Advances in SAN Coverage Architectural Modeling

Trace coverage, modeling, and analysis across IBM systems test labs world-wide

Tara Astigarraga
IBM CHQ
Rochester, NY
United States
asti@us.ibm.com

Yoram Adler and Orna Raz
IBM Research
Haifa, Israel
adler@il.ibm.com
ornar@il.ibm.com

Robin Elaiho and Sheri Jackson
IBM Systems
Tucson, AZ
United States
rlelaiho@us.ibm.com
sheribj@us.ibm.com

Jose Roberto Mosqueda Mejia
IBM Systems
Guadalajara, Mexico
mosqueda@mx1.ibm.com

***Abstract* - Storage Area Networks (SAN) architectural solutions are highly complex, often with enterprise class quality requirements. To perform end-to-end customer-like SAN testing, multiple complex interoperability test labs are necessary. One key factor in field quality is test coverage; in distributed test environments this requires a centralized view and coverage model across the different areas of test. We define centralized coverage models and apply our novel trace coverage technology to automatically populate these models. Early results indicate that we are able to create a centralized view of SAN architectural coverage across the multitude of IBM test labs world-wide. Moreover, we are able to compare test lab coverage models with customer environments. Since its inception, this distance matrix project has shown added value in many foreseen and unforeseen ways. The largest benefit of this project is the ability to systematically extract and model coverage across a large number of test and client SAN environments, enabling increased coverage without expanding resource requirements or timelines. One of the key success factors for this model is its scalability. The scalability and reach of the distance matrix project has also uncovered additional unforeseen benefits and efficiencies. As the project matures we continue to see improvements, new capabilities, use case extensions and scaled architectural coverage advances.**

Keywords - Software Test; SAN Coverage; SAN Architecture Coverage; SAN Architectural Modeling; Software Engineering; SAN Test; System Test; Distance Matrix; Trace Coverage Models; SAN Hardware Test Coverage; Test Coverage Analysis; IBM Test; IBM Systems Test.

I. INTRODUCTION AND MOTIVATION

This article is an updated and extended version of a work-in-progress report that was presented and published at the VALID 2014 conference in Nice, France [1].

IBM is a global technology and innovation company with more than 400,000 employees serving clients in 170 countries [2]. The IBM test structure consists of thousands of test engineers world-wide. In addition to function test teams for product streams, there is also an entire world-wide organization of many hundreds of people dedicated to systems and solution test. IBM has interoperability and complex test labs world-wide [3]. Systems test strategies focus on customer-like, end-to-end solution integration testing designed to cover the architectural design points of a broad range of customer environments and operations with the end goal of increased early discovery of high-impact defects, resulting in increased quality solutions. One key area of systems and solution test is innovation. As configurations supported continue to climb, with over 237 million configurations supported on the IBM System Storage Interoperation Center (SSIC) site, test engineers are continually challenged to find ways to test smarter [4]. As part of ongoing test cycles test engineers are continually updating their environments to best represent ever changing technologies, configurations, architectures and integrated technologies and virtualization layers in the server, storage and network environments. In order to keep pace with technology demands test engineers are expected to perform integrated systems planning and recommend new technologies, techniques or automation that will enhance current systems test coverage and support the larger goal of optimized test coverage and minimized field incidents.

One IBM test transformational project we have been working on is the storage area network (SAN) distance matrix project. This project arose from the IBM Test and Research groups as a joint-project aimed at better quantifying and understanding the systems test SAN coverage across IBM test groups world-wide [1]. The project emerged from IBM systems test as a set of requirements and early vision of automated capabilities for SAN coverage modeling. In partnership with the IBM Haifa Research lab we formed a small working team and began to document, model and prototype innovative solutions. At the start of this project we had many questions related to world-wide hardware and SAN coverage, but we did not have a centralized view of the test labs across IBM. Test labs were designed, built, monitored and architected on an individual basis without the ability to easily extract coverage models across the test locations and understand on a global scale the combined IBM test coverage model. Another missing piece was the ability to do broad coverage reviews looking at IBM test labs in comparison to its clients. We have always worked hard to build our test environments to include key characteristics from a diverse range of IBM clients, however, we lacked data environment modeling tools to take customer environment variables and systematically map them against our test environments. The IBM distance matrix project was designed to address these concerns and help to centralize visibility and configuration details about the systems and solution SAN test labs across IBM and its clients.

The SAN distance matrix project has the abilities to look at key architectural design points across the SAN environments and extract coverage summaries for deep-dive reviews, comparisons and ultimately architecture changes to continually improve our solution test coverage, scalability and customer focus.

In this paper, we will further describe the SAN distance matrix project goals, methods, and advancements achieved within the following sections: Section II. Related Work, Section III. Project Strategy, Section IV. Collecting Data, Section V. Analyzing the Data, Section VI. Early Results, Section VII. Additional Benefits Realized, and Section VIII. Conclusion and Further Development.

II. RELATED WORK

The SAN distance matrix project provides a means for better quantifying and understanding the SAN coverage over the entire test organization, across its different test groups. The same solution also provides the ability to do broad coverage reviews looking at an organization test labs in comparison to its clients. No existing technology that we are aware of provides that.

There are existing tools, including Cisco Data Center Network Manager [5] and Brocade Network Advisor [6] that provide in-depth and detailed modeling capabilities for single environments or environments managed by a single entity; however, there is a gap in the ability to easily look across a heterogeneous group of environments controlled by different companies, divisions or organizations.

