Karlsruher Institut für Technologie

# Modeling and Experimental Analysis of Virtualized Storage Performance using IBM System z as Example

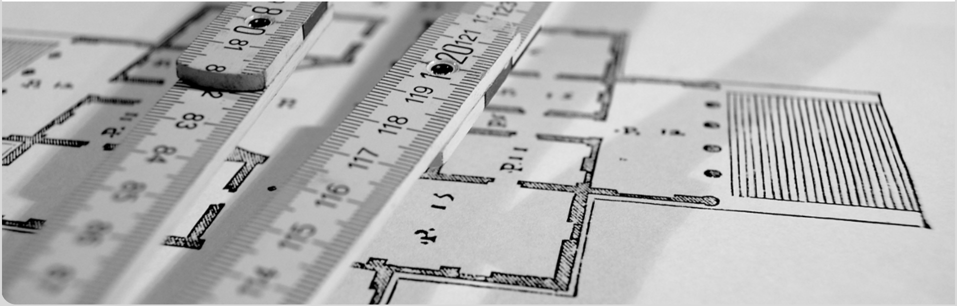
Diploma Thesis Presentation

October 12, 2012

Dominik Bruhn

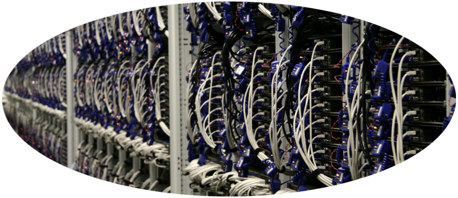
Reviewers: Prof. Dr. Ralf H. Reussner, Prof. Dr. Walter F. Tichy
Advisors: Qais Noorshams, Dr. Samuel Kounev

CHAIR FOR SOFTWARE DESIGN AND QUALITY

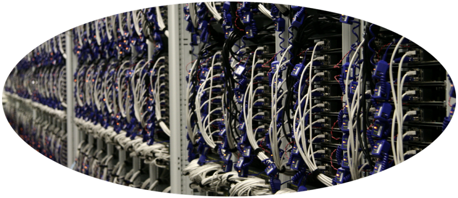


# Motivation

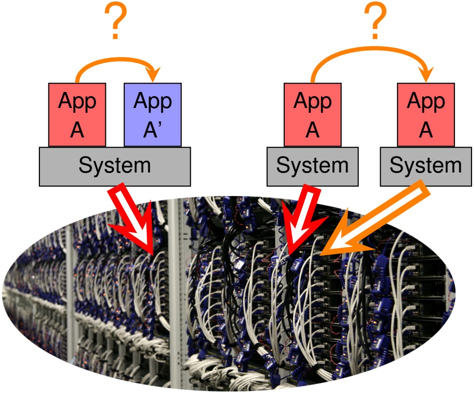
# Motivation



# Motivation



# Motivation




# Problem & Idea & Benefit & Action

## Problem

- Complex systems with many layers
- Difficulty to obtain good performance prediction models

## Idea

Derivation of storage performance models from systematic measurements using regression techniques

## Benefit

- Possibility to predict the performance
- Easier decisions on configurations and systems

## Action

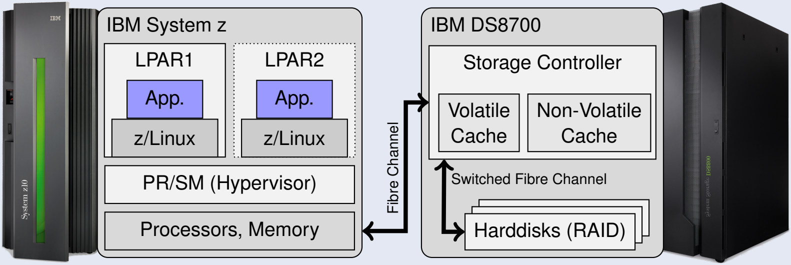
- Creation and evaluation of performance models
- Evaluation of techniques and optimization possibilites

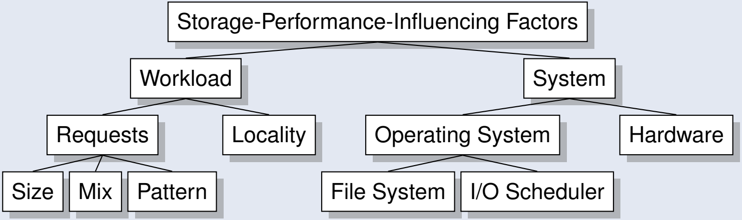
# Contribution

## Contribution

- Creation and evaluation of regression models for storage performance prediction
- Evaluation, analysis and comparison of regression techniques valid for storage performance prediction
- Repeatable process validated in a representative real-world environment

