Database Technology Group

TECHNISCHE UNIVERSITÄT DRESDEN

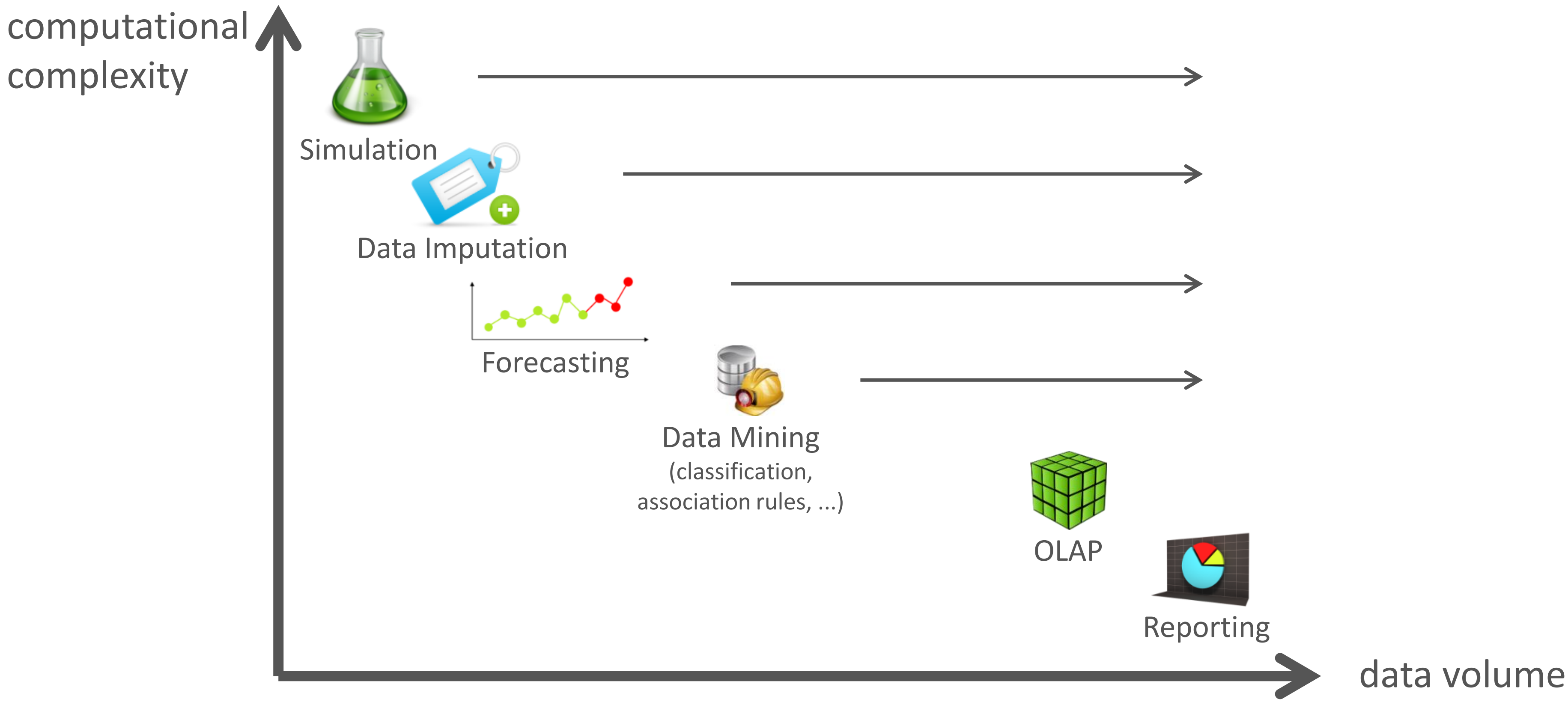
Wolfgang Lehner

Forecasting and Data Imputation Strategies in Database Systems

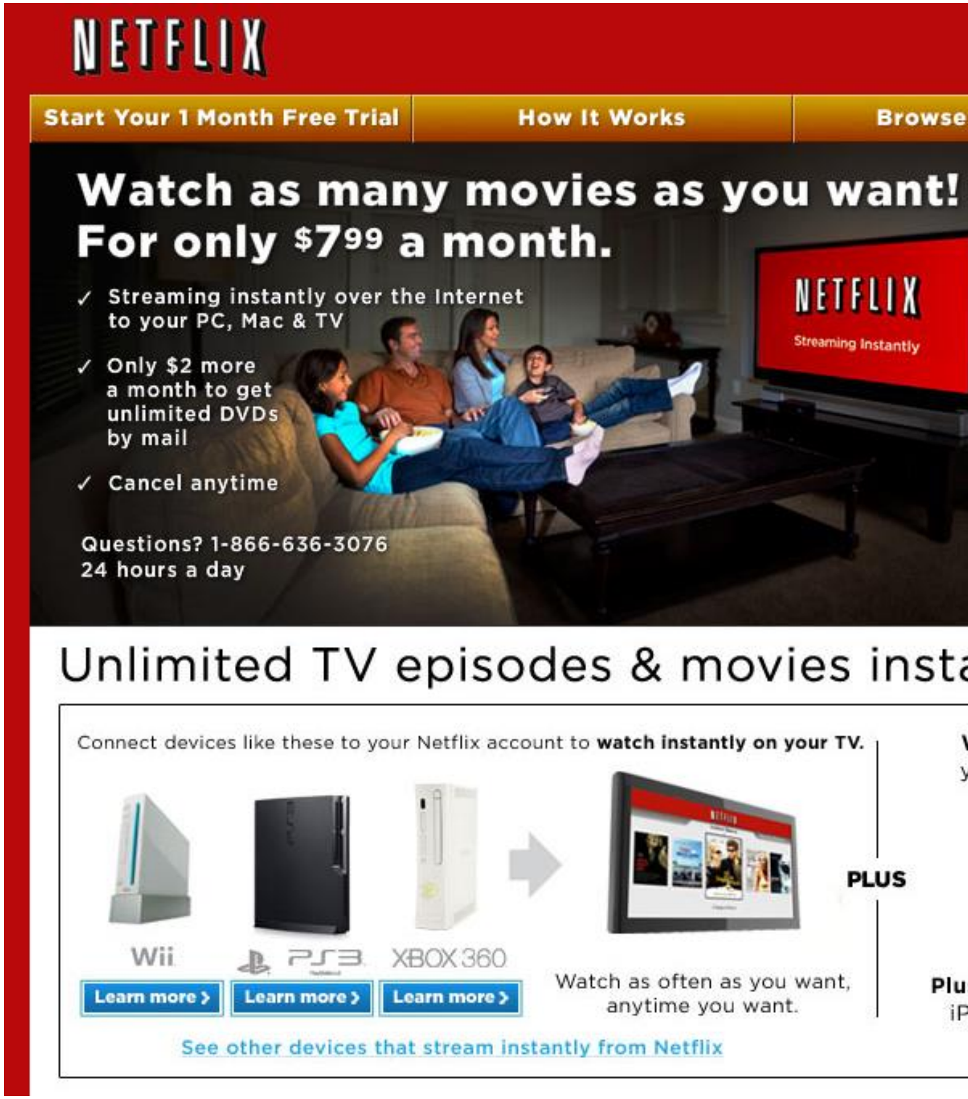
11.07.2013

> Novel type of applications !!!

data crunching** meets **number crunching



> The NetFlix Competition



> The NetFlix Competition (2)

Netflix' star rating system helps determine personalized movie recommendations. Now the company is looking to outside developers to improve those recommendations.

BUSINESS

The $1 Million Netflix Challenge

FRIDAY, OCTOBER 6, 2006 | BY KATE GREENE

VP Jim Bennett discusses how recommendation systems suggest your next movie and the challenges of building a better one.

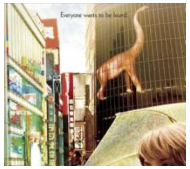
E-mail Audio » Print

Earlier this week, Netflix, the online movie rental service, announced it will award $1 million to anyone who can come up with an algorithm that improves the accuracy of its movie recommendation service.

In doing so, the company is putting out a call to researchers who specialize in mac
learning--the type of artificial intelligence used to build systems that recommend m
books, and movies. The entrant who can increase the accuracy of the Netflix
recommendation system, which is called Cinematch, by 10 percent by 2011 will w
prize.

Recommendation systems such as those used by Netflix, Amazon, and other We
retailers are based on the principle that if two people enjoy the same product, they'
likely to have other favorites in common too.