There are other tools that can be used to get a consolidated view of the status and performance of your storage and network devices, including SolarWinds Storage Manager [7] and IBM Tivoli Monitoring [8], however, there is a gap in the ability to define new values or parameters to be monitored and generate reports across environments controlled by different companies, divisions or organizations. Additionally, these tools are not developed with the purpose of comparing coverage and architectural models across environments.

In our solution, we deploy the novel idea of trace coverage, relying on the extraction of a functional model from existing switch dump data. In functional modeling and one of its optimization techniques Combinatorial Test Design (CTD), the system under test is modeled as a set of parameters, respective values, and restrictions on value combinations that may not appear together in a test. A test in this setting is a tuple in which every parameter gets a single value. A combinatorial algorithm is applied in order to come up with a test plan (a set of tests) that covers all required interactions between parameters. Kuhn, Wallace and Gallo [9] conducted an empirical study on the interactions that cause faults in software that is the basis for the rationale behind CTD. Nie and Leung [10] provide a recent survey on CTD. The SAN distance matrix that we create can be viewed as a functional model. This functional model could be optimized with tools such as IBM Functional Coverage Unified Solution (IBM FOCUS) [11, 12]. In our case, we automatically extract the model from switch dumps. We term the creation of coverage models from existing traces 'trace coverage'.

III. PROJECT STRATEGY

The SAN distance matrix project strategy is composed of two main phases as shown in Figure 1. Phase 1 consists of collecting switch dump data; a scripted process to extract key data across multiple SAN environments. By identifying key switch data, the script we execute has little impact to the regular activity of the switches. Test team members, with expertise in configuring complex SAN solutions, and in-depth knowledge of best practices and supported configurations, identified the set of switch commands to collect the data required for phase 2 of the project. While the initial set of commands executed were chosen carefully we also built the project structure and scripting capabilities with the assumption that the list of commands executed will likely grow, change and expand with time and project maturity.

Phase 2 consists of analyzing the collected switch dump data. Within this phase, hundreds of switch dump data files from various test and customer labs collected in phase 1 environments were analyzed and parsed into a structured format that would aid in our comparison, analysis and reporting of the collected data.

The SAN distance matrix project is currently extracting data quarterly across teams world-wide. While we chose to implement an ongoing quarterly collection cycle, we also have the capabilities to kick-off a collection stream at any time should the need for new or specified data emerge from any given lab or combination of labs across IBM. In the following sections, we describe each phase and activities in detail

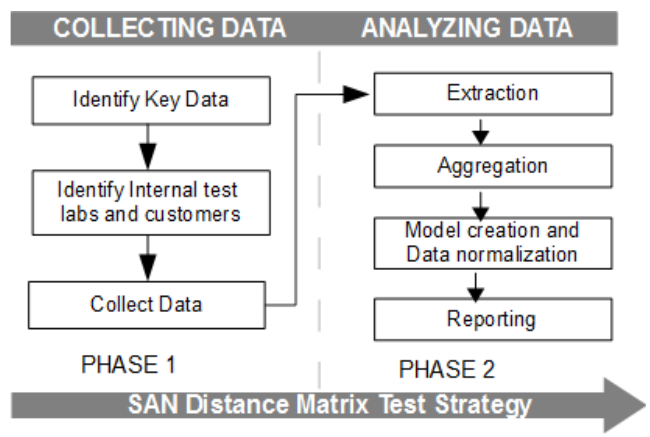

Figure 1. SAN Distance Matrix Project Strategy

IV. COLLECTING DATA

The data collection phase is composed of 3 main activities:

A. *Identify Key Data*

Using switch dump data, we've selected specific switch query commands, which are used to systematically extract the key data for usage and coverage statistics across different IBM test teams and select customers. The switch query commands allow us to extract dump data focused on topologies, coverage points, performance, utilization and other environmental aspects in our SANs. Topology data points include port speeds, port counts and port types. Environmental data points include the switch hardware platforms, protocols used such as Fibre Channel (FC), Fibre Channel over Ethernet (FCoE) and

Fibre Channel over IP (FCIP), code levels, switch up-time and switch special functions/features that are enabled.

Architectural design points include port-channel/trunk usage, virtual storage area network or virtual local area network (VSAN/VLAN) coverage, virtualization data and initiator/target to inter-switch link ratios. Using this raw dump data and subsequent processing logic, we were able to create a summary of all the different port speeds being tested, switch utilization rates, general architecture modeling and software and hardware versions being covered across the initial scope of IBM systems test and customer environments. Additional insights of interest that were identified via analysis of key data include host to storage ratios, host and storage to ISL ratios, architectural design complexity and port and bandwidth utilization rates.

This approach enabled us to easily gather promising data, avoid limitations of manual investigation and create a model that is scalable and easy to use for ongoing analysis. Further, the data structure and quarterly data pulls provided us with results and data that we are then able to use in compiling trending reports and pattern discovery across IBM test labs world-wide

B. *Identify internal test labs and customers*

The initial IBM test teams added to the project scope were selected based on our team's previous connections and working relationships with the different IBM systems test labs. We had an introductory meeting with several teams that covered the objectives, process and benefits of the SAN distance matrix project. Participation at the early stages of this project was voluntary. As the project progressed and initial results were reviewed with vested management teams the scope was expanded to include a broader list of test labs across systems test and even to include select function test labs.

