point which is also

1:17:11

redundant

1:17:19

so various architectures examples multiple architectures this is

1:17:27

this is through routers multiple servers switching

1:17:33

generally when you have flat architectures or dynamic content adding and removing

1:17:38

nodes is difficult manual configuration may be required

1:17:44

uh two level architecture as we have seen at the beginning

1:17:50

is better level one is considered as master it is

1:17:56

called as it is including static and dynamic content level 2 is only dynamic content and i'm

1:18:03

going to show that this is hierarchical architecture

1:18:09

there are masters and there are slaves behind

1:18:15

so master part

1:18:20

static and dynamic content is level two only dynamic content

1:18:26

in this case static content is served immediately

1:18:32

but if there is dynamic if there is dynamic content like this is database this part is database

1:18:38

it is accessed as if there is one network and used

1:18:44

as a separate entity as a single entity some of them are dedicated some of them

1:18:51

are not dedicated for a particular

1:18:57

data

1:19:05

master takes all the requests cgi requests results load information

1:19:13

these are all distributed by master and slave

1:19:18

slave network response to the local area network so this is

1:19:24

this is level two this is level one and requests are coming from here

1:19:35

here we have database here we have web server

1:19:42

okay

1:19:50

it is good for failover support if the slave fails another one can start

1:19:58

and separation of dynamic and static content is good for data management

1:20:06

resource intensive jobs are run by slave processors so customer response times

1:20:12

are improved sometimes locality aware distribution is

1:20:20

preferred like where the file is you should direct that request to the where the file is

1:20:26

therefore file replication is reduced copying everything is everywhere is not

1:20:32

required anymore your load balancer knows where the things are so

1:20:38

requests are distributed based on locality

1:20:43

that improves hitting correct servers

1:20:49

secondary storage capacity is therefore increased

1:20:54

it may require specialized servers in terms of software

1:21:01

[Music] but [Music]

1:21:07

it is relatively cheaper to implement than copying everything onto servers

1:21:18

this is one nice figure for that you have the request coming these are

1:21:24

the requests coming front and not resolves what the where the requests are

1:21:29

coming from these triangles are usually on this server so it is taken from this server

1:21:36

the cmbs are already in the part b

1:21:43

of the network so they are taking from these nodes and combined and short back

1:21:49

so processor uses this and big companies and

1:21:56

government also uses this kind of approach such as salt bacon or other banks you

1:22:02

can use this pre-processing and redirection

1:22:07

operations sometimes this part is done through

1:22:33

now we have seen this finally caching on the web

1:22:40

uh web is increasingly uh popular and

1:22:46

expanding every year every day therefore in order to in order to

1:22:52

reduce the burden on the

1:22:58

in order to reduce burden on the server static information

1:23:04

unchanged information can be copied in content delivery networks

1:23:15

also local cache servers

1:23:21

both in server side

1:23:27

or client side

1:23:34

you have client and server

1:23:43

when you are talking to the server this is client

1:23:50

server does something right

1:23:55

if it is a difficult problem each request

1:24:00

causes server to run and some response comes back

1:24:12

if it is the same response and if i am going to make the same

1:24:17

request again maybe i should put something

1:24:24

another server called as cache server

1:24:30

to monitor what is going back and forth if this guy is asking same thing

1:24:37

and if there is no change let's not bother let's not bother the

1:24:42

server let's immediately return

1:24:48

the previous result because no change has happened

1:24:53

that is what the cash server in server side is

1:24:58

similar thing can happen in the client side it can be done by the isp

1:25:05

so this client side also checks the requests if the same request is coming then

1:25:10

also stores temporary information if the same information is asked without

1:25:17

requesting it from server actually it goes back to the client saying that i have it

1:25:24

already so it automatically serves

1:25:30

the content so that speeds up the networks significantly you have clients you have web servers

1:25:37

this is what general or traditional architecture and with the

1:25:43

with the servers with the web proxy there is an intermediate server between client and web server

1:25:52

usually it is connected to the firewall together

1:25:57

performance of the network therefore response time is increased unchanged things like images and text files and

1:26:04

pieces of pieces of code like javascript codes

1:26:11

that are static all the time can be cached through these servers so

1:26:16

actual servers will be free of doing these things again and again

1:26:23

it improves the network it also improves the server load with the

1:26:29

web proxies cache servers these are the proxies

1:26:38

the content is support this is actually client side content is stored

1:26:45

this is client side this is server side uh the leftist client site

1:26:52

right the server side [Music]

1:26:59

the content is stored and if there is no change in the content it is automatically served in a fast manner

1:27:06

[Music] caching is done through for the isp

1:27:11

this is these are done in the isp site

1:27:17

so clients are connected to the cache servers cache server checks whether there is a change if there is no change

1:27:24

it returns back the data

1:27:32

i have explained this so it increases the bandwidth performance

1:27:38

it reduces the latency reduces the workload of the web server

1:27:43

increases the robustness of the web service and also you can

1:27:49

do some analytics use some you you can do some checking about how

1:27:55

usage is performing disadvantages management is difficult

1:28:01

latency is sometimes cache server latency is sometimes significant

1:28:08

yeah that's basically

1:28:15

cache coherency and cache replacement cache coherence means whether the cache

1:28:21

content is the same as the server content that's a big problem

1:28:26

therefore in order to coherent in order to have a similar content same content not similar same

1:28:34

content on the cache and the server you need to continue to update the cache and if the

1:28:39

cache is too big then it also affects the server

1:28:45

so cache should be in the correct size determined by

1:28:52

experiments correct size so it is not too big and it's not so small

1:28:58

sometimes prefetching is done things that are going to be used are predicted

1:29:05

and then cached in advance it is called as prefetching

1:29:19

yeah there are mechanisms to replace and manage caches there are algorithms for replacing

1:29:25

caches and it's not free actually there should be algorithms such as list recently used

1:29:32

list frequently used first in first out etc

1:29:38

uh i'm going to skip these uh

1:29:45

cache coherence is basically whether the cache is the same or

1:29:51

different than to original data

1:29:58

there are strong cache coherencies and weak cache coherencies available stroke cache coherency requires

1:30:05

you are sure that it is validated every time that's what i have

1:30:12

asked that's what i have said at the beginning of the class it is validated a weak currency checks every day if it

1:30:19

is checked today then we say it is good enough so we check every day

1:30:28

that's week time to live week consistency

1:30:35

so it is not easy to manage caches because content is changing sometimes it is dynamic

1:30:43

for example it has your name your name and your for example your shopping basket

1:30:49

should not be cached because shopping basket is only belonging to some person so there is an in the url there is a

1:30:56

session id if the session id is different then cash is not shown etc so

1:31:02

there are quite interesting issues there

1:31:08

i think i think we are done

1:31:15

dynamic update propagation yes that's great any questions

1:31:20

if you have any questions i will try to answer i think

1:31:25

we are good for today

AllListenableRecently uploadedWatched