# System Under Study

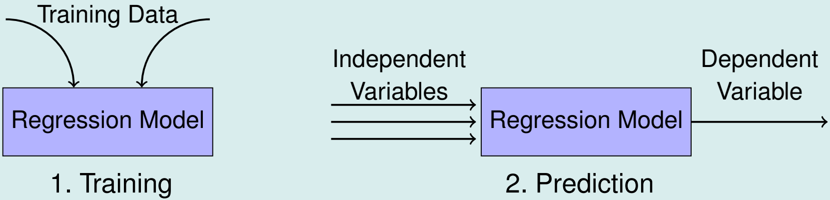



*Derived from Noorshams et al. (2012)*

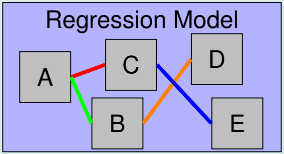
# Modeling

## Regression Models




## Black Box

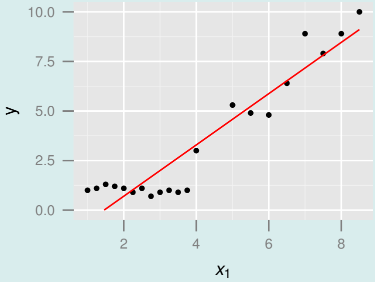

## Model Introspection



# Regression Techniques

KIT Karlsruher Institut für Technologie

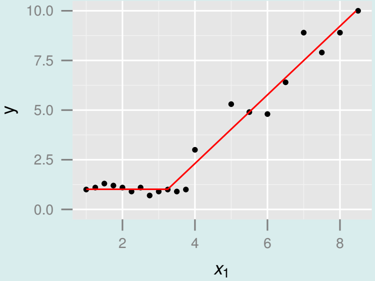
## Linear Regression Models




$$y = -1.884 + 1.293x_1$$

Parameters: *None*

## MARS

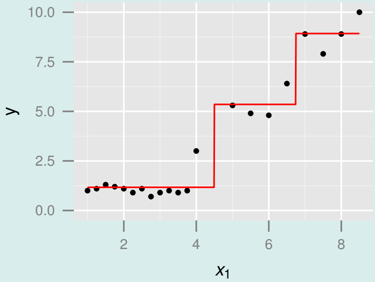


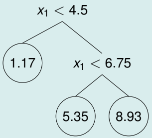

$$y = 1.014501 + 1.72866h(x_1 - 3.25)$$

Parameters: `nk`, `threshold`

# Regression Techniques

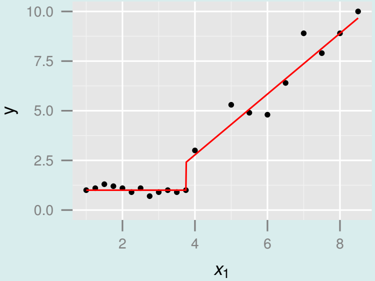
## Regression Trees (CART)

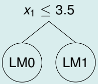



Parameters: `minsplit`, `cp`

## M5

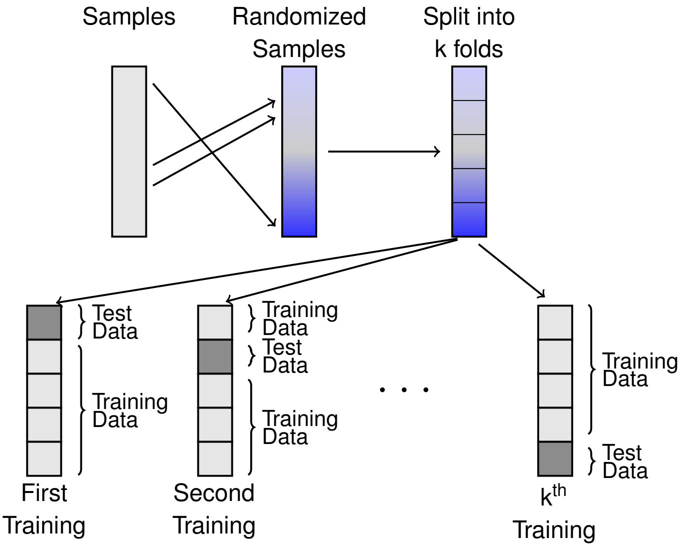



| Model | LM0 | LM1 |
|---|---|---|
| (Intercept) | 1 | -3.34 |
| $x_1$ | | 1.53 |