The process to select customers and include them in the project was different. Since we do not have direct access into customer labs, we looked into different options. One option we chose was to leverage existing client relationships and the IBM customer advocate program to invite customers to submit data for use in this project. Clients who submitted data were incented to do so with the goal of better environment understanding and future IBM test coverage models built utilizing their environment architecture as a piece of the modeling puzzle for future test cycles. Additionally, we reached out to the IBM SAN support organization and requested interlock capabilities to allow selected client dump data be utilized for modeling capabilities for the distance matrix project. These two avenues have been successful in the early stages of the project and we continue to look for ways to systematically expand the number of clients we are able to include. The goal is to ensure that the customer data sets we receive and leverage are balanced across industry, company size, scale, and environment complexity. Although we are not able to replicate and test every environment data set we receive, the distance matrix project allows us to extract key data points and ensure those combined client data points are used as coverage requirements in upcoming test cycles.

C. *Collect Data*

For data collection within internal IBM test labs, we designed automated scripts to collect the dumps and command query data. The scripts use a source comma separated values (CSV) file, which contains the list of switches, switch types, IPs and credentials. It uses a telnet connection to login the different switches, then executes the appropriate switch query commands and generates a log containing the switch dump data for each switch. The series of commands run in the background and are non-disruptive to the test lab's switch fabric. For the initial scope of this project a subset of IBM test labs was chosen. That subset group included fourteen IBM system test labs, which contained a combined total of four hundred and eighty five SAN top of rack edge and core switches. The output from the fourteen test labs is raw data that consists of a text file for each of the four hundred and eighty five switches that need consolidation and further formatting of the pertinent information for use in the project.

For data collection at customer locations, we do not execute any command in the customer's environment. We instead ask them to send in a switch support dump or specified command query output depending on the brand of switches deployed in the customer environments. The dump information supplied by customers is similar in nature to the data we collected internally, and will also need further formatting during analysis of the data.

V. ANALYZING THE DATA

The problem: SAN switch dump data is heterogeneous based on switch vendor, platform and code levels. Further, the data is collected from various sources and unique collection methods across IBM test labs and customer locations.

The switch dump data is a text file created for each switch. It contains output from multiple switch queries/commands that are executed against the switch. Each switch type has its own set of commands and a unique output format.

The goal: Parse the various switch dump semi structured data and transfer it to structured format.

The solution: the solution relies on the novel notion of trace coverage and the IBM EASER [13] easy log search tool.

Trace coverage extracts report data from traces that already exist in a system or are easy to create according to a defined coverage model. The coverage model can be code coverage – automatically created from the code locations that emit trace data, or functional coverage – manually created to define the system configuration or behavior. In SAN coverage, the traces are created by switch dumps, and the coverage model is a functional coverage of the possible SAN environments. A functional coverage model describes the test space in terms of variation points or attributes and their values. For example, attributes may be port types, port rates, or port utilization percentages. The IBM EASER tool supports extraction of semi-structured data from traces and transforms it into a structured format. It provides both a graphical user interface (GUI) for interactive exploration and a headless mode of operation for automating the extraction and analysis process.

After defining a functional coverage model, the IBM EASER tool is used to extract, aggregate, and compare data:

- Extract functional model values from switch dumps
- Aggregate the coverage of multiple logs from both customers and IBM test labs.
- Compare coverage between a defined set and subsets of labs by generating multiple summary reports.

The SAN Test functional coverage model is extendable; it can be updated to include additional values seen in customer environments. The collected data is aggregated by IBM test groups and customers and definitions are flexible and can be supplied by the end-user.

The automated functional coverage analysis process includes three phases: Extraction, Aggregation and Reporting.

A. Extraction

The functional model attributes' values are extracted from each switch dump file. By using EASER, the log is divided into entries and then the relevant data is extracted, computed and inserted into the relevant model attributes' values. One file with attributes and values is created for each switch log file.

Figure 2 and Figure 4 are excerpts from the original switch dump file, while Figure 3 and Figure 5 are a result of the various stages of our analysis.

Figure 2 shows a sample of a single cisco_fc switch dump data log file, which is created using the automated scripts. In addition to the switch summary, the log file includes the switch query commands and corresponding switch data output. The figure shows a single entry out of the entire switch dump. This is achieved via the EASER parser through its support for smart data partitioning.

```
2014-03-24 14:24:55 INFO Switch Summary
   Name:     slswc10f2cis
   IPAddr:   9.11.195.75
   Brand:    cisco
   Type:     fc
   Area:     cisco san
   Location: tucson
2014-03-24 14:24:55 INFO Log in to  device slswc10f2cis.tuc.stglabs.ibm.com
2014-03-24 14:25:00 INFO Log in to slswc10f2cis.tuc.stglabs.ibm.com successful
2014-03-24 14:25:00 INFO --------------------------------------------------
```

Figure 2. Switch Log File Sample

The EASER parser extracts values from the entry in Figure 2 and updates them into the attributes shown in Figure 3. Figure 3 shows for each category (column header)
it's extracted value. For example, the SwitchType() category has the value cisco_fc, which was extracted from the original switch dump entry, shown in Figure 2.

