Doctoral Project Presentation

DEDS Summer School 2022

July 8, 2022

Abduvoris Abduvakhobov

Center for Data Intensive Systems, Daisy
Department of Computer Science
Aalborg University
Denmark

AALBORG UNIVERSITY
DENMARK

Agenda

Introduction

PhD Topic: ESR:2.3. Model-based storage for time series

Supervisors:

- Associate Professor Christian Thomsen, AAU
- Professor Esteban Zimanyi, ULB

Secondment: Siemens Gamesa Renewable Energy

PhD Start Date: February 1, 2022

Current progress:

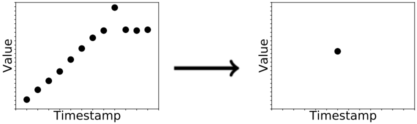
- Started collaboration with secondment partner and got access to data
- Started working on evaluation tool for ModelarDB
- Parental leave from June 1, 2022 to September 29, 2022

Motivation

Research undertaken by Jensen et al. 2021

Meetings with manufacturers, owners, and energy traders:

- Modern turbines are monitored by up to 7,000 high-quality sensors
- The storage needed makes storing high-frequency sensor data infeasible
- Simple aggregates (e.g. 10-minute averages) are stored instead of the high-frequent series, thereby removing useful fluctuations and outliers:



- Users believe problems can be found earlier with high-frequency data.
- Compression need only be lossless for some types of time series.

Motivation
Data

Meetings with manufacturers, owners, and energy traders:

- The sensors are installed with wired power and connectivity.
- Each sensor produces a data stream sampled to, e.g., a 10 Hz series.
- Collected measures include: Air Pressure, Humidity, Voltage, Power, Rotation Speed, Temperature, Wind Direction, Wind Speed, Internal Controller Measurements.
- The time series are regular, cleaned, but gaps without values can occur.
- Metadata for each time series must also be stored, e.g., as dimensions.

Motivation

Paramount Properties

Paramount properties for a system managing wind turbine data:

Distribution: The system must be able to scale to many nodes.

Stream Processing: Data points are arriving continuously as a regular time series and must be queryable with a short latency.

Compression: High compression is needed for high-frequency data.

Efficient Retrieval: Indexes or ordered storage for fast retrieval.

Approximate Query Processing: Approximate answers can be accepted for some time series and enables use of lossy compression.

Extensibility: Allows users with domain knowledge to implement new storage methods optimized specifically for their data sets.

State of the Art
Time Series Management Systems

Time Series Management Systems or TSDBs:

- Store time series that consist of time stamp and a value or a set of values.
- Optionally contain metadata or tags.
- Process queries on time series.
- Queries contain timestamp or a time range.

Categorization based on architecture (Jensen et al. 2017):

- Internal Data Stores:
  - Mostly centralized.
  - Tightly coupled data storage to processing component.
  - Few mature implementations.
  - Examples: Plato, LittleTable, VergeDB, Chronos, Apache IoTDB

State of the Art
Time Series Management Systems

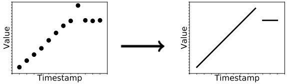
Categorization based on architecture:

- External Data Stores:
  - Predominantly distributed.
  - New processing engine on top of external data store.
  - Most number of mature implementations.
  - Examples: Apache Druid, Bolt, Gorilla, BTrDB and ModelarDB
- Extension for RDMS:
  - Extends popular RDBMS.
  - predominantly centralized.
  - Small number of mature implementations.
  - Examples: Chronix, EdgeDB, and Heracles

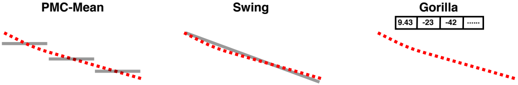
State of the Art
ModelarDB

ModelarDB

- Individual time series can be described with models:
- E.g., $v = a * t + b$ can represent a sub-sequence using only a and b.