Parameter: `nsplits`

# Cross-Validation



# Experimental Setup

## Workload

| | |
|---|---|
| *System Parameters* | |
| File system | ext4 |
| I/O scheduler | CFQ, NOOP |
| *Workload Parameters* | |
| Threads | 100 |
| File set size | 1 GB, 25 GB, 50 GB, 75 GB, 100 GB |
| Request size | 4 KB, 8 KB, 12 KB, 16 KB, 20 KB, 24 KB, 28 KB, 32 KB |
| Access pattern | random, sequential |
| Read percentage | 0%, 25%, 30%, 50%, 70%, 75%, 100% |

## Benchmark - FFSB

- Existing benchmark
- At application layer

## System Setup

- Virtual Machines: z/Linux Virtualized by PR/SM in an LPAR
- DS8700 System Storage with 50 GB volatile and 2GB non-volatile cache.

# Approach

## Goal/Question/Metric (GQM)

- **Measurements Analysis**
  - Setup Analysis
    - Stability of the Results
  - Parameter Analysis
    - Parameter Influence
    - Virtualization Influence
- **Performance Modeling**
  - Model Analysis
    - Interpolation
      - Synthetic
      - Random
    - Extrapolation
      - Synthetic
      - Random
    - Reduced Training Sets
    - Nominal Split
  - Technique Analysis
    - Quality Comparison
    - Parameter Tuning
    - Tradeoff Analysis

# Measurement Analysis - Results

## GQM

- **Measurements Analysis**
  - Setup Analysis
    - Stability of the Results
  - Parameter Analysis
    - Parameter Influence
    - Virtualization Influence
- **Performance Modeling**
  - Model Analysis
    - Interpolation
      - Synthetic
      - Random
    - Extrapolation
      - Synthetic
      - Random
    - Reduced Training Sets
    - Nominal Split
  - Technique Analysis
    - Quality Comparison
    - Parameter Tuning
    - Tradeoff Analysis

# Performance Modeling - Results

## GQM

- Measurements Analysis
  - Setup Analysis
    - Stability of the Results
  - Parameter Analysis
    - Parameter Influence
    - Virtualization Influence
- **Performance Modeling**
  - Model Analysis
    - Interpolation
      - Synthetic
      - Random
    - Extrapolation
      - Synthetic
      - Random
    - Reduced Training Sets
    - Nominal Split
  - Technique Analysis
    - Quality Comparison
    - Parameter Tuning
    - Tradeoff Analysis

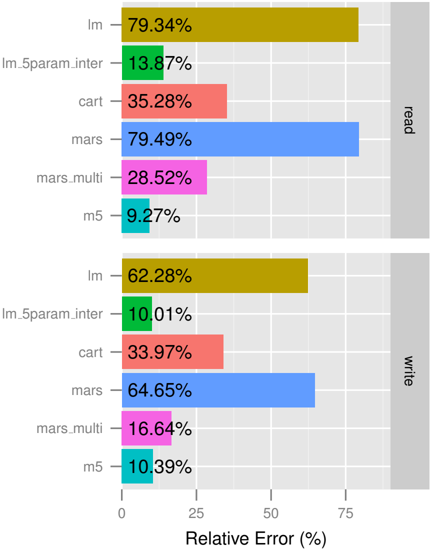
# Interpolation Using Random Samples

*What interpolation abilities do the regression models show when being tested using newly collected samples?*

## Method

- Creation of six regression models:
  - Linear regression model (`lm`)
  - Linear regression model including interaction terms (`lm_5param_inter`)
  - CART tree (`cart`)
  - MARS model without interactions (`mars`)
  - MARS model including all interaction terms (`mars_multi`)
  - M5 model (`m5`)
- Training using all measurements
- Validation using newly collected random samples

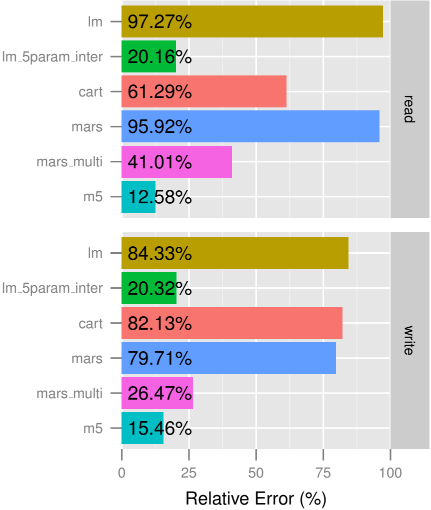
# Interpolation Using Random Samples




- Models without interactions (`lm`, `mars`) do not perform well.
- With an error of ∼10%, M5 works well.
- Linear regression with interactions works surprisingly well.
- CART and MARS (with interactions) rank in the midfield.