Area	Locatio	SwitchType	Name	Valu	Name1	Value2
test-team1	tucson	brocade_fc	TotalSwitchWithDataCount	51	SwitchesNames	(XXXXXX
test-team1	tucson	cisco_fc	TotalSwitchWithDataCount	6	SwitchesNames	(XXXXXX
test-team1	tucson	cisco_fcoe	TotalSwitchWithDataCount	8	SwitchesNames	(XXXXXX
test-team1	tucson	brocade_fcoe	TotalSwitchWithDataCount	4	SwitchesNames	(XXXXXX
test-team2	hursley	brocade_fc	TotalSwitchWithDataCount	100	SwitchesNames	(XXXXXX
test-team2	hursley	cisco_fc	TotalSwitchWithDataCount	4	SwitchesNames	(XXXXXX

Figure 3. Parser extracted data Sample

Figure 4 shows a sample of an entry in a Cisco switch dump data extract, as extracted by the EASER parser. The EASER parser then uses this data to compute category summary values. These values become part of the distance model, as shown in Figure 5.

```
--------------------------------------------------------------------------------
Interface  Vsan   Admin  Admin   Status        SFP    Oper  Oper   Port
                  Mode   Trunk                        Mode  Speed  Channel
                         Mode                               (Gbps)
--------------------------------------------------------------------------------
fc1/30     200    F      auto    up            swl    F     16     --
fc1/31     200    auto   off     up            swl    F     4      --
fc1/32     200    auto   off     notConnected  swl     --   --     --
fc1/33     200    auto   off     up            swl    F     16     --
fc1/34     200    auto   off     up            swl    F     16     --
fc1/35     200    auto   off     up            swl    F     16     --
fc1/36     200    auto   auto    notConnected  swl     --   --     --
fc1/37     200    auto   off     up            swl    F     16     --
fc1/38     200    auto   off     up            swl    F     16     --
fc1/39     200    auto   off     up            swl    F     8      --
fc1/40     200    auto   off     up            swl    F     8      --
fc2/20     100    auto   auto    up            swl    E     8      4
fc2/21     100    auto   auto    up            swl    E     8      1
fc2/22     100    auto   auto    up            swl    E     8      1
fc2/23     100    auto   auto    up            swl    E     8      1
fc2/24     100    auto   auto    up            swl    E     8      1
fc3/43     200    E      off     up            swl    E     8      9
fc3/44     200    E      off     up            swl    E     8      9
--------------------------------------------------------------------------------
Interface              Vsan  Admin  Status         Oper  Oper    IP
                             Trunk                 Mode  Speed   Address
                             Mode                        (Gbps)
--------------------------------------------------------------------------------
port-channel1          100   auto   up              E    64      --
port-channel2          200   auto   up              E    64      --
port-channel4          100   auto   up              E    32      --
port-channel8          200   off    up              E    8       --
port-channel9          200   off    up              E    16      --
```

Figure 4. Cisco MDS extract data snippet

Figure 5 shows an example of an abbreviated model per switch dump. For example, the line name TotalFcPortsCount in the figure is calculated by counting the number of relevant entries in the original switch dump. The line name FcFPortSpeedsUsed aggregates the speeds used for Cisco FC F-Ports from the original switch dump. For the sake of brevity, only a small portion of the parser extract and model data are shown in these figures.

Area	Location	SwitchType	Name	Value
test_team1	Tucson, AZ	cisco_fc	TotalSwitchWithDataCount	1
test_team1	Tucson, AZ	cisco_fc	TotalFcPortsCount	192
test_team1	Tucson, AZ	cisco_fc	TotalFcPortsLoggedIn	74
test_team1	Tucson, AZ	cisco_fc	PercentageFcPortsUtilized	38
test_team1	Tucson, AZ	cisco_fc	TotalVSANsCount	2
test_team1	Tucson, AZ	cisco_fc	AvgFcEPortRate	8
test_team1	Tucson, AZ	cisco_fc	HighesFctEPortRate	8
test_team1	Tucson, AZ	cisco_fc	AvgFcFPortRate	10
test_team1	Tucson, AZ	cisco_fc	HighestFcFPortRate	16
test_team1	Tucson, AZ	cisco_fc	FcFPortSpeedsUsed	(16,4,8)
test_team1	Tucson, AZ	cisco_fc	PortChannelUsage	yes
test_team1	Tucson, AZ	cisco_fc	LongestKernelUptime	41
test_team1	Tucson, AZ	cisco_fc	ShortestKernelUptime	41
test_team1	Tucson, AZ	cisco_fc	AssociatedSWHWVersions	6.2(7),cisco_MDS_9710

Figure 5. Cisco MDS single switch abbreviated base model.

B. Aggregation

All data from **Extraction** output files is grouped by switch type and switch locations into three files:

1. Summary of all entries,
2. Summary of all samples that contains “full data”
3. Summary of files with “no” or “partial” data.

The contents of the first two files reflect the model: Attributes and their aggregated values from the extraction phase output files. The third file contains an ‘illegal’ list that should be reviewed by IBM experts for the cause of the failure during collection. Figure 6 contains a subset example.

Area	Location	SwitchType	Name	Value
Test-lab-i	Austin, TX	cisco_fc	#Switches	6
Test-lab-c	Tucson, AZ	cisco_fc	#Switches	26
Test-lab-n	Raleigh, NC	brocade_fc	#Switches	5
Test-lab-d	Tucson, AZ	brocade_fc	#Switches	13
Test-lab-o	China	cisco_fc	#Switches	4
Test-lab-b	Tucson, AZ	brocade_fc	#Switches	20
Client1	NY	cisco_fc	#Switches	14
Test-lab-i	Austin, TX	cisco_fc	PortCount	516
Test-lab-i	Austin, TX	brocade_fc	PortCount	112
Test-lab-c	Tucson, AZ	cisco_fc	PortCount	1394
Test-lab-n	Raleigh, NC	brocade_fc	PortCount	568
Test-lab-d	Tucson, AZ	brocade_fc	PortCount	496
Test-lab-o	China	cisco_fc	PortCount	340
Test-lab-b	Tucson, AZ	brocade_fc	PortCount	2380
Client1	NY	cisco_fc	PortCount	2213
Test-lab-i	Austin, TX	cisco_fc	VSANCount	6
Test-lab-c	Tucson, AZ	cisco_fc	VSANCount	22
Test-lab-n	Raleigh, NC	cisco_fc	VSANCount	6
Test-lab-d	Tucson, AZ	cisco_fc	VSANCount	5
Test-lab-o	China	cisco_fc	VSANCount	34
Client1	NY	cisco_fc	VSANCount	23
Test-lab-n	Raleigh, NC	cisco_fc	PortSpeedsUsed	(4,8)
Test-lab-i	Austin, TX	cisco_fc	PortSpeedsUsed	(2,4,8)
Test-lab-b	Tucson, AZ	brocade_fc	PortSpeedsUsed	(4,8,16)
Test-lab-c	Tucson, AZ	cisco_fc	PortSpeedsUsed	(4,8,10,16)
Client1	NY	cisco_fc	PortSpeedsUsed	(1,2,4,8,10)

