

LOAD DYNAMIX ENTERPRISE WORKLOAD ITERATOR

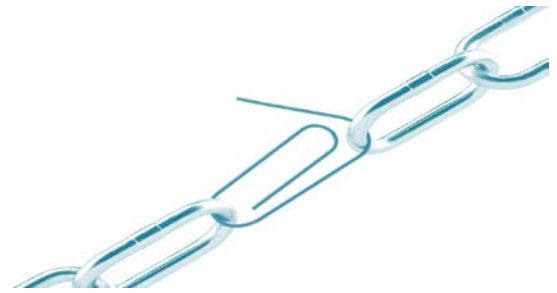
Simplified storage system characterization to find sweet spots and blind spots

It should be of no surprise that production outages are extremely costly, both financially and to company brand image. To minimize storage outages, you need to know how your storage system will perform under worst case scenarios – in advance of production deployment. An ideal approach is to test these scenarios with highly accurate workload models based on real-world production applications.

Load Dynamix empowers you to do this.

But it's not always easy to access production workload data. Our approach includes putting your storage arrays under an extremely diverse range of workload characteristics and analyzing which types of workloads are optimal (sweet spots) and which ones could cripple performance (blind spots). Load Dynamix helps ensure that your storage infrastructure will perform well in environments where future workload patterns are not easy to predict.

Such functionality differentiates Load Dynamix from freeware tools like Iometer and Vdbench. No other storage testing product can identify blind spots, decrease overall testing time or accurately measure the effects of dedupe / compression and metadata on storage performance.



Highlights

- Test storage arrays with workload iterations for comprehensive behavior analysis, including:
 - Breaking points (blind spots)
 - Optimal workloads (sweet spots)
- Simplified testing interface for improved usability and time savings

“Storage architects can actually test their systems with workloads that closely model their real-world applications. With Load Dynamix, architects can easily determine the strengths and weaknesses of any networked storage array.”

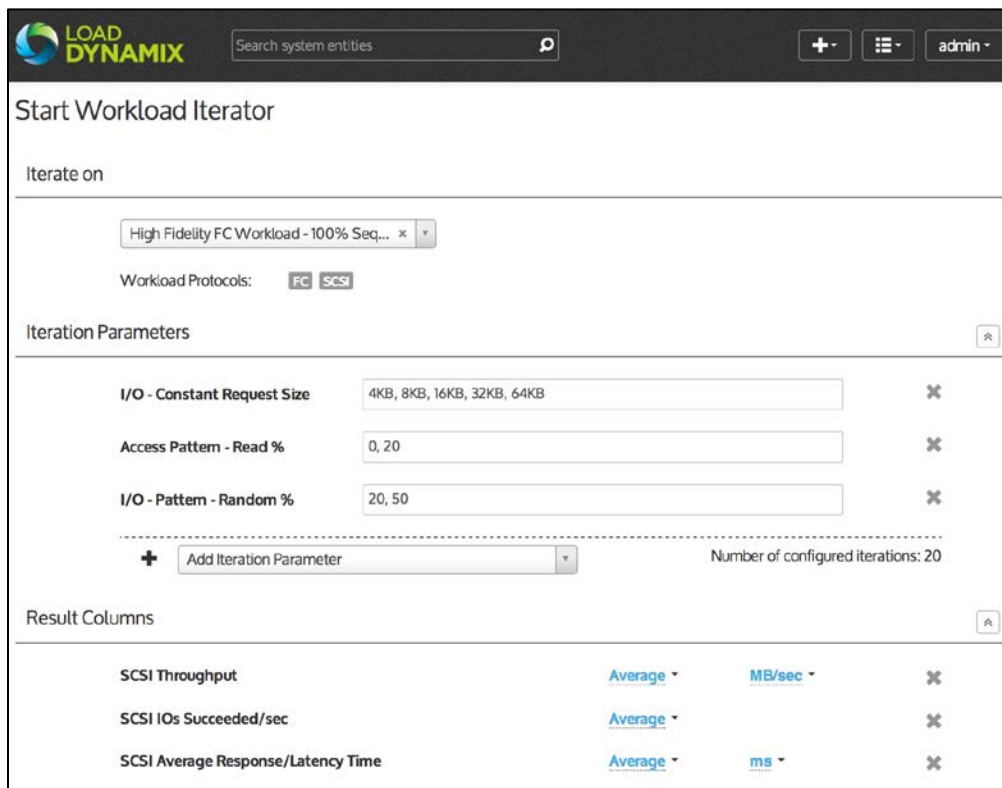
Howard Marks
Chief Scientist
Deep Storage, LLC.