- Uses Apache Cassandra for storage and Apache Spark for query processing.
- Approximates the time series values using mathematical functions (models) and stores only model coefficients.
- Currently includes three different model types:

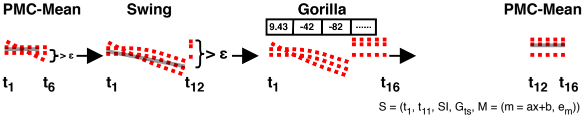


Jensen et al. 2021

ModelarDB

Correlated Time Series

- A data set often contains redundant information across time series:
  - E.g, co-located temperature sensors often produce similar values.
- ModelarDB can group correlated time series together and compress them as one stream of models to reduce the storage required.
- A list of model types fit models to data points, e.g., a constant (PMC-Mean), linear (Swing), and lossless (Gorilla) model type:



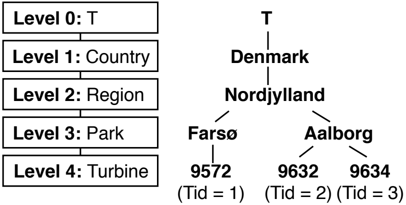
(Jensen et al. 2021)

ModelarDB
Multi-model Group Compression

- Time series are grouped based on user hints given using primitives.
- The primitives can be combined and allow users to state that series are correlated based on their source or their dimensional hierarchy.
- Users can use their domain knowledge, analyze historical data, or use ModelarDB's automatic grouping method built on the primitives.

Grouping 9632 and 9634:

- From specific sources: 9632 and 9634
- Sharing a specific member: Location 3 Aalborg
- Share members until a level: Location 3
- The dimension's distance: 0.25
- Automatically (distance): auto



(Jensen et al. 2021)

ModelarDB
Open Issues

Open Issues

- No integrated functionality for evaluating the efficiency and use of model types;
- Model types are quite generic;
- Sampling interval and error bound could be changed dynamically;
- System only supports automatic and manual grouping using heuristics (domain knowledge and metadata);
- Time series must be ordered and have a regular sampling interval;

Project Objectives

Research Questions

RQ1: How can we efficiently evaluate the compression performance of model types and the quality of compression to varying error bounds of ModelarDB on different datasets?

RQ2: Depending on the the outcomes of RQ1, what other model types can be implemented to improve the compression and query performance of ModelarDB on real-life RES datasets?

RQ3: How can time series automatically be grouped using different correlation statistics and provided heuristics during the ingestion process?

RQ4: How can model-based ingestion of time series with a dynamic sampling interval and error bound be supported in ModelarDB?

Work and Publication Plan

Work and Time Plan

Time	Plan
Spring' 22 (Home)	Literature study Problem formulation Preparation of the Doctoral Project Plan Begin work on Paper 1 Begin development of ModelarDB performance evaluation tool Data collection Start general and project-related courses Trying out ModelarDB on real-life datasets and analyzing the results Parental leave from June 1 to September 29
Milestones	**Submission of 2-month Doctoral Project Plan** **Establish collaboration with secondment partner and get access to data**
Fall' 22 (Home)	Continue project-related courses Develop and test performance evaluation tool for ModelarDB
Milestones	**Submission of Paper 1** **Submission of 11-month Doctoral Project Plan** **Secondment completed**

Work and Publication Plan

Work and Time Plan

Time	Plan
Spring' 23 (Host)	Develop new model types for ModelarDB Begin work on Paper 2 Refine and test new model types with real-life datasets
Milestones	**Submission of Paper 2**
Fall' 23 (Host)	Develop new method for correlation-based grouping in ModelarDB Begin work on Paper 3
Milestones	**Submission of Paper 3**
Spring' 24 (Home)	Develop and test new method for compressing dynamic sampling intervals and model error bounds Begin work on Paper 4
Milestones	**Submission of Paper 4** **Completed all general and project-related courses**
Fall' 24 (Home)	Writing the Thesis
Milestones	**Submission of PhD Thesis**

Work and Publication Plan

Tentative Publication List

Tentative Title of Paper 1: A tool for analysis of the efficiency of model-based compression in ModelarDB
Type: Conference paper.
Description:

- Analysis of the efficiency of current model types deployed by ModelarDB.
- Integrated tool that explains the system performance and its usage of model types.
- Performance indicators and visualization.

Datasets: Siemens Gamesa Renewable Energy and ENGIE data.
Authors: A. Abduvakhobov, S.K. Jensen, C. Thomsen, T. B. Pedersen, E. Zimányi, T. Pasma.
Length: 12 pages.
Time of submission: September, 2022 (alternatively March, 2023).
Outlet: IEEE BigData (alternatively DOLAP).