# Extrapolation Using Random Samples

*How is the extrapolation ability of the regression models when testing using newly collected data?*

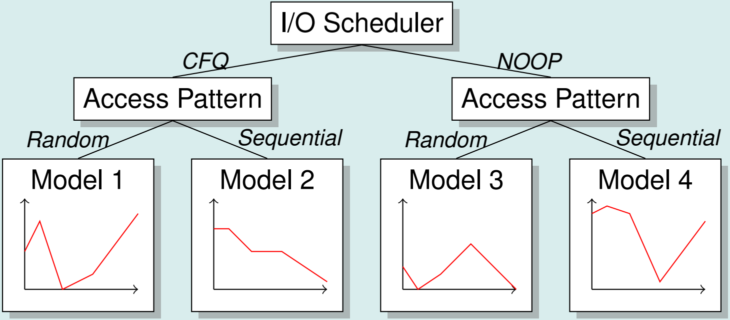



- Again, the models without interactions do not work well.
- CART models can not be used for extrapolation.
- M5 still performs well with an error of ∼14%.

# Nominal Split Model Optimization

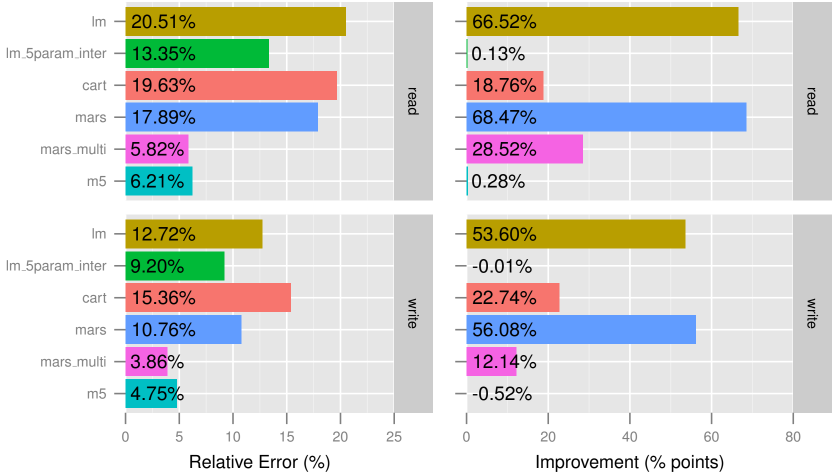
*How can the regression modeling of nominal scale parameters be improved?*

## Method




- Create four models for each value of each nominal parameter.
- Remaining three parameters are all on ordinal scale.

# Nominal Split Model Optimization




- Models without interactions improve the most.
- The best performing models do not benefit.

# Regression Technique Parameter Tuning

*Which configuration parameters of the regression techniques can improve the prediction results?*

## Problem

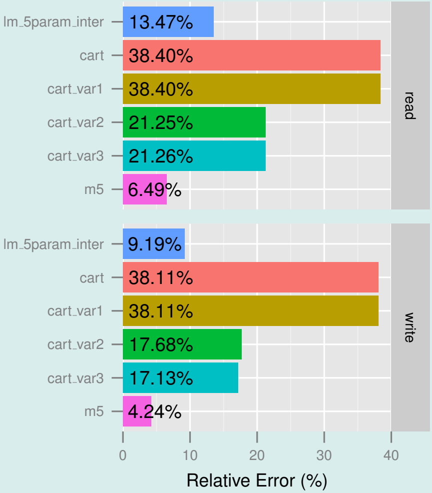
- Different regression techniques have different configuration parameters.
- Default values might not be well-suited.
- It is difficult to find the right parameters.

