CSR: Quality of Configuration in Large Scale Data Centers (CNS 4107876)

PI: Krishna Kant, Temple University

(Ibrahim El-Shekeil, Ph.D. Student, Amitangshu Pal, Postdoctoral Fellow)

Introduction

Misconfiguration Problem
- The vendors often advertise five 9’s or better availability
- But, too many downtime episodes, to meet these goals.
- Substantial impacts in many areas:
  - unsatisfactory online customer experience, lost revenue,
  - lost customer goodwill,
  - high infrastructure & operational costs

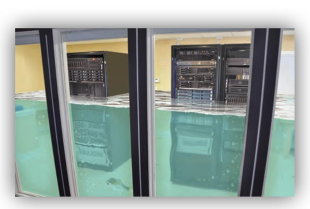
How bad are misconfigurations?
- Responsible for 62% of downtime and 65% of security exploits (2011 study).
- Expected to continue increasing due to
  - Extensive virtualization,
  - Architectural heterogeneity, and
  - increasing size and complexity

Why Downtimes?
- Misconfigurations:
  - inadequate or flawed operating procedures coupled with hardware and software misconfigurations and human mistakes,
  - ad-hoc procedures are used in the first place, or
  - ad-hoc fixes are implemented to fix problems.
- Failures:
  - hardware failure,
  - operating system or software failure,
  - intrusion, virus outbreak, or
  - natural disaster
- Planned outages for relocations, upgrades, etc.

The goal
- Systematic analysis of the operational issues and misconfigurations to minimize downtime or impact on performance.
- Study mechanisms to improve the data center availability and resilience.

Progressive Recovery in Data Centers

Introduction
- Disruptions in data centers:
  - Natural disaster → Japan earthquake, Hurricane Sandy
  - Storms or lightning took down Google’s St. Ghislain data center operations for five days
  - Technical hiccups (hardware, software failure, virus outbreak) affected the services of Bank of America and Amazon centers for 4-6 days
  - Relocation of data center or upgrade
- Problems in restoration
  - Requires multiple stages → sometimes take a few weeks to several months
  - In each stage, the partially recovered infrastructures are can provide limited services at some degraded service level

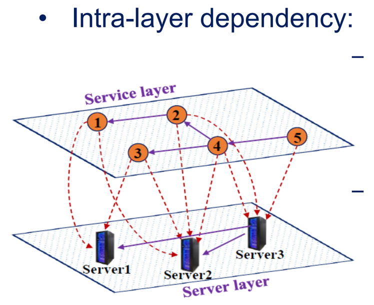


https://www.fiber.net/blog/data-center-disaster-and-emergency-preparedness-not-a-priority-for-some/

https://thecustomizewindows.com/2014/01/disaster-recovery-what-happens-in-the-data-center-after-a-natural-disaster/

Key Design challenges
- Devise a restoration plan to support partial business continuity, that allows applications to progressively come back online after failures or disruptions.
- Speed of restoration limited by several factors
- A large disaster brings down multiple data center services
  - Services in a data center are often interdependent
  - Precise recovery sequence of services crucial to restore the most critical applications back first
- The availability of human resources with the desired expertise is limited

2-layer Interdependency Framework
- Service Layer: User level and low level helper services
  - Human resource system, Active directory authentication, Microsoft DNS etc.
- Server Layer: Servers that need to be restored
  - Web front end server, application server, database server etc.
- Inter-layer dependency:
  - Human resources → Application server, database server
- Intra-layer dependency:

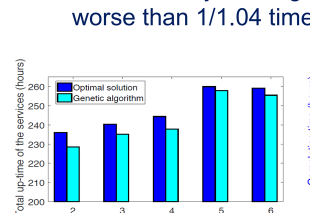


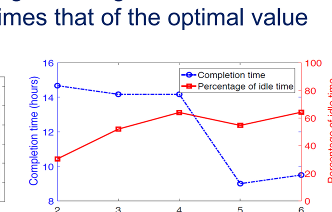
  - Email services, human resources and SharePoint → depend on the DNS and Active Directory services
  - Web client → depends on the front and back end servers for email services