Figure 6. Summary of select full data samples

C. Model creation and data normalization

A functional model encapsulates the combination of data and analysis based on human expert knowledge to allow analysis and comparison of configurations. After carefully identifying the key data points we worked to create functional models and analysis capabilities based on domain expertise in SAN coverage and SAN test architecture. We created several different functional models. Functional models that

1. Identify interesting information in switch dumps, per switch type
2. Summarize the switch information per test lab
3. Summarize the switch information across test labs.

For all the models we worked with the SAN architecture and test experts to both identify the interesting data and define the attribute resulting from various computations over these data. Extracting the right data is prerequisite for a functional model, however, understanding, normalizing and properly qualifying the data values is essential to creating reliable analysis.

Figure 5 provides an example of a model that identifies interesting attributes. These attributes are computed from the raw switch data. Figure 6 provides an example of a model that summarizes the information per test lab. Figure 8 provides an example of a model that summarizes the information across test labs.

We found it essential to define model attributes that summarize data into single measures. This allows immediate comparison of configurations among different test labs and customers. For example, looking at Figure 6 we see a significant difference between Test-lab-n and Test-lab-b in the Brocade FC port count (2380 compared with 568).

Another example can be seen in Figure 8. In terms of Cisco NX-OS code levels. Test-lab-c is more similar to Client1 than Test-lab-i. These examples demonstrate a simple and straight forward comparison between configurations. This allows us to immediately spot differences at an eye glance. If needed, complex comparisons can be defined as well. Of course, the comparison can be automated.

D. Reporting

Data from the **Aggregation** phase is broken into several reports. There are two summary reports types: code levels and machine types, which are based on aggregation summary of all entry files and results report, which contains data including: switch functions, SAN design principles, switch utilization, port speeds, errors, peak traffic rates and average traffic rates. We also took into consideration the switches which may have been offline during the data collection phase. If our scripted process was unable to gather the switch dumps, the parser would attempt to analyze the data and if unsuccessful the parser will create an illegal switch summary report. Figure 7 shows a sample of illegal switches, with their given problem.

Directory	file	problem
/home/switchtoo	cisco_san_fcoe_20150216_152059.txt	switch_with_no_data
/home/switchtoo	brocade_san_fc_20150216_151801.txt	switch_with_no_data
/home/switchtoo	brocade_san_fcoe_20150216_155913.txt	NumberPorts_eq_0

Figure 7. Illegal switch summary

Figure 8 contains an example of number of switches running select Cisco NX-OS code levels from two IBM test labs and one client location. As you can see in Figure 8 Test-lab-c has a large variety of code levels running in its test environment, which include coverage of the levels in use by Client1. However, Test-lab-i has a smaller number of switches in test and the code level coverage is limited to the NX-OS 5.2.x, NX-OS 6.2.x, and NX-OS 7.0.x code streams. In order to best summarize the code coverage the switch code levels have been abstracted to show only two numeric values in the code stream. For example, both NX-OS 6.2.5a and NX-OS 6.2.9 would be referenced as NX-OS 6.2.x. This method of reporting was built into the aggregation model to better categorize and compare broad samples of data across a multitude of client and test labs. Although the data has been abstracted in this model, the full code stream data is also stored in a more detailed model for use by test teams focused more closely on specific switch code qualification test efforts.

Cisco NX-OS	Test-lab-c	Test-lab-i	Client1
3.2.x	4	0	2
3.3.x	3	0	0
4.1.x	3	0	2
5.0.x	5	0	0
5.2.x	8	1	10
6.2.x	6	5	0
7.0.x	4	2	0

Figure 8. Code level sample report

VI. EARLY RESULTS

We established a functional model, which gives a unified view of hundreds of SAN switches. See Figure 9 for details. IBM Systems Test switch count ratio is proportionate to the global SAN market share where Brocade is the #1 player in SAN [14] owning more that 54% of the Fibre Channel market in 2013 [15]. Although Brocade, Cisco and Lenovo are not the only SAN switches in IBM systems test environments, for the distance matrix project we made the conscious decision to focus on these brands to best align with market penetration and the majority of IBM SAN support statements. The goal of the SAN distance matrix project is to extract quarterly data in order to create trend reports, continually update test coverage and to understand what variables are changing or remaining static across test environments.

The first round of analysis completed in December 2013. As stated earlier, we utilized EASER Log Analysis to extract the information from the dumps. Coverage comparisons were established as we reviewed how the different test teams utilized their switches. Upon formulating the data, we created a functional model that has enabled us to provide results to IBM test teams. Those results have proven useful in driving interlock and complementary coverage models between IBM and switch vendors, and ensuring our test environments are representative of our clients.

