Maria Voinea - 4317602
m.voinea@student.tudelft.nl

Paper Review 2

First Chosen Paper

Abel Gordon, Nadav Amit, Nadav Har'El, Muli Ben-Yehuda, Alex Landau, Assaf Schuster, and Dan Tsafrir, "ELI: bare-metal performance for I/O virtualization," ASPLOS 2012, pp. 411-422.

Abstract

The performance of virtualized environments is greatly affected in the case of I/O intensive activity, due to the expensive context switches that take place when handling device interrupts, as both host and guest systems must receive the interrupts and respond to them. ELI (ExitLess Interrupts) comes as a purely software solution to this issue. It enables the host system to run its guest using a modified version of its IDT (Interrupt Descriptor Table) which contains the original handler assignments for the interrupt codes used by the guest's devices, but forces an exception to occur for all other interrupt types, triggering an exit to the host. Thus, the guest handles the interrupts from its assigned devices and the host handles the rest. The host can still forward non-assigned interrupts to its guest by entering a special 'injection mode' which works with the original guest IDT and forced context change on any physical interrupt, as when not using ELI. Furthermore, ELI can take advantage of the new generation of interrupt controller interface (2xAPIC) to allow the guest to also signal completion of interrupts directly to the physical system. The paper shows that these modifications achieve 97%-100% of the performance of a non-virtualized environment, while not compromising the host's security and proper functioning due to dedicated safety mechanisms.

Main Strengths

- The paper is well structured and well written.
- All internal system mechanisms involved are well explained.
- The experiments are detailed and can be reproduced.
- The suppositions made in order to achieve higher performance are verified in experiments.

Main Weaknesses

- The nature of the mechanisms explained makes the paper hard to follow at times.
- The assignment of the same interrupt vectors for host and guest does not account for multiple guests.

- There are no experiments with multiple guests sharing the hardware.

Detailed Comments

- The paper is well organized and all specific internal mechanisms such as interrupt handling, the signaling of interrupt completion, and the approaches for lowering interrupt rates are thoroughly explained.
- The experiments are retraceable and offer relevant measurements and intuitive illustrations on performance. The authors do not just suppose the workload should be I/O intensive to gain a major performance increase, or that 2MB pages should be used on the host, but actually measure at what computation-I/O ratio ELI becomes irrelevant and how interrupt coalescing parameters and page size affect its throughput improvement.
- However, the intensely technical nature of the paper makes it hard to follow at times, especially in the *Security and Isolation* section. This feels like it could be improved with diagrams or a clearer structure of the narrative.
- Also, when assigning the interrupt vectors that the guest wants, the host changes its own entries to avoid duplicates. But the case of another guest using the same interrupt code is not explored. Since the other guest is supposed to be unaware it is not alone, it cannot change its interrupt vectors.
- This supposition of a single guest operating system extends to the experiments. Even though the plural 'guests' is used, there seems to be only one guest running at a time and there is no mention of what would happen in the case of multiple guests.

Overall Evaluation

The paper is well written and extremely relevant for its domain. Only minor revisions are needed. The numeric evaluation is as follows:

- Originality: 4/5
- Technical content: 5/5
- Readability: 4/5
- Overall: 4/5

Suggestion for Improvement

The future work section mentions the intent of enabling 'shadow handlers' which are hidden from the guest to be invoked directly from the shadow IDT. Intuitively, if one can modify the guest IDT, this is exactly what one would modify it to do: run host interrupt handlers without switching context. But the authors do not explain why this is not possible in the original ELI configuration. If the guest cannot have access to host code, or it does not have the privileges to run it, then it will never be possible. But if it does, then it should have been possible even in the original experiments.

Maria Voinea - 4317602
m.voinea@student.tudelft.nl

Second Chosen Paper

Aaron J. Elmore, Sudipto Das, Divyakant Agrawal, and Amr El Abbadi, "Zephyr: live migration in shared nothing databases for elastic cloud platforms," SIGMOD 2011, pp. 301-312.

Abstract

The paper proposes Zephyr as a new method for live migration of a tenant's data in multi-tenant databases spread across multiple servers, in order to achieve load balancing in a live setting. Zephyr aims to support almost full availability of data during the migration process, by establishing a series of operation modes for transferring data pages and their 'ownership' (the right to access them) from the source node to the destination node. Migration starts by switching from Normal to Init Mode, where the database wireframe (schema, data definitions, privileges and the leafless tree index) is transferred. Progression to Dual Mode involves assigning all newly arrived transactions to the destination, while the source is still allowed to finish its ongoing transactions, having exclusive ownership of the pages affected by them. For each new transaction, the destination must pull the data page it wants to access and claim ownership of it before it can process the transaction. Finally, in Finish Mode, all transactions are already completed on the source node and a final transfer of data and ownership of the affected pages to the destination takes place. The authors rigorously prove correctness, fault tolerance, safety and 'liveness' of the migration before conducting a series of experiments to compare Zephyr to the standard stop and copy method of migration in terms of latency and operation failure rate. As expected, Zephyr reduces failures by one to two orders of magnitude and latency by 10-20%.

Main Strengths

- The mechanism of modes and message exchanges is well explained.
- The properties of the migration are rigorously demonstrated.
- The experiments are detailed and can be reproduced.
- The experiments account for different workload structures.