Work and Publication Plan

Tentative Publication List

Tentative Title of Paper 2: New model types to achieve better compression rate and lower error bound for ModelarDB.
Type: Conference paper.
Description:

- Develops new model types for better ingestion and storage use.
- Mainly tailored to match real-life use cases.
- Novel time series compression method.

Datasets: Siemens Gamesa Renewable Energy and ENGIE data.
Authors: A. Abduvakhobov, S.K. Jensen, C. Thomsen, T. B. Pedersen, E. Zimányi, T. Pasma.
Length: 12 pages.
Time of submission: June, 2023.

Outlet: EDBT.

Work and Publication Plan

Tentative Publication List

Tentative Title of Paper 3: Automatic grouping of time series by deploying correlation statistics in ModelarDB
Type: Journal paper.
Description:

- More optimized and faster grouping (possibly in a streaming fashion).
- Leverages correlation and other statistical attributes of time series.
- Supported by user heuristics and metadata.

Datasets: Siemens Gamesa Renewable Energy and ENGIE data.
Authors: A. Abduvakhobov, S.K. Jensen, C. Thomsen, T. B. Pedersen, E. Zimányi, T. Pasma.
Length: 12 pages.
Time of submission: March 2024.

Outlet: PVLDB.

Work and Publication Plan

Tentative Publication List

Tentative Title of Paper 4: Adding dynamic sampling intervals and error bounds for time series ingestion of ModelarDB.
Type: Conference paper.
Description:

- Dynamic error bound and sampling interval.
- User controls the error bound and sampling interval.
- More fine-grained data for exceptional cases.
- Also discusses automatic adjustment of error bound.

Datasets: Siemens Gamesa Renewable Energy and ENGIE data.
Authors: A. Abduvakhobov, S.K. Jensen, C. Thomsen, T. B. Pedersen, E. Zimányi, T. Pasma.
Length: 12 pages.
Time of submission: November, 2024.

Outlet: ICDE.

Plan for PhD Courses

Course Name	**At**	**Type**	**ECTS**	**Time**	**Status**
General Courses	AAU	General	13.75	’22-’25	Planned
Project-related Courses	AAU	General	17	’22-’25	Planned
Winter School (ARC)	ARC	General	3	Spring’22	Finished
Summer School (ULB)	ULB	Project	3	Summer’22	Mandatory
Winter School (AAU)	AAU	General	3	Winter’22	Mandatory
Summer School (UPC)	UPC	Project	3	Summer’23	Mandatory
Conference Attendance	TBD	Project	3	TBD	Planned
Danish Lessons	TBD	General	TBD	TBD	Mandatory
Total: 30.75					

Secondment

Partner Organization: Siemens Gamesa Renewable Energy (SGRE)
Secondment Supervisor: Tjip Pasma
Secondment start date: March 21, 2022
Data: power, voltage, reactive power, controller measurements, and other metadata
Size: ≈110TB

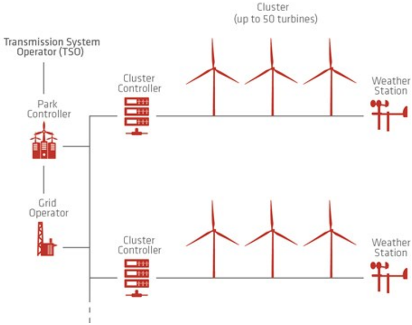
Out[13]:

	TimeStamp	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	...	[illegible]
0	2021-10-01 16:00:00.123	1	200.0	173.825897	165.654297	165.905594	165.905380	173.295059	-9.18	200.0	...	
1	2021-10-01 16:00:00.273	1	200.0	173.824905	165.663910	165.903992	165.905212	173.291351	-9.18	200.0	...	
2	2021-10-01 16:00:00.423	1	200.0	173.825897	165.673157	165.902466	165.905060	173.287689	-9.18	200.0	...	
3	2021-10-01 16:00:00.570	1	200.0	173.821899	165.680008	165.900955	165.904907	173.284042	-9.18	200.0	...	
4	2021-10-01 16:00:00.723	1	200.0	173.820908	165.723297	165.900543	165.904861	173.280426	-9.18	200.0	...	
5	2021-10-01 16:00:00.870	1	200.0	173.820908	165.806335	165.901611	165.904968	173.276825	-9.18	200.0	...	
6	2021-10-01 16:00:01.023	1	200.0	173.824905	165.848724	165.902130	165.905014	173.273270	-9.18	200.0	...	
7	2021-10-01 16:00:01.173	1	200.0	173.825897	165.911987	165.903549	165.905151	173.269730	-9.18	200.0	...	
8	2021-10-01 16:00:01.323	1	200.0	173.821899	165.975845	165.905457	165.905350	173.266220	-9.18	200.0	...	
9	2021-10-01 16:00:01.470	1	200.0	173.825897	166.084717	165.909119	165.905716	173.262741	-9.18	200.0	...	