## Method

- Identify the most promising configuration parameters
- Analyze the influence of these configuration parameters on the prediction quality

# Regression Technique Parameter Tuning

## CART

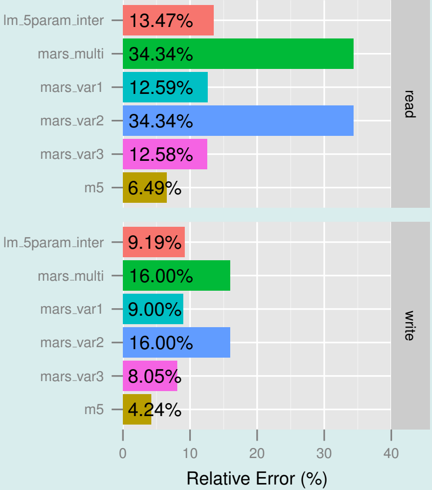



| Model | minsplit | cp |
|---|---|---|
| cart | 20 | 0.01 |
| cart_var1 | 5 | 0.01 |
| cart_var2 | 20 | 0.001 |
| cart_var3 | 5 | 0.001 |

- Decreasing `minsplit` does not improve the models.
- Decreasing `cp` does improve the models by about 50%.

# Regression Technique Parameter Tuning

## MARS




| Model | nk | threshold |
|---|---|---|
| mars_multi | 20 | 0.001 |
| mars_var1 | 40 | 0.001 |
| mars_var2 | 20 | 0.0001 |
| mars_var3 | 40 | 0.0001 |

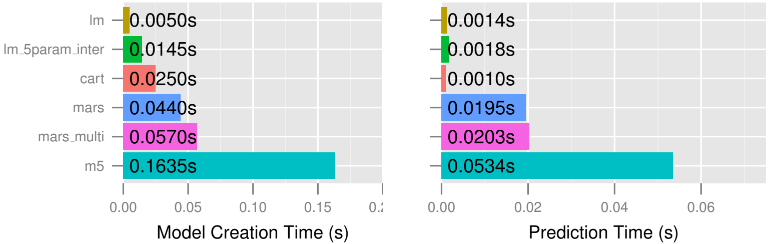
- Increasing `nk` improves the models by about 50%.
- Decreasing `threshold` improves the models only by a small amount.

# Regression Technique Tradeoff Analysis

*What are the advantages and the disadvantages of the modeling techniques?*

| Criterion | Linear Regression | CART | MARS | M5 |
|---|---|---|---|---|
| Prediction Quality | ★★★ | ★ | ★★ | ★★★★ |
| Modeling Time | ★★★★ | ★★★ | ★★ | ★ |
| Prediction Time | ★★★★ | ★★★★ | ★★ | ★ |
| Interpretability | ★★ | ★★★★ | ★ | ★ |

*Stars are only ordered relative ranking.*



# Lessons Learned

## GQM

- Measurements Analysis
  - Setup Analysis
    - Stability of the Results
  - Parameter Analysis
    - Parameter Influence
    - Virtualization Influence
- **Performance Modeling**
  - Model Analysis
    - Interpolation
      - Synthetic
      - Random
    - Extrapolation
      - Synthetic
      - Random
    - Reduced Training Sets
    - Nominal Split
  - Technique Analysis
    - Quality Comparison
    - Parameter Tuning
    - Tradeoff Analysis

# Related Work

## Storage Performance Modeling

Model storage performance using various techniques:

- Predict only the virtualization overhead: *Ahmad et al. (2003)*
- Use fine-grained models: *Kraft et al. (2011), Huber et al. (2010)*
- Omit system parameters *Wang et al. (2004), Anderson (2001), Lee and Katz (1993)*

## Measurement Based Regression Analysis

Use, evaluate and compare regression techniques on other systems:

*Westermann et al. (2012), Courtois and Woodside (2000), Kim et al. (2007)*

# Conclusion

## Summary

- Creation and evaluation of storage performance prediction models using regression techniques
- Evaluation of techniques and optimization possibilities

## Analysis Results

- Extra- and interpolation of storage performance using regression models works well: Errors $\leq 15\%$ possible
- M5 and linear regression models are the best choice in these case.
- Optimization possibilities: Nominal parameters and configuration of the regression techniques.

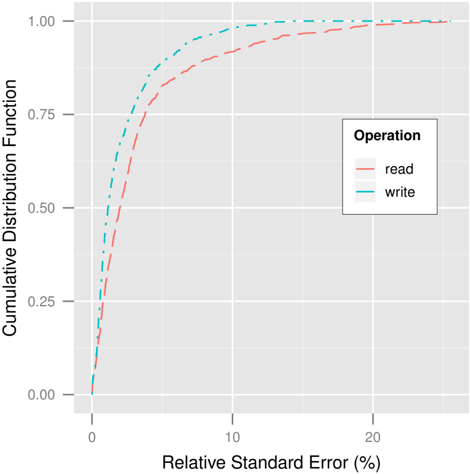
## Outlook

- Further analysis of the optimization possibilities of regression techniques.
- Application and validation using true applications.