The Workload Iterator

The Workload Iterator is powerful and extremely easy to use feature of Load Dynamix Enterprise. It is built on top of a sophisticated High-Fidelity Networked Storage Workload Model, allowing users to easily automate iterations over hundreds of parameter permutations that define many different characteristics of the workloads. These characteristics include block sizes, read vs. write percentages, sequential vs. random percentages, number of files and folders (NAS), LUN hot spots (SAN), load profiles, deduplication and compression ratios, FC queue depth, and more.

Configuring the Workload Iterator

Each Protocol Workload Model is made up of a rich set of parameters. The majority of these parameters are made available to the Workload Iterator. For each parameter made available, it is selectable as an Iteration Parameter. Iteration Parameters allow the user to define the values that will be used during the execution of the Iteration suite. For example, the user can simply define an Iteration suite that can be permuted over five different Request Sizes, two different read vs. write percentages, and two different sequential vs. random percentages; which result in 20 permutations that will be executed automatically without user involvement.



The screenshot shows the 'Start Workload Iterator' configuration page. At the top, there is a search bar and a user profile 'admin'. The main configuration area is titled 'Start Workload Iterator' and includes the following sections:

- Iterate on:** A dropdown menu set to 'High Fidelity FC Workload - 100% Seq...'. Below it, 'Workload Protocols' are listed as 'FC' and 'SCSI'.
- Iteration Parameters:** A section with three input fields:
 - 'I/O - Constant Request Size' with values '4KB, 8KB, 16KB, 32KB, 64KB'.
 - 'Access Pattern - Read %' with values '0, 20'.
 - 'I/O - Pattern - Random %' with values '20, 50'.
 Below these fields is a '+ Add Iteration Parameter' button and a 'Number of configured iterations: 20' indicator.
- Result Columns:** A table defining the output metrics for the iterations.

Metric	Aggregation	Unit	Action
SCSI Throughput	Average	MB/sec	×
SCSI IOs Succeeded/sec	Average		×
SCSI Average Response/Latency Time	Average	ms	×

Figure 1: Set-up GUI to configure 20 iterations of varying Request Size and read v write percentage and random v sequential percentage.

Running the Workload Iterator

Once the Iteration suite is defined, the Workload Iterator will automatically execute every permutation, and provide the user with a summarized real-time report. Each permutation runs independently, and the permutations are executed sequentially one by one until all permutations are completed.

During run time, a summary table is presented in real-time, allowing the user to monitor the progress and the key performance indicators such as throughput, IOPS and latency for each iteration. In addition, the user has the ability to view a detailed dashboard by clicking on the status of a particular iteration, giving full insight to each permutation.

# ↑	Status	Duration	I/O - Constant Request Size	Access Pattern - Read %	I/O - Pattern - Random %	SCSI Throughput (average)	SCSI IOs Succeeded/sec (average)	SCSI Average Response/Latency Time (average)
1	Finished	01:00	4KB	0	20	315.6 MB/sec	80483.196	1.1 ms
2	Finished	00:59	4KB	0	50	314.7 MB/sec	80250.466	1.2 ms
3	Finished	01:00	4KB	20	20	355.1 MB/sec	90550.401	1.2 ms
4	Finished	01:00	4KB	20	50	355.0 MB/sec	90531.437	1.5 ms
5	Finished	01:00	8KB	0	20	539.4 MB/sec	68913.79	1.2 ms
6	Finished	01:00	8KB	0	50	539.4 MB/sec	68901.995	1.4 ms
7	Finished	00:50	8KB	20	20	610.3 MB/sec	77966.048	1.3 ms
8	Finished	01:00	8KB	20	50	622.1 MB/sec	79471.78	1.7 ms
9	Finished	00:54	16KB	0	20	646.2 MB/sec	41319.51	2.1 ms
10	Finished	00:57	16KB	0	50	645.3 MB/sec	41258.716	2.2 ms
11	Finished	00:52	16KB	20	20	781.3 MB/sec	49953.298	2.1 ms
12	Finished	01:00	16KB	20	50	779.1 MB/sec	49810.164	2.4 ms
13	Finished	00:52	32KB	0	20	703.7 MB/sec	22508.867	3.6 ms
14	Finished	00:52	32KB	0	50	703.9 MB/sec	22513.143	3.7 ms
15	Finished	00:54	32KB	20	20	856.4 MB/sec	27388.315	3.4 ms
16	Finished	00:55	32KB	20	50	858.1 MB/sec	27447.536	4.1 ms
17	Finished	00:54	64KB	0	20	745.0 MB/sec	11920.168	6.3 ms
18	Finished	01:00	64KB	0	50	746.4 MB/sec	11940.8	7.0 ms
19	Finished	00:52	64KB	20	20	918.8 MB/sec	14694.932	6.9 ms
20	Finished	00:56	64KB	20	50	918.0 MB/sec	14684.169	7.7 ms

Figure 2: Summary table showing real-time storage testing results including key performance metrics for each workload iteration.