10 rows × 50 columns

SGRE Data

Parks: ≈800
Total Size of Data: ≈110TB
Storage: Azure blob storage, 2020-2022
Turbine Data: 90 %
Controller Data: 10 %
Sampling interval: 150 ms



SGRE Data

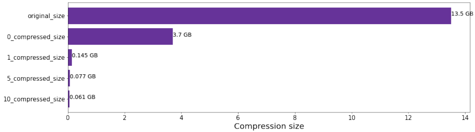
Use Cases

- Exploratory data analysis
- Rule-based algorithms for predictive maintenance
- Measuring time series data with functions/measurements to control to increase or decrease in the power
- Preserving anomalies and peaks is very important
- 1 week, 1 day, 4 hour, 6 hour and 12 hour aggregates
- Some additional custom functions or KPIs might be added by owners

Controller Data

- Comes from 47-50 turbines
- Always multivariate time series, ≈50 columns
- For one year single controller produces around 35 GB of data
- 3 decimal point precision rate is the minimum desired i.e: 0 % error bound

ModelarDB on SGRE Data

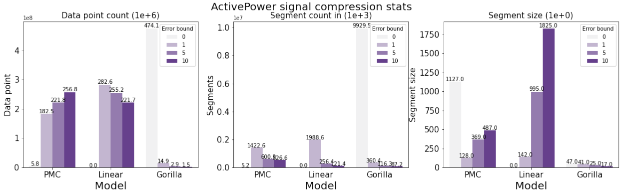


	ActivePower	ActivePower60	PowerError
count	4.804198e+08	4.804198e+08	4.804198e+08
mean	8.022625e+01	8.022626e+01	1.157672e+02
std	6.836718e+01	6.836195e+01	6.988551e+01
min	-1.278090e+01	-3.676155e+00	-1.217273e+02
25%	1.187640e+01	1.187571e+01	3.693950e+01
50%	6.334000e+01	6.335620e+01	1.302957e+02
75%	1.616622e+02	1.616549e+02	1.858317e+02
max	1.785946e+02	1.767819e+02	2.049355e+02

- Original data comprises 3 years of data from PowerLog controller
- Three signals (columns) were chosen: ActivePower, ActivePower60, Powererror
- Each signal makes 4,5 GB of parquet file

ModelarDB on SGRE Data

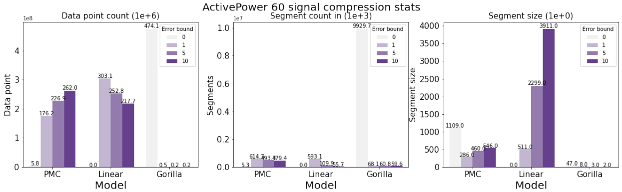
Compression of ActivePower Signal



- Segment size refers to the average segment size "Ingested Data Points / Segments"
- Signal with high variance requires mostly lossless compression for 0 percent error bound.