# BACKUP

# Stability of the Results

*How reproducible are the experiments results?*

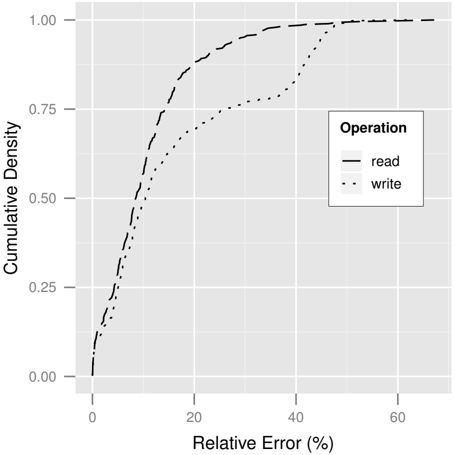



$$\text{rError} = \frac{\sigma_R \cdot 100\%}{\sqrt{5} \cdot \overline{R}}$$

- Read Requests:
  - Mean Standard Error: 3.35%
  - 90th percentile: 8.45%
- Write Requests:
  - Mean Standard Error: 2.10%
  - 90th percentile: 5.35%
- Each measurement run issues up to 2.7M requests.
- Measurements are sufficient repeatable and stable.

# Virtualization Influence

*What is the influence of virtualization?*

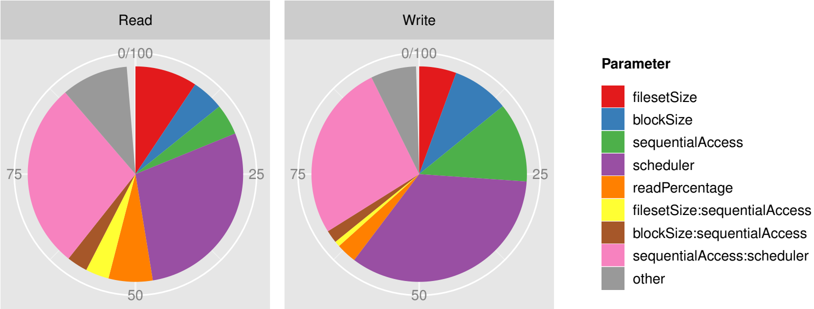



$$\text{rError} = \left| \frac{\frac{\text{dualVM1}+\text{dualVM2}}{2} - \text{singleVM}}{\text{singleVM}} \right|$$

- Mean read requests: 10.60%
- Mean write requests: 16.67%

# Parameter Influence

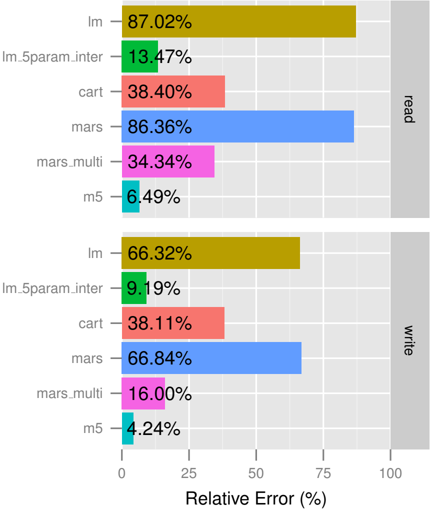
*Which parameters have an influence on the response time?*




- 98.71% (Read) and 99.53% (Write) of the variation can be explained.
- Without interaction terms: Only 54.03% (Read) or 63.36% (Write)
- Interactions terms are necessary.
- All five parameters influence the response time.

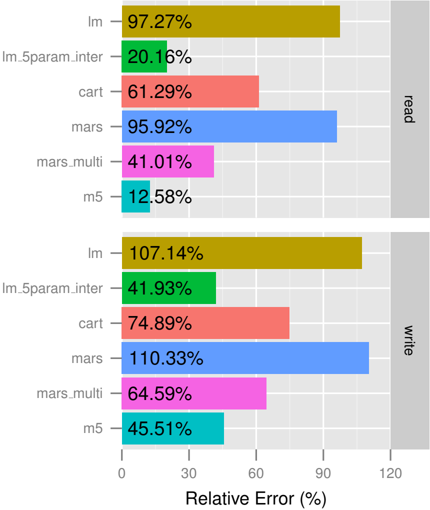
# Interpolation Using Existing Data

*How good is the interpolation of the regression models when using synthetic test sets?*