	Cisco	Brocade	SND	Total
FC	65	335		400
FCoE	25	33	27	85
Total	90	368	27	**485**

Figure 9. Total number of FC and FcoE switches across Systems Test Groups

This information identifies key SAN coverage and test variants. For example, switch type, code versions, switch functions (enabled/disabled) and switch utilization (port speeds, errors, peak traffic rates, and average traffic rates). After analysis and review of the data within our team, we provide deep dive environment cross-test-cell reviews with test technical leads from IBM systems test labs world-wide.

Figure 10 shows a sample summary of two test groups located in Tucson, AZ and Hursley, UK. From this summary, we can easily examine the high-level switch usage across the two test groups. When the data is looked at over time it provides better insight into the environment variability and utilization rates for a given environment. The insight that can be derived from this high-level data summary is valuable, but limited. However, when the high-level utilization numbers are combined with other data factors and SAN coverage analytics, they can present powerful data points for skilled test architects and engineers to utilize in order to better adapt, design and drive the ideal levels of stress across test labs.

The detailed SAN coverage review allows test teams to easily identify their switch utilization rates and compare their environment numbers to a range of customer environments. The utilization data across time provides a better understanding of our global SAN test environments and drill-down capabilities for individual test labs. Additionally, when used in combination with trace coverage analysis teams are able to better perform gap analysis, code coverage reviews, and improve our larger system test coverage strategies.

Area	Location	SwitchType	Name	Value1	Value2
test-team1	tucson	brocade_fc	TotalSwitchWithDataCount	51	
test-team1	tucson	cisco_fc	TotalSwitchWithDataCount	6	
test-team1	tucson	cisco_fcoe	TotalSwitchWithDataCount	8	
test-team1	tucson	brocade_fcoe	TotalSwitchWithDataCount	4	
test-team2	hursley	brocade_fc	TotalSwitchWithDataCount	100	
test-team2	hursley	cisco_fc	TotalSwitchWithDataCount	4	
test-team2	hursley	brocade_fcoe	TotalSwitchWithDataCount	4	
test-team1	tucson	brocade_fc	TotalPortCount	2892	
test-team1	tucson	cisco_fc	TotalPortCount	500	
test-team1	tucson	cisco_fcoe	TotalPortCount	360	
test-team1	tucson	brocade_fcoe	TotalPortCount	124	
test-team2	hursley	brocade_fc	TotalPortCount	3443	
test-team2	hursley	cisco_fc	TotalPortCount	96	
test-team2	hursley	brocade_fcoe	TotalPortCount	128	
test-team1	tucson	brocade_fc	TotalPortsLoggedIn	1667	
test-team1	tucson	cisco_fc	TotalPortsLoggedIn	197	
test-team1	tucson	cisco_fcoe	TotalPortsLoggedIn	127	
test-team1	tucson	brocade_fcoe	TotalPortsLoggedIn	68	
test-team2	hursley	brocade_fc	TotalPortsLoggedIn	1536	
test-team2	hursley	cisco_fc	TotalPortsLoggedIn	23	
test-team2	hursley	brocade_fcoe	TotalPortsLoggedIn	108	
test-team1	tucson	brocade_fc	PercentageFcPortsUtilized	57	
test-team1	tucson	cisco_fc	PercentageFcPortsUtilized	48	
test-team2	hursley	brocade_fc	PercentageFcPortsUtilized	44	
test-team2	hursley	cisco_fc	PercentageFcPortsUtilized	23	
test-team1	tucson	brocade_fc	FcEPortSpeedsUsed	(16,2,4,8)	
test-team2	hursley	brocade_fc	FcEPortSpeedsUsed	(2,4,8)	
test-team1	tucson	brocade_fc	FcFPortSpeedsUsed	(1,16,2,4,8)	
test-team1	tucson	cisco_fc	FcFPortSpeedsUsed	(1,2,4,8)	
test-team2	hursley	brocade_fc	FcFPortSpeedsUsed	(16,2,4,8)	
test-team2	hursley	cisco_fc	FcFPortSpeedsUsed	(2,4)	
test-team1	tucson	brocade_fc	LongestKernelUptime	441	
test-team1	tucson	cisco_fc	LongestKernelUptime	329	
test-team2	hursley	brocade_fc	LongestKernelUptime	152	
test-team2	hursley	cisco_fc	LongestKernelUptime	152	
test-team1	tucson	brocade_fc	ShortestKernelUptime	107	
test-team1	tucson	cisco_fc	ShortestKernelUptime	108	
test-team2	hursley	brocade_fc	ShortestKernelUptime	3	
test-team2	hursley	cisco_fc	ShortestKernelUptime	16	
test-team1	tucson	brocade_fc	AssociatedSoftwareHardwareVersions	v6.4.3f	58.2
test-team1	tucson	brocade_fc	AssociatedSoftwareHardwareVersions	v7.2.1a	58.1
test-team1	tucson	cisco_fc	AssociatedSoftwareHardwareVersions	6.2(9)	cisco_MDS_91
test-team1	tucson	cisco_fc	AssociatedSoftwareHardwareVersions	5.2(8c)	cisco_MDS_91
test-team1	tucson	brocade_fcoe	AssociatedSoftwareHardwareVersions	v6.4.2b	76.7
test-team1	tucson	brocade_fcoe	AssociatedSoftwareHardwareVersions	v7.1.1c	76.7
test-team2	hursley	brocade_fc	AssociatedSoftwareHardwareVersions	v6.3.1b	26.2
test-team2	hursley	brocade_fc	AssociatedSoftwareHardwareVersions	v6.1.0b	42.2
test-team2	hursley	cisco_fc	AssociatedSoftwareHardwareVersions	3.2(2c)	cisco_MDS_91

Figure 10. Switch compare sample summary

A. Interlock Test Coverage

With the various test teams located world-wide, the need for a central list of SAN switch hardware across IBM test has become apparent. The information gathered from the switches is an initial step in allowing IBM systems test groups to more closely interlock and drive test coverage across test labs. The SAN distance matrix project has helped us to identify test labs that are closely aligned and those that provide unique coverage points. While continuity is important and we need to ensure we are covering the most typical SAN field deployments we also realize the need to balance that model with one of broad coverage.