Main Weaknesses

- The paper is difficult to understand without knowledge of detailed internal database organization.
- A specific underlying database organization is assumed.
- There is no experimental comparison with the main rival technique, Iterative State Replication (ISR).

Detailed Comments

- The paper is neatly organized and Zephyr mechanisms as well as its possibility for extension is well explained.
- There is mathematical proof of correctness, fault tolerance, safety and 'liveness' of Zephyr's technique, as well as an analysis of the migration cost and how it compares to existing migration solutions.
- Moreover, the experiments are rigorously documented and several workload structures are explored before drawing conclusions.
- However, the paper is hard to follow due to the fact that the underlying database organization and functionality is not explained. The authors assume pre-existing knowledge of clustered indexing, pagination, serializable isolation, etc.
- Also, the authors assume a page-organized database that uses only clustered indexes, without motivating with a study of its frequency in practice, or how that change in structure would impact the tenant.
- There is an inconsistency in the possible cases of blocked access. For instance, in comparing to ISR there are two types of operations that fail during migration, but later on they become three.
- Zephyr's advantage over the stop and copy method is intuitive. It would have been more resonant to compare to the rival ISR technique in the experiments. In failing to position itself to related work, the study remains more theoretical than proven.

Overall Evaluation

The paper is relevant, but not revolutionary for its domain and there is no performance comparison with its main competitor solution. Revisions are needed. The numeric evaluation is as follows:

- Originality: 3/5
- Technical content: 4/5
- Readability: 3/5
- Overall: 3/5

Suggestion for Improvement

Even if there were no experiments conducted to directly compare the two approaches, the authors could have attempted to recreate the same testing conditions as with the ISR and compare their results to ISR's. It does not suffice to just mention that the goal of Zephyr is to minimize service interruption, since that same goal is mentioned in the rival study.

Maria Voinea - 4317602
m.voinea@student.tudelft.nl

Third Chosen Paper

I. Goiri, K. Le, J. Guitart, J. Torres, and R. Bianchini, "Intelligent Placement of Datacenters for Internet Services," 31st Int' Conference on Distributed Computing Systems (ICDCS), 2011.

Abstract

This paper introduces a tool for placing datacenters to service population centers in a way that minimizes costs while not sacrificing quality of service. The authors first propose a framework to quantify and parameterize all aspects of the construction and operation of a datacenter, such as costs, response times, data consistency, and availability. They then explore solutions for solving the non-linear optimization problem obtained with the above parameters. The first is an over-simplification to a linear programming model (LP0), which yields suboptimal results. By pre-setting the datacenters' locations and sizes they obtain another linear model (LP1) that can be used to calculate costs and serviced population. This offers an opportunity for a brute force approach, as well as a heuristic one, by using LP0, LP1 and brute force. Another approach is that of simulated annealing (SA) that starts from a random acceptable configuration and uses LP1 to explore neighboring configurations that yield better results. The final solution, applies pruning to SA. The authors conduct experiments on real-life data gathered from reports of utility costs, worker wages, average temperatures, resource network distributions etc., on the whole of the US territory. The brute force outcomes are used as reference to assess the quality of each algorithm derived in the previous stage, rendering the optimized SA the winner, as it achieves brute force precision. The last section of the paper illustrates the results obtained by varying the constraints for availability, consistence, latency, emissions and chiller presence.

Main Strengths

1. The construction framework is original and its explanation is thorough.
2. The scope of aspects considered in the model is broad.
3. The extent of the input data gathered for the study is considerable.
4. The rendering of experimental results is intuitive and meaningful.

Main Weaknesses

1. The explanation for the linear programming models is somewhat unclear.
2. Input data is not entirely valid.
3. The values obtained for real-life Power Usage Efficiency (PUE) are not sustained.

Detailed Comments

- The proposed framework for parameterizing the expenses and constraints of datacenter placement is original and very detailed.
- The range of aspects considered as parameters for the model is considerably broad (even consistency latency, datacenter tier type and CO2 emissions).
- Also, a great amount of work is put into gathering the data for a wide range of locations and putting it together so that each section of the locations grid is characterized in terms of consumption, emissions and resource network connection opportunities.
- However, the derivation of the non-linear programming model to a linear one is not well explained. For instance, the SB and PB functions are said to be non-linear but the S and P values are removed instead of these for LP0. At the same time, the actual number of servers at a location is expressed as a sum over all locations, and it is equaled to the maximum number of servers for that location as a simplification. All in all, the final version of the LP0 is unclear and not well motivated.
- Another weak point is the validity of the input data, which, for some locations, does not exist and has to be approximated even more than it already is.
- The values are generally well motivated, but it is not the case of the PUE analysis. There is no reference for the behavior of the cooling system and the values in the graph or that 8% power delivery loss. Moreover, the explanation of the cooling procedure would warrant a two-value PUE function (chiller off/ chiller on), not a linearly increasing one.

Overall Evaluation

The paper is original in its approach of the cost and constraint model for datacenters. Since the input data can be further refined and the constraints extended, the resulting tool seems promising for the field. Revisions are needed. The numeric evaluation is as follows:

- Originality: 5/5
- Technical content: 3/5
- Readability: 4/5
- Overall: 4/5

Suggestion for Improvement

The inclusion of both a general characterization of random parts of the US, as well as the case study of a specific datacenter location seems a bit superfluous. I believe an annexed table with the acquired data would be more useful as an overview of all of the US, and the case study could be used as an example of how to read the table if interested.