Proposed Heuristics
- The problem is NP-hard → Solved using a genetic algorithm based meta-heuristics
- The server restoration sequence is embedded in the chromosome structure → a gene represents a server to be restored
  - S1—S2—S3—S4—S5
- Different than traditional genetic algorithm → because of the inter-dependencies
  - Precedence constraint
  - S5—S4—S3—S2—S1 is a wrong chromosome structure

	S_1	S_2	S_3	S_4	S_5
S_1	0	0	0	0	0
S_2	0	0	0	0	0
S_3	1	0	0	0	0
S_4	1	1	0	0	0
S_5	0	0	1	1	0

Performance Evaluations
- The optimal solution obtained from PuLP solver
- The total up-time increases by ~12% as the number of workers increase from 2 to 6
- There is a significant worker idle time due to the inter-dependencies
- The accuracy of the genetic algorithm is no worse than 1/1.04 times that of the optimal value

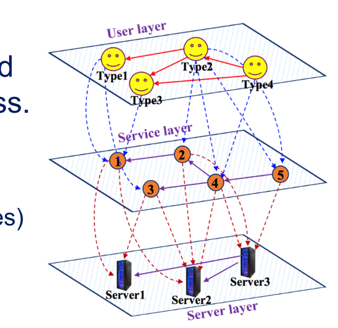




I. El-Shekeil, A. Pal, K. Kant, Progressive recovery of interdependent services in enterprise data centers, in: IEEE Resilience Week, 2016, pp. 27–32.

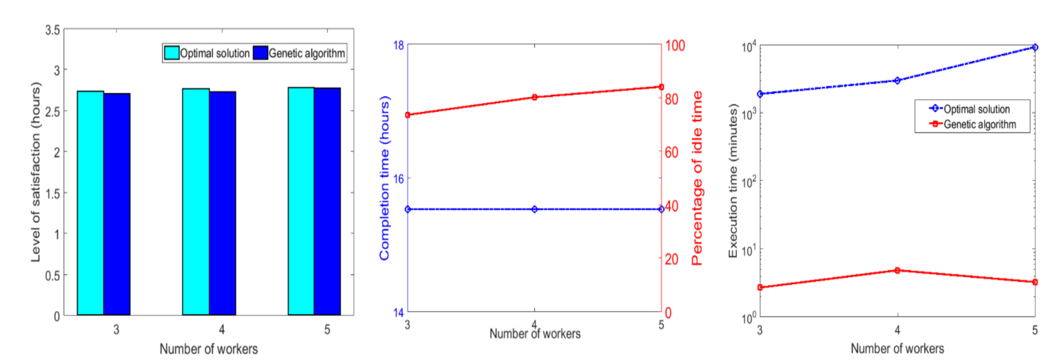
Progressive Recovery in Data Centers

Contribution
- Characterize data center users based on their service requirements, and divide them in different types.
- Goal: Maximize #requests served during the entire recovery process.
- Considerations
  - Interdependencies between various services & servers (or server features)
  - Multiple types of user requests
  - Human-related constraints and expertise.



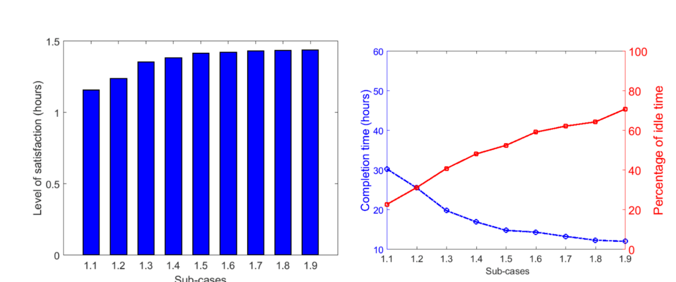
3-layer Interdependency Framework
- Layer-1 or user layer consists of different types of user requests
- Layer-2 or service layer consists of the set of services that the enterprise provides
- Layer-3 or server layer consists of the servers that need to be restored to bring back the services