Post Iteration Analysis

Find storage array blind spots and sweet spots

Insightful reports can be created by Load DynamiX Enterprise to quickly understand which workloads will be potentially crippling for the storage array (blind spots) and which types of workloads the storage array is optimized for (sweet spots). In addition, the test results from all iterations are archived in the Load DynamiX Enterprise database, allowing any user with access rights to browse the results for further detailed analysis.

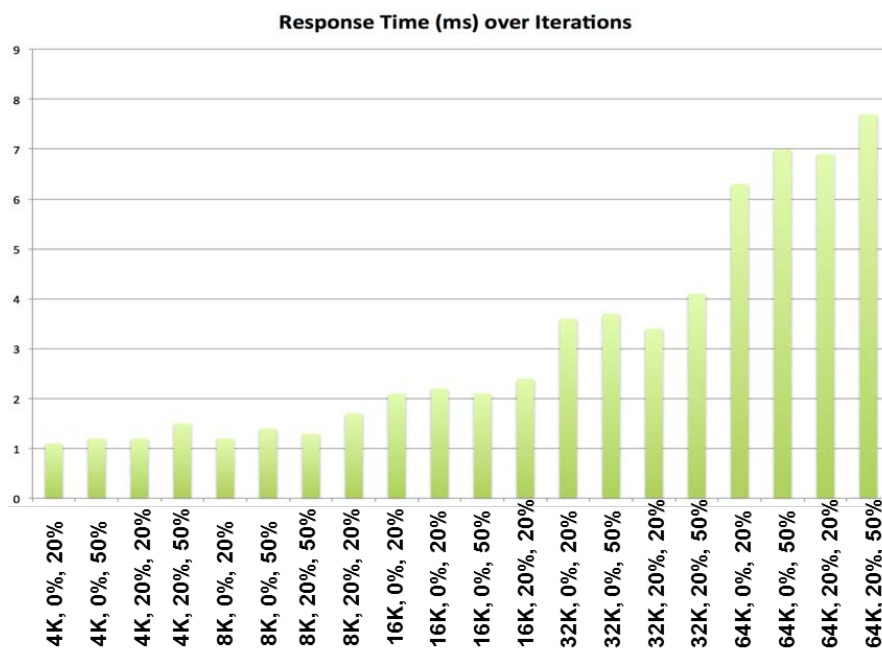


Figure 3: Load DynamiX Enterprise report showing a storage array's performance characteristics under a variety of workload patterns.

Summary

Thorough pre-deployment validation is the only proven approach to ensure the storage arrays can properly support the production workloads at scale while delivering the expected quality of service. In addition to stressing the storage arrays with highly sophisticated models of the known production workloads, it is essential to know ahead of time the sweet spots and the blind spots of the storage arrays by stressing them with a wide range of workload characteristics.

Workload Iterator Feature Specifications

SAN	Fibre Channel, iSCSI
NAS	NFSv3, SMB 2.0
Iteration Parameters	
FC and iSCSI	<ul style="list-style-type: none"> • Access pattern <ul style="list-style-type: none"> - Read vs. Write percentage • LUN Region <ul style="list-style-type: none"> - LUN Region offset, LUN Region size • I/O characteristics <ul style="list-style-type: none"> - Sequential vs. random percentage - Constant and Random I/O Request Size - Number of Asynchronous I/Os • Queue Depth • Dedupe / compression ratios • Load profile <ul style="list-style-type: none"> - Number of concurrent workers - IOPS - Throughput
NFSv3 and SMB 2.0	<ul style="list-style-type: none"> • Access pattern <ul style="list-style-type: none"> - Read vs. Write percentage - Data vs. Metadata percentage • File system <ul style="list-style-type: none"> - Number of files - Number of subfolders per folder - Folders depth - Number of files per folder - Constant and Random file size • I/O characteristics <ul style="list-style-type: none"> - Sequential vs. random percentage - Constant I/O / Random I/O Request Size • Load profile <ul style="list-style-type: none"> - Number of concurrent workers - IOPS - Throughput