# Extraplation Using Existing Data

*How good is the interpolation of the regression models when using synthetic test sets?*

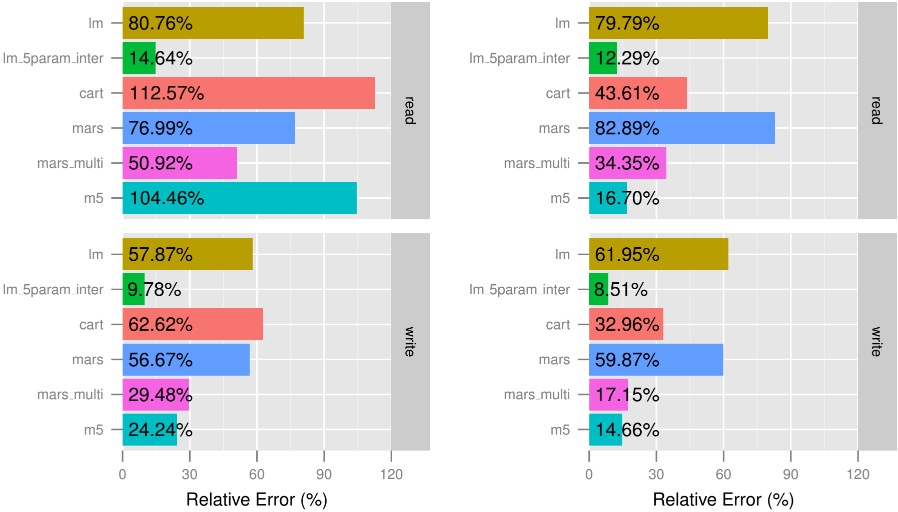


# Reduced Training Sets

*How many measurements are needed for an accurate model?*

| | `red1` | `red2` |
|---|---|---|
| Block size | 4 kB, 32 kB | 4 kB, 16 kB, 32 kB |
| Read percentage | 25%, 75% | 25%, 50%, 75% |
| File set size | 1 GB, 100 GB | 1 GB, 50 GB, 100 GB |
| Access | random, sequential | random, sequential |
| Scheduler | NOOP, CFQ | NOOP, CFQ |
| # of configurations | 32 | 108 |

# Reduced Training Sets




red1 red2

# Parameter Tuning–Paired T-Test

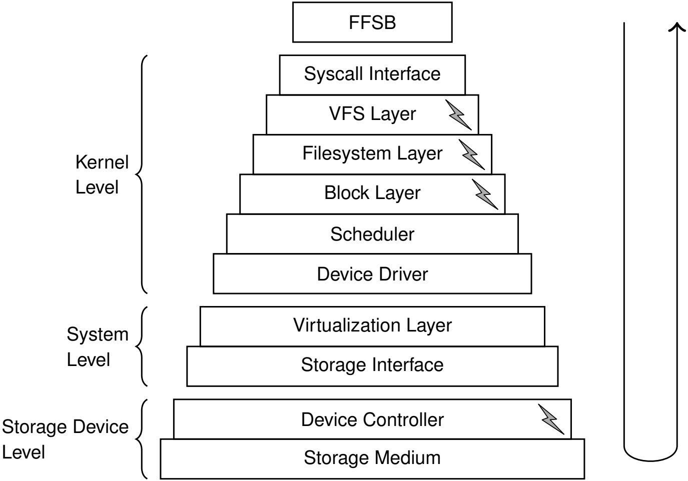
## CART

| Model | minsplit | cp | p (comparing to `cart`) |
|---|---|---|---|
| `cart` | 20 | 0.01 | |
| `cart_var1` | 5 | 0.01 | *identical* |
| `cart_var2` | 20 | 0.001 | 0.014 |
| `cart_var3` | 5 | 0.001 | 0.015 |

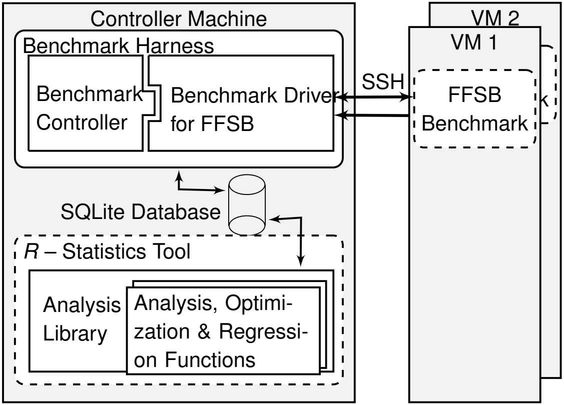
## MARS

| Model | nk | threshold | p (comparing to `mars_multi`) |
|---|---|---|---|
| `mars_multi` | 20 | 0.001 | |
| `mars_var1` | 40 | 0.001 | 0.04 |
| `mars_var2` | 20 | 0.0001 | *identical* |
| `mars_var3` | 40 | 0.0001 | 0.04 |