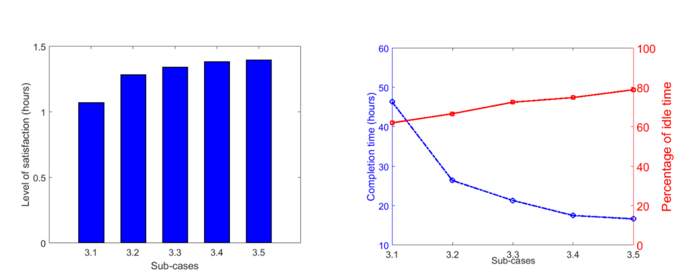
Users	Description	Enterprise services	Necessary servers
Type 1	Authenticate	MS Active Directory	Domain controller
	Access to network resources	DNS	DNS server
Type 2	Send emails to internal users	Microsoft Exchange	Internal email servers
Type 3	Run payroll	Human Resources	Payroll server and employee database
	Access Project	Microsoft Project	Project application and database
Type 4	Send email to external users	Microsoft Exchange	Email forwarding servers
	Access documents in SharePoint	Microsoft SharePoint	SharePoint application and database

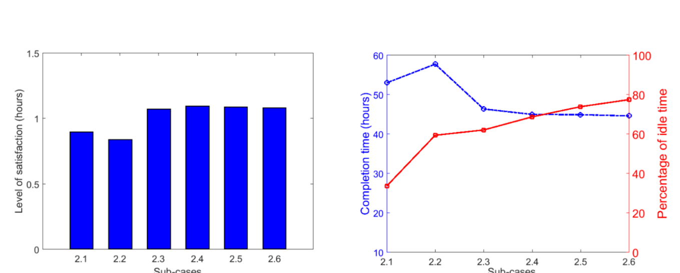


Cases	Sub-cases	W	Worker skills
Case 1	1.1	3	$\mathbb{W}_1, \mathbb{W}_2, \mathbb{W}_3 \rightarrow$ Group 1-7
	1.2	4	$\mathbb{W}_1 - \mathbb{W}_4 \rightarrow$ Group 1-7
	1.3	6	$\mathbb{W}_1 - \mathbb{W}_6 \rightarrow$ Group 1-7
	1.4	8	$\mathbb{W}_1 - \mathbb{W}_8 \rightarrow$ Group 1-7
	1.5	10	$\mathbb{W}_1 - \mathbb{W}_{10} \rightarrow$ Group 1-7
	1.6	12	$\mathbb{W}_1 - \mathbb{W}_{12} \rightarrow$ Group 1-7
	1.7	14	$\mathbb{W}_1 - \mathbb{W}_{14} \rightarrow$ Group 1-7
	1.8	16	$\mathbb{W}_1 - \mathbb{W}_{16} \rightarrow$ Group 1-7
	1.9	20	$\mathbb{W}_1 - \mathbb{W}_{20} \rightarrow$ Group 1-7
Case 2	2.1	2	$\mathbb{W}_1 \rightarrow$ Group 1,6,7; $\mathbb{W}_2 \rightarrow$ Group 2-5
	2.2	3	$\mathbb{W}_1 \rightarrow$ Group 1-2; $\mathbb{W}_2 \rightarrow$ Group 3-4; $\mathbb{W}_3 \rightarrow$ Group 5-7
	2.3	4	$\mathbb{W}_1 \rightarrow$ Group 1; $\mathbb{W}_2 \rightarrow$ Group 2,4,5; $\mathbb{W}_3 \rightarrow$ Group 3; $\mathbb{W}_4 \rightarrow$ Group 6,7
	2.4	5	$\mathbb{W}_1 \rightarrow$ Group 1; $\mathbb{W}_2 \rightarrow$ Group 2,4,5; $\mathbb{W}_3 \rightarrow$ Group 3; $\mathbb{W}_4 \rightarrow$ Group 6; $\mathbb{W}_5 \rightarrow$ Group 7
	2.5	6	$\mathbb{W}_1 \rightarrow$ Group 1; $\mathbb{W}_2 \rightarrow$ Group 2,5; $\mathbb{W}_3 \rightarrow$ Group 3; $\mathbb{W}_4 \rightarrow$ Group 4; $\mathbb{W}_5 \rightarrow$ Group 6; $\mathbb{W}_6 \rightarrow$ Group 7
	2.6	7	$\mathbb{W}_1 \rightarrow$ Group 1; $\mathbb{W}_2 \rightarrow$ Group 2; $\mathbb{W}_3 \rightarrow$ Group 3; $\mathbb{W}_4 \rightarrow$ Group 4; $\mathbb{W}_5 \rightarrow$ Group 5; $\mathbb{W}_6 \rightarrow$ Group 6; $\mathbb{W}_7 \rightarrow$ Group 7