B. Additional benefits from early results

Along with balancing our coverage, the early results provided insight on switch utilization that provided additional benefits to our test teams.

It also allowed us to identify groups utilizing dated switch hardware and place them into a hardware refresh pool to help us get new switches to the teams that may need it the most. It also allowed us to collect information on which test groups were on IBM supported Cisco and Brocade switch code levels. Testing a variety of code levels helps in our testing coverage since customers have a variety of environments and update at different rates.

Another benefit from the results was that we were able to look at switch utilization and stress rates to ensure we are accurately stressing our equipment and in the identified cases where we were not, to put plans in place to help increase load coverage. With this type of review of environment architecture designs we can recommend changes or complexity additions where appropriate and create more customer-like environments.

Figure 11 gives an example of a cross-test-cell review, which was done with one systems test group that consisted of a main test coverage mission spread across five environments at unique site locations.

Each of these test groups were responsible for unique IBM Server and Storage focused system test. This project provided the framework and data to bring the groups together to collectively review, compare and analyze how each group architected, deployed and utilized their SAN switches. The groups benefited from having a better understanding of the broader SAN coverage model. From these reviews, we are able to recommend changes and/or complexity additions to each SAN environment. Additionally, the broader coverage review exercise proved to be useful and was later implemented on a more frequent basis across the labs in this illustrative example.

Name	test-team1	test-team2	test-team3	test-team4	test-team5
AvgFcEPortRate	8	N/A	3	8	4
AvgFcFPortRate	6	4	4	9	8
HighesFctEPortRate	8	N/A	10	8	4
HighestFcFPortRate	8	4	8	16	8
LongestKernelUptime	329	152	195	56	54
ShortestKernelUptime	108	16	108	1	54
PercentageEthPortsUtilized	3	N/A	N/A	N/A	N/A
PercentageFcPortsUtilized	48	23	25	27	12
PercentagePortsUtilized	39	23	25	26	12
PercentageVfcPortsUtilized	50	N/A	N/A	N/A	N/A
PortChannelUsage	no,yes	no	no,yes	yes	no
TotalEthPortsCount	96	N/A	N/A	16	N/A
TotalEthPortsLoggedIn	3	N/A	N/A	N/A	N/A
TotalFcPortsCount	402	96	452	610	48
TotalFcPortsLoggedIn	193	23	113	169	6
TotalPortCount	500	96	452	626	48
TotalPortsLoggedIn	197	23	113	169	6
TotalVfcPortsCount	2	N/A	N/A	N/A	N/A
TotalVfcPortsLoggedIn	1	N/A	N/A	N/A	N/A
TotalSwitchWithDataCount	6	4	5	4	1
TotalVLANsCount	3	N/A	N/A	2	N/A
TotalVSANsCount	44	N/A	4	9	1

Figure 11. Cisco FC Cross-test-cell Results Table

Overall, we were able to systematically collect data from global IBM systems test labs and create a centralized view of SAN switch equipment and coverage across IBM systems test. The scripted process extracted data from the switch dumps was used to build compare logic to define and understand meaningful distances (comparisons) among the groups as well as summarize the charted data to compare trends and coverage analysis over time. We were also able to gather dump data from select customers representing a broad range in company size and industry focus. The comparison of our test lab coverage models with customer environments allows our test teams to continually alter test configurations and architectures to be more customer-like and helping to ensure our testing is continually evolving and relevant. The goal of this project was not to be used as a SAN report card, grading tool, or micromanaging utility, but rather an overall method to look across IBM test groups and understand large scale SAN coverage models and gaps and continual areas for improvement.

VII. ADDITIONAL BENEFITS REALIZED

Since its inception, the distance matrix project has shown added value in many foreseen and unforeseen ways. The largest benefit of this project is the ability to systematically extract and model coverage across a large number of test and client SAN environments enabling increased coverage without expanding resource requirements or timelines. One of the key success factors for this model is its scalability. The scalability and reach of the distance matrix project has also uncovered additional unforeseen benefits. In this section we will introduce and expand briefly on a few of these benefits:

1. Centralized visibility of switch inventory and distribution across IBM test teams
2. Decreased root cause analysis time,
3. Client critical situation recreate advancement opportunities
4. Increased technical interlock across systems test labs

The value of asset knowledge and a centralized view of deployed SAN switches across IBM test labs is a critical success factor to make enlightened decisions considering the infrastructure as a whole. The distance matrix project provided a centralized list of switches and switch characteristics across IBM test labs world-wide. This centralized view allowed vested parties to review deployed assets and increase asset pooling, sharing and roll-off sharing. For example, when a large SAN lab in Tucson, AZ was undergoing a reconfiguration project and upgrading its SAN infrastructure the team was able to make better informed decisions of which teams could benefit from the surplus switches removed from the previous environment.

Decreased root cause analysis time is another side benefit that can be extracted from the distance matrix project. Having the data to understand which switch configurations encountered certain defects provides valuable insight that can lead to decreased root cause analysis time frames. Additionally, the data can be used to extract trending information on SAN topologies and characteristics that most often lead to increased defect discovery.