ModelarDB on SGRE Data

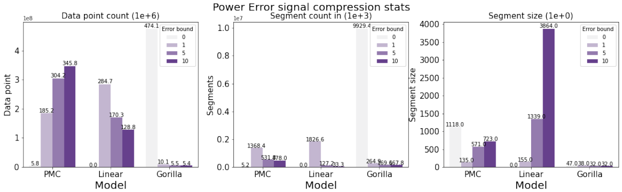
Compression of ActivePower60 Signal



- Linear model comes into play only starting from 1 percent error bound
- Very small chunks of data with 0 error bound was compressed by PMC

ModelarDB on SGRE Data

Compression of PowerError Signal



- PMC mean model remains important to compress long sequences of constant data even with 0 error bound
- Lossless compression is the major choice with 0 error bound in all cases

Time Drifts in SGRE data

- Turbine has a life cycle for 25-30 years
- Failure in NTP server, GPS server or other components causes drifts in sampling interval
- On average 1-5 seconds of drifts are possible per day
- Errors may go up to several hours by the end of the year
- Happens with ≈5% of parks

Out[9]:

		TimeStamp	AvailablePower	RawPower
	148033228	2020-09-15 13:47:37.427	0.143263	-4.310272
146 ms	**148033229**	2020-09-15 13:47:37.573	0.142192	-4.319488
154 ms	**148033230**	2020-09-15 13:47:37.727	0.141130	-4.326656
180 ms	**148033231**	2020-09-15 13:47:37.907	0.140075	-4.313856
	148033232	2020-09-15 13:47:38.057	0.139029	-4.340736
	148033233	2020-09-15 13:47:38.207	0.137990	-4.324608
146 ms	**148033234**	2020-09-15 13:47:38.353	0.382000	-4.346112
164 ms	**148033235**	2020-09-15 13:47:38.517	0.382000	-4.438016

Initial Analysis Results on SGRE Data

- It should be investigated if the new model type could further improve compression rate and query processing at 0 percent error bound
- New compression method is needed to compress irregular time series with time drifts
- Error bounds can vary depending on signal priority or method of generation
- User defined analytics metadata e.g. clusters, outliers, other aggregate statistics could be stored depending on user requirements

Bibliography

[1] Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen, "Time Series Management Systems: A Survey", *TKDE, 29(11)*, 2017.

[2] Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen, "ModelarDB: Modular Model-Based Time Series Management with Spark and Cassandra", *PVLDB, 11(11)*, 2018.

[3] Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen, "Demonstration of ModelarDB: Model-Based Management of Dimensional Time Series", in *SIGMOD*, 2019.

[4] Søren Kejser Jensen "Model-Based Time Series Management at Scale", *PhD Thesis*, 2019.

[5] Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen, "Scalable Model-Based Management of Correlated Dimensional Time Series in ModelarDB$_+$", in *ICDE*, 2021.

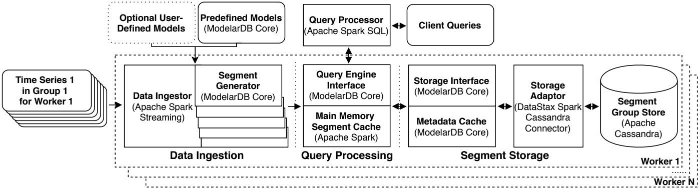
Thank you!

Abduvoris Abduvakhobov
abduvorisa@cs.aau.dk

AALBORG UNIVERSITY
DENMARK

ModelarDB Architecture

- ModelarDB is a portable Java library (ModelarDB Core) interfaced with a query engine (Apache Spark) and storage (Apache Cassandra).



- The architecture of a worker consists of three sets of components: Data Ingestion, Query Processing, and Segment Storage.

Reusing slides by Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen

ModelarDB Segment Structure

Segment

Gid (PK)	StartTime (PK)	Gaps (PK)	EndTime	Mid	Parameters
1	1460442200000	[]	1460442620000	1	0x3f50cfc0
3	1460642900000	[2]	1460645060000	2	0x3f1e ...
...	...	...	...	...	...

Model

Mid (PK)	Classpath
1	PMC-Mean
2	Swing
3	Gorilla

Time Series

Tid (PK)	Gid	Scaling	SI	Country	Region	Park	Entity	Level 1	Level 2	Level 3
1	1	1.0	60000	Denmark	Nordjylland	Farsø	9572			
2	3	1.0	30000	Denmark	Nordjylland	Aalborg	9632			
3	3	4.75	30000	Denmark	Nordjylland	Aalborg	9634			
...	...	...	...	...	...	...	...			

Location Dimension — 2nd Dimension ...

- Time Series and Model store metadata for time series and model types.
- Segment stores sub-sequences of time series as segments with a model.

Reusing slides by Søren Kejser Jensen, Torben Bach Pedersen, Christian Thomsen