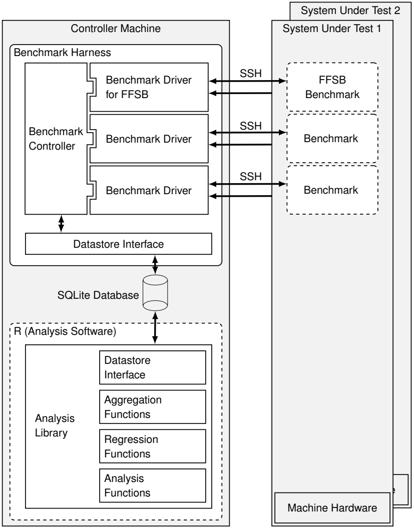
# System Layers



# Benchmarking Setup (Simplified)



# Benchmarking Setup



# Related Measurement Approaches

- Differences from all three tools:
  - No automated analysis of the benchmark results
  - No automated model generation
  - No automated analysis of the regression models
  - No integration for storage benchmark
- Differences from *Software Performance Cockpit*:
  - No simoultaneaous execution of benchmarks on multiple hosts
  - Relies on RMI for the transport
- Differences from *Ginpex*:
  - Missing integration of external benchmarks
  - No regression technique integration
- Differences from *Faban*:
  - No specification of multiple jobs to be run.
  - No analysis possibilities

# References I

I. Ahmad, J.M. Anderson, A.M. Holler, R. Kambo, and V. Makhija. An analysis of disk performance in VMware ESX Server virtual machines. In *Workload Characterization, 2003. WWC-6. 2003 IEEE International Workshop on*, pages 65–76. IEEE, 2003. doi: 10.1109/WWC.2003.1249058.

Eric Anderson. Simple table-based modeling of storage devices. Ssp technical report, HP Laboratories, July 2001.

Marc Courtois and Murray Woodside. Using regression splines for software performance analysis. In *Proceedings of the 2nd international workshop on Software and performance*, WOSP '00, pages 105–114, New York, NY, USA, 2000. ACM. ISBN 1-58113-195-X. doi: 10.1145/350391.350416.

Nikolaus Huber, Steffen Becker, Christoph Rathfelder, Jochen Schweflinghaus, and Ralf H. Reussner. Performance modeling in industry: A case study on storage virtualization. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 2*, ICSE '10, pages 1–10, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-719-6. doi: http://doi.acm.org/10.1145/1810295.1810297.

# References II


Hyunjoong Kim, Wei-yin Loh, Yu-shan Shih, and Probal Chaudhuri. Visualizable and interpretable regression models with good prediction power. *IIE Transactions*, 39: 565–579, 2007.

Stephan Kraft, Giuliano Casale, Diwakar Krishnamurthy, Des Greer, and Peter Kilpatrick. IO performance prediction in consolidated virtualized environments. In *Proceedings of the second joint WOSP/SIPEW international conference on Performance engineering*, ICPE '11, pages 295–306, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0519-8. doi: 10.1145/1958746.1958789.

Edward K. Lee and Randy H. Katz. An analytic performance model of disk arrays. *SIGMETRICS Perform. Eval. Rev.*, 21(1):98–109, June 1993. ISSN 0163-5999. doi: 10.1145/166962.166994.

Qais Noorshams, Samuel Kounev, and Ralf Reussner. Experimental evaluation of the performance-influencing factors of virtualized storage systems. In *Proceedings of the 9th European Performance Engineering Workshop (EPEW'12), Munich, Germany, July 30-31*, 2012.

# References III

Mengzhi Wang, Kinman Au, Anastassia Ailamaki, Anthony Brockwell, Christos Faloutsos, and Gregory R. Ganger. Storage device performance prediction with CART models. In *Proceedings of the The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*, pages 588–595, Washington, DC, USA, 2004. IEEE Computer Society. ISBN 0-7695-2251-3.

Dennis Westermann, Jens Happe, Rouven Krebs, and Roozbeh Farahbod. Automated inference of goal-oriented performance prediction functions. In *27th IEEE/ACM International Conference On Automated Software Engineering (ASE 2012)*, September 2012.