Case 3	3.1	4	$\mathbb{W}_1 \rightarrow$ Group 1; $\mathbb{W}_2 \rightarrow$ Group 2,4,5; $\mathbb{W}_3 \rightarrow$ Group 3; $\mathbb{W}_4 \rightarrow$ Group 6,7
	3.2	8	$\mathbb{W}_1, \mathbb{W}_2 \rightarrow$ Group 1; $\mathbb{W}_3, \mathbb{W}_4 \rightarrow$ Group 2,4,5; $\mathbb{W}_5, \mathbb{W}_6 \rightarrow$ Group 3; $\mathbb{W}_7, \mathbb{W}_8 \rightarrow$ Group 6,7
	3.3	12	$\mathbb{W}_1 - \mathbb{W}_3 \rightarrow$ Group 1; $\mathbb{W}_4 - \mathbb{W}_6 \rightarrow$ Group 2,4,5; $\mathbb{W}_7 - \mathbb{W}_9 \rightarrow$ Group 3; $\mathbb{W}_{10} - \mathbb{W}_{12} \rightarrow$ Group 6,7
	3.4	16	$\mathbb{W}_1 - \mathbb{W}_4 \rightarrow$ Group 1; $\mathbb{W}_5 - \mathbb{W}_8 \rightarrow$ Group 2,4,5; $\mathbb{W}_9 - \mathbb{W}_{12} \rightarrow$ Group 3; $\mathbb{W}_{13} - \mathbb{W}_{16} \rightarrow$ Group 6,7
	3.5	20	$\mathbb{W}_1 - \mathbb{W}_5 \rightarrow$ Group 1; $\mathbb{W}_6 - \mathbb{W}_{10} \rightarrow$ Group 2,4,5; $\mathbb{W}_{11} - \mathbb{W}_{15} \rightarrow$ Group 3; $\mathbb{W}_{16} - \mathbb{W}_{20} \rightarrow$ Group 6,7

Group #	Technology	# of Servers
1	Oracle DB	30
2	MS SQL	8
3	.NET	28
4	Java	7
5	Microsoft	7
6	Oracle App	10
7	Others	17







Performance Evaluations
- Case 1: All workers are fully skilled
  - Average types of user requests up-time increase with the increase of number of workers
  - Improvement do not scale with the increase in number of workers
- Case 2 and 3: Each worker has expertise to restore servers of certain group or set of groups
  - Uptime of user requests does not increase consistently with the increase of the number of workers
  - Expected because workers do not share expertise and can only work on limited number of server
- The completion time is less in case 1 compared to case 2 and 3
  - Every worker in case 1 has the skills to restore all servers

Ongoing and Future work
- Accurate dependencies are not known in practice all the time
  - Studying the use of uncertainty theory in this context
- Some services may support partial load if some critical servers are restored
  - We will examine this consideration in our model to make it more practical

I. El-Shekeil, A. Pal, K. Kant, PRECESION: Progressive Recovery and Restoration Planning of Interdependent Services in Enterprise Data Centers, Submitted to Elsevier DCN Journal