But behind this simple premise is a complex algorithm that incorporates millions o
ratings, tens of thousands of items, and ever-changing relationships between user
preferences.



$$
\begin{aligned}
\hat{r}_{ui} = b_{ui} &+ |\mathrm{N}(u)|^{-\frac{1}{2}} \sum_{j \in \mathrm{N}(u)} e^{-\beta_u \cdot |t_{ui} - t_{uj}|} c_{ij} + \\
&|\mathrm{R}(u)|^{-\frac{1}{2}} \sum_{j \in \mathrm{R}(u)} e^{-\beta_u \cdot |t_{ui} - t_{uj}|} ((r_{uj} - \tilde{b}_{uj}) w_{ij}) + \\
&\sum_{j \in \mathrm{R}(u)} e^{-\gamma_u \cdot |t_{ui} - t_{uj}|} ((r_{uj} - \tilde{b}_{uj}) d_{ij}).
\end{aligned}
$$

★★★★★ ? ★★★

> The NetFlix Competition (4)

	About Schmidt (2.24)	Lost In Translation (1.92)	Sideways (1.18)
Alice (1.98)	4.4	**4** 3.8	**2** 2.3
Bob (1.21)	**3** 2.7	**2** 2.3	2.7
Michael (2.30)	**5** 5.2	4.4	**3** 2.7

> The NetFlix Competition (6)

Movies

>100.000

Users

>10.000.000

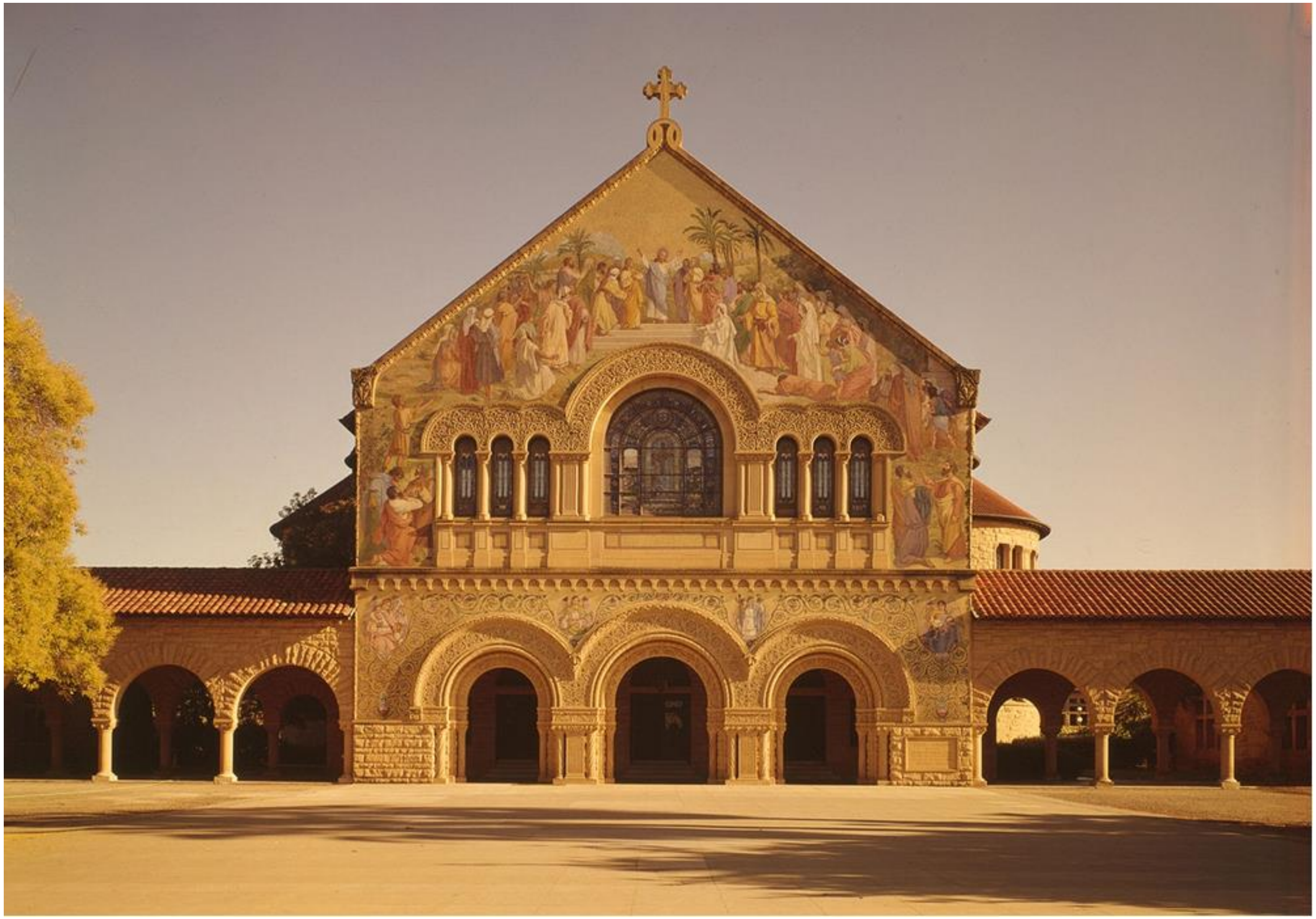
> A simple experiment ...