Another side benefit realized during the course of this project is the ability to utilize the centralized switch and topology data across test labs to select the most appropriate lab and location for customer debug or recreate activities. For example, if a Customer is experiencing an issue in a Brocade SAN environment with XIV storage we can take the Brocade environment specifics including port speeds, code levels and environment complexity and search across IBM test labs for the environment best resembles the customer environment to setup the recreate. Utilizing this method helps cut recreate time by mitigating the time needed for test or support teams to reconfigure an environment to closely resemble the customer environment.

This project also led to increased technical interlock and technical sharing across worldwide systems test labs. Since its inception, the project has been well received across systems test labs and has helped to create an open dialogue and tool for sharing coverage and best practices across the systems test labs world-wide. By forming a review and sharing process across technical test leads and architects the distance matrix project has sparked strong ongoing relationships and dialogue across key technical leaders world-wide. Teams that originally created their designs in a more isolated environment now have extended resources and lab models available to them for review and leverage. In a company as large as IBM, bringing together test leaders across systems test labs and providing an open sharing SAN coverage model for continued technical leverage across world-wide test environments is a critical step in the right direction.

VIII. CONCLUSION AND FURTHER DEVELOPMENT

As solution complexity and the number of supported configurations increase in the IT industry, we must continue to re-invent the ways we do solution testing. In our global test environment, the need to have procedures in place to extract data and create advanced comparison and coverage models is essential.

This project has shown tremendous promise for being able to systematically extract and model coverage across a large number of test and client SAN environments. One of the key factors of this models continuing success is its scalability. The IBM test group started with business requirements and an early operational model vision and worked directly with the IBM Haifa Research lab to expand and translate early visions into a working model that is currently being deployed and leveraged across test labs world-wide.

We are currently working on plans to extend the distance function beyond reducing the data to a single dimension. For example, today one distance function is the difference in the average rates among different groups. We could instead compute a distance metric over the rate vectors. We are also looking into opportunities to expand the areas of coverage, the scope of the environments we are able to capture and working on data optimization and smart analytics to help ensure we continue to provide leading edge test coverage and innovation.

As we continue to implement the distance matrix project across test labs within IBM we are gathering key data and making methodical changes is SAN test architecture to provide better test coverage points for IBM products and solutions.

REFERENCES

[1] Y. Adler, T. Astigarraga, S. Jackson, Jose R. Mosqueda, and O. Raz, "IBM SAN Distance Matrix Project," Proc. VALID, 2014, The Sixth International Conference on Advances in System Testing and Validation Lifecycle (VALID 2014), IARIA October 2014, pp. 84-87, ISSN: 2308-4316, ISBN: 978-1-61208-370-4.

[2] "IBM Basics," ibm.com [Online]. Available from: http://www.ibm.com/ibm/responsibility/basics.shtml. [Accessed: 2015-05-19].

[3] T. Astigarraga, "IBM Test Overview and Best Practices" SoftNet 2012, Available from: http://www.iaria.org/conferences2012/filesVALID12/IBM_Test_Tutorial_VALID2012.pdf, [Accessed: 2015-05-19].

[4] "IBM System Storage Interoperation Center (SSIC)," ibm.com [Online]. Available from: http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss, [Accessed: 2015-05-19].

[5] "Cisco DCNM Overview," cisco.com [Online]. Available from: http://www.cisco.com/c/en/us/td/docs/switches/datacenter/mds9000/sw/5_2/configuration/guides/fund/DCNM-SAN-LAN_5_2/DCNM_Fundamentals/fmfundov.html, [Accessed: 2015-05-19].

[6] "Brocade Network Advisor," brocade.com [Online]. Available from: http://www.brocade.com/products/all/management-software/product-details/network-advisor/index.page, [Accessed: 2015-05-19].

[7] "EMC Storage Performance Monitoring," solarwinds.com [Online], Available from: http://www.solarwinds.com/es/solutions/emc-storage-performance.aspx, [Accessed: 2015-05-19].

[8] "IBM Tivoli Monitoring," ibm.com [Online]. Available from: http://www-03.ibm.com/software/products/en/tivomoni/, [Accessed: 2015-05-19].

[9] Kuhn, D. Richard, Dolores R. Wallace, and Jr AM Gallo. "Software fault interactions and implications for software testing," Software Engineering, IEEE Transactions on 30.6 (2004), pp. 418-421.

[10] Changhai Nie and Hareton Leung, 2011, "A survey of combinatorial testing," ACM Comput. Surv. 43, 2, Article 11 (February 2011).

[11] I. Segall, R. Tzoref-Brill, E. Farchi, "Using Binary Decision Diagrams for Combinatorial Test Design," ACM, 2011, Proc. 20th Intl. Symp. on Software Testing and Analysis (ISSTA'11).

[12] "IBM Functional Coverage Unified Solution (IBM FOCUS)," ibm.com [Online]. Available from: http://researcher.watson.ibm.com/researcher/view_group.php?id=1871, [Accessed: 2015-05-19].

[13] Y. Adler, A. Aradi, Y. Magid, and O. Raz, "IBM Log Analysis Tool (EASER)," unpublished.

[14] "Beating the Tech Titan" [Online], Available from: http://www.investingdaily.com/22245/beating-the-tech-titan/, [Accessed: 2015-05-19].

[15] "Brocade gains SAN market share in 2013, Cisco dips, says Infonetics" [Online], Available from: http://www.infotechlead.com/networking/brocade-gains-san-market-share-2013-cisco-dips-says-infonetics-20880, [Accessed: 2015-05-19].