IP Address Consolidation and Reconfiguration in Enterprise Networks

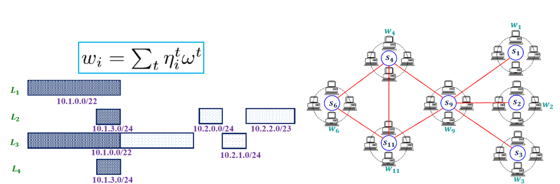
Introduction
- Private IP addressing is used inside enterprise networks
- Different enterprises or even different locations/business units of the same enterprise use the same IP address ranges
  - As long while those networks are separate
- During mergers and acquisitions, or network consolidations within an enterprise
  - Overlapped and conflicted IP segments (subnets) arise frequently
    - Result in misrouting or endless looping of traffic if not corrected
  - Must be identified and resolved
  - The combined network may unnecessarily use many disparate IP address ranges
    - Increases the size of the routing tables
    - Makes routing integrity verification difficult
- Resolving IP ranges overlaps and conflicts
  - IP subnet change
    - Need to coordinate changing firewall rules, DNS entries, proxy servers, load balancers, etc.
  - Network Address Translation (NAT)
    - NAT an alternative solution to IP change
    - Some applications don’t work natively with NAT; additional administration cost
- Networks may use many dis-contiguous IP address ranges
  - Increases the size of the routing tables
  - Makes routing integrity verification difficult
- The problem is further complicated in case of partial merger
  - Company A may split itself into parts A1 and A2
  - A2 merges with company B
  - Routes between A1 and A2 must be restricted via suitable firewalls
  - It is necessary to establish new routes between A2 and B that do not go through A1
  - Existing misconfigurations, overlaps and conflicts within and across entities make this transition more challenging

Contribution:
- We identify different conflict scenarios and consider ways of resolving those conflicts to minimize manual changes
- Consolidating the subnets in large organizations
  - Such that the reassignment of IP addresses (which often must be done manually) is minimized
  - The number of distinct routes that must be recorded in the routing table is minimized

Address Conflict Resolution
- Objective: Minimize IP address change cost to resolve all conflicts
- Step1: Identify subnets for movement
  - Subnets in location L3 overlap with locations L1, L2 and L4
  - There are overlaps in between locations L1, L2 and L4 itself.
- Step 2: Form the conflict graph
  - Vertices: subnets, Edges: conflicts
- The cost (weight) of a subnet (vertex)
  - The number of physical entities that needs to be manually
  - Their relative importance depending on their types
- Step 3: Build the Maximum weight independent set (WIS) of the conflict graph
  - To retain the best possible combination of subnets and change others
- The WIS problem is NP-hard

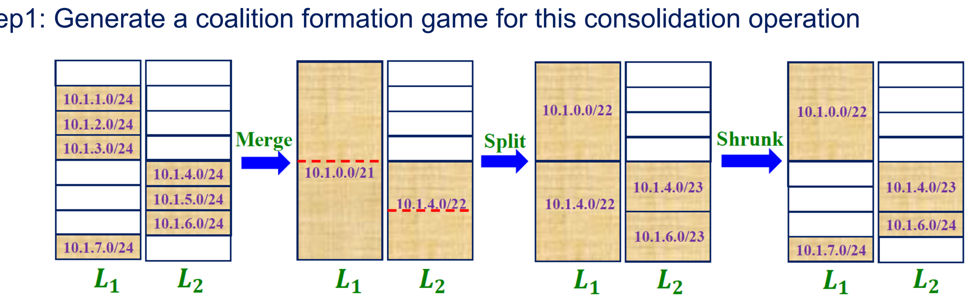
Subnet no	Location L	Subnet Addr	#Active IPs of type 1	2	3
s_1	1	10.1.0.0/24	10	12	0
s_2	1	10.1.1.0/24	200	1	1
s_3	1	10.1.2.0/24	200	0	0
s_4	1	10.1.3.0/24	200	0	0
s_5	2	10.2.0.0/24	200	0	0
s_6	2	10.1.3.0/24	200	10	0
s_7	2	10.2.2.0/24	200	10	0
s_8	2	10.2.3.0/24	200	0	0
s_9	3	10.1.0.0/22	500	0	0
s_{10}	3	10.2.1.0/24	0	10	0
s_{11}	4	10.1.3.0/24	0	0	100

$w_i = \sum_t \eta_i^t \omega^t$

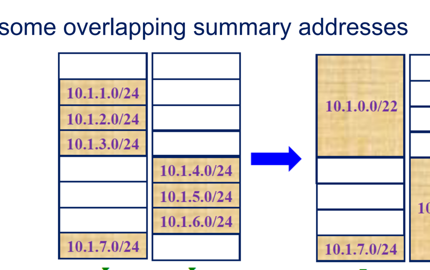


$$\begin{aligned} \text{Maximize} \quad & \sum_{i \in V} w_i x_i \\ \text{subject to} \quad & x_i + x_j \le 1, \quad \forall (i,j) \in E \\ & x_i \in \{0,1\} \quad \forall i \in V \end{aligned}$$

Address Space Consolidation
- After the conflict resolution and reallocation stage the subnets are now in a non-conflicting stage
- Address space consolidation stage
- Purpose: Routing table entries of the routers and gateways are largely minimized
- Step1: Generate a coalition formation game for this consolidation operation

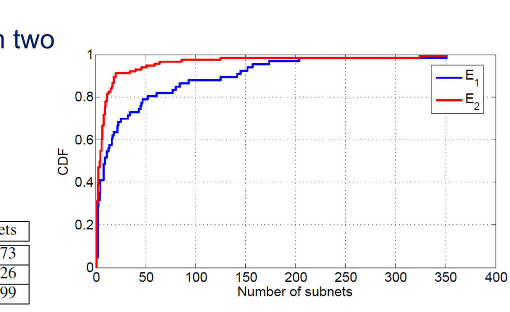


- Step2: Resolve another level of conflicts that may arise due to the consolidation phase
  - Removing conflict due to the consolidation
  - After the coalition formation stage there may be some overlapping summary addresses
  - Solve the WIS:
    - Vertices are the coalitions
    - Edge weights are the cumulative weights of all the subnets in that coalitions

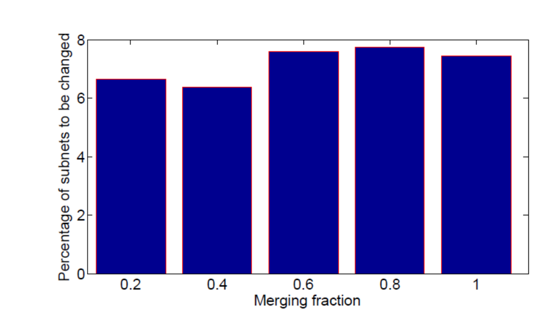


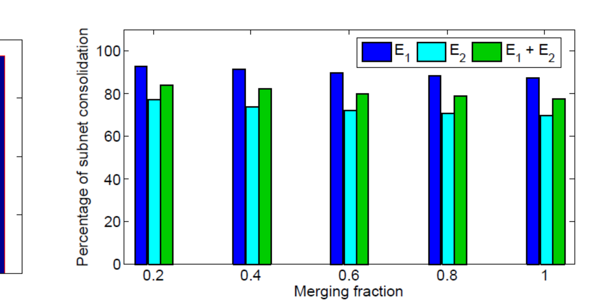
Performance Evaluations
- Simulations are done based on real-world data from two recent mergers E_1 and E_2
- Distribution of the number of subnets in different locations
  - The distribution is highly skewed
  - 80-90% of the locations have < 50 subnets
  - Maximum number of subnets at any location is around 350

Scenario	Locations	Subnets
E_1	66	2473
E_2	113	1726
$E_1 + E_2$	179	4199



Performance Evaluations
- Percentage of subnets to be changed for a conflict-free merging of E_1 and E_2
  - Number of subnets that needs to be changed varies from 6-8%
  - By changing few subnet addresses the enterprises can effectively remove all conflicts
  - The amount of change needed does not vary significantly with the increase in merging fraction
    - Because of the highly skewed nature of the subnet distribution in these enterprises
- Percentage of subnet entries reduced after the consolidation process
  - Reduce the subnet entries by 80-90%
  - The subnet consolidation is also effective even within an enterprise
  - The amount of improvement does not change significantly with the increasing merging fraction → skewed nature of the distribution





Conclusions
- We discussed the problem of resolving IP address conflicts and consolidating the IP address space in large enterprises
  - Devised a conflict resolution scheme
  - Discussed an address space consolidation mechanism
  - Preliminary study to overcome the address space issues in large enterprises especially at the time of merging
- Resolve the subnet conflicts by changing 6-8% of the subnet addresses
- Reduces the number of subnet entries by 80-90% by consolidating the subnet entries.

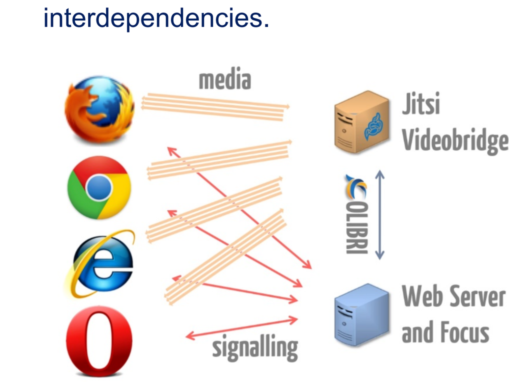
Ongoing and Future work
- Studying the aggressive change of subnet addresses
  - Tradeoff between the extent of IP address changes and the resulting reduction in number of subnets
- Optimal ordering of the address change may be too expensive
  - On-line methods of conflict resolution/consolidation that can be run concurrently with the normal operation of the network
- A complete solution to ensure smooth functioning of all resources that depend upon IP addresses (e.g., firewalls, routers, DNS, load balancers, etc.)

I. El-Shekeil, A. Pal, and K. Kant. “IP Address Consolidation and Reconfiguration in Enterprise Networks”. In: 2016 25th International Conference on Computer Communication and Networks (ICCCN). 2016, pp. 1–9.

Fast Diagnosis and Localization of Problem Area(s) in Data Centers

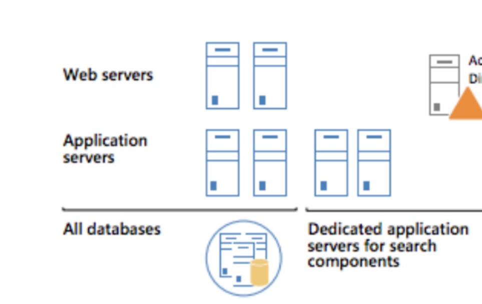
Introduction
- Fault diagnosis comprises of detection and localization
- Challenging problem
  - Many components,
  - large set of configuration parameters
  - restrictions on parameter values,
  - interdependencies between parms, services, etc.
  - Virtualization makes finding the root cause of failures and localization of faults even more complex.
  - Testing some services require simultaneously running tests from multiple stations

Goal
- Develop mechanisms to find the optimal set of tests to localize the fault(s) in the presence of multiple layers of capabilities/services and interdependencies.



E. Ivov, “Hangout-like video conferences with jitsi videobridge and xmpp,” 2013.

Ongoing Works
- Detection failures
  - Select the minimal number of tests to detect failures
  -
- Fault localization
  - set of tests that detected a failure is not sufficient for fault localization
  - goal is to minimize the diagnoses time by selecting the optimal set of tests that lead to finding the fault.



Technical diagrams for SharePoint 2013, https://technet.microsoft.com/en-us/library/cc263199.aspx

I. El-Shekeil, A. Pal, and K. Kant. “Fast Diagnosis and Localization of Problem Area(s) in Data Centers”. Work in progress