> ... our test object

color code := user rating

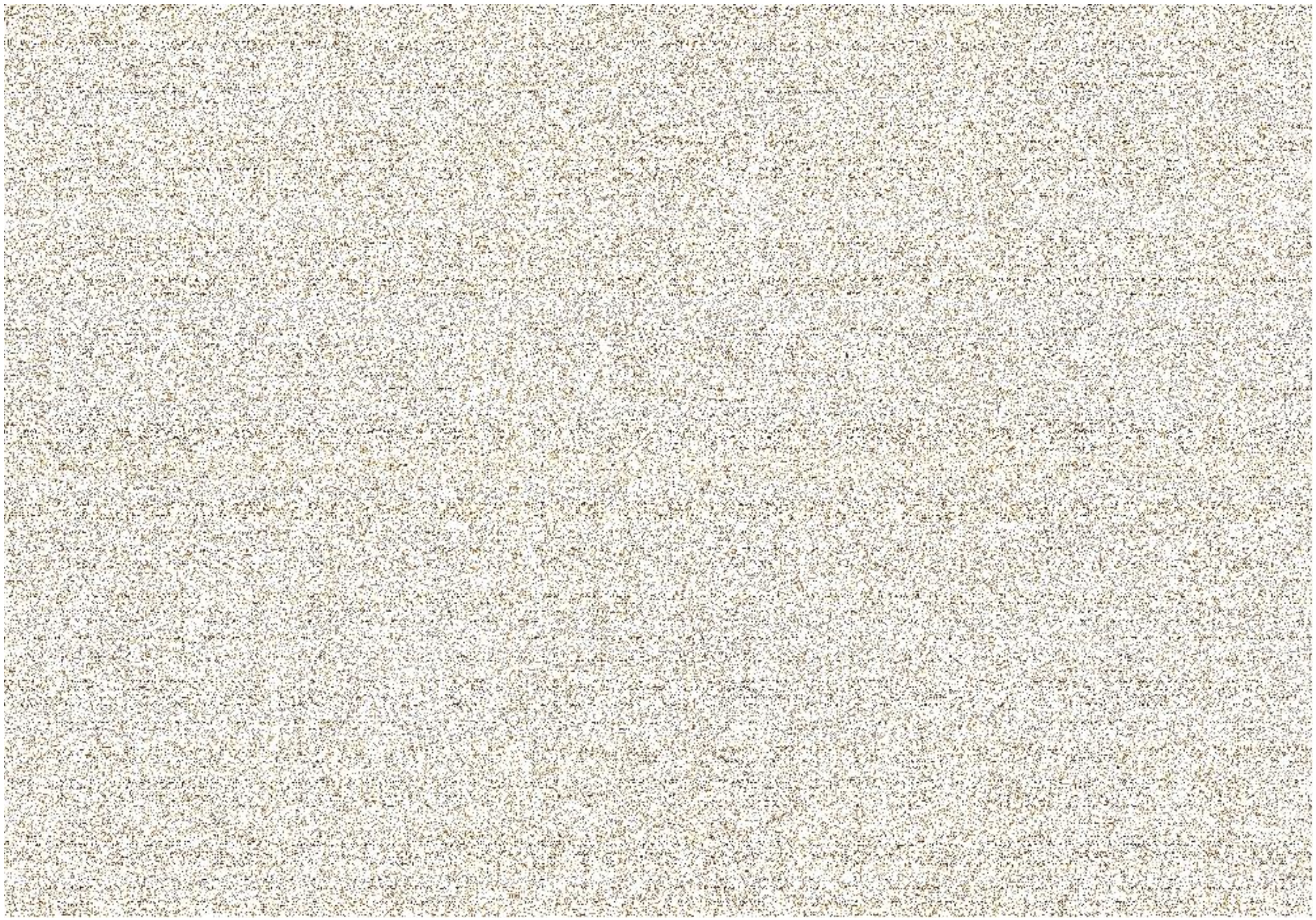
different movies

different users



Phase 1: drop 75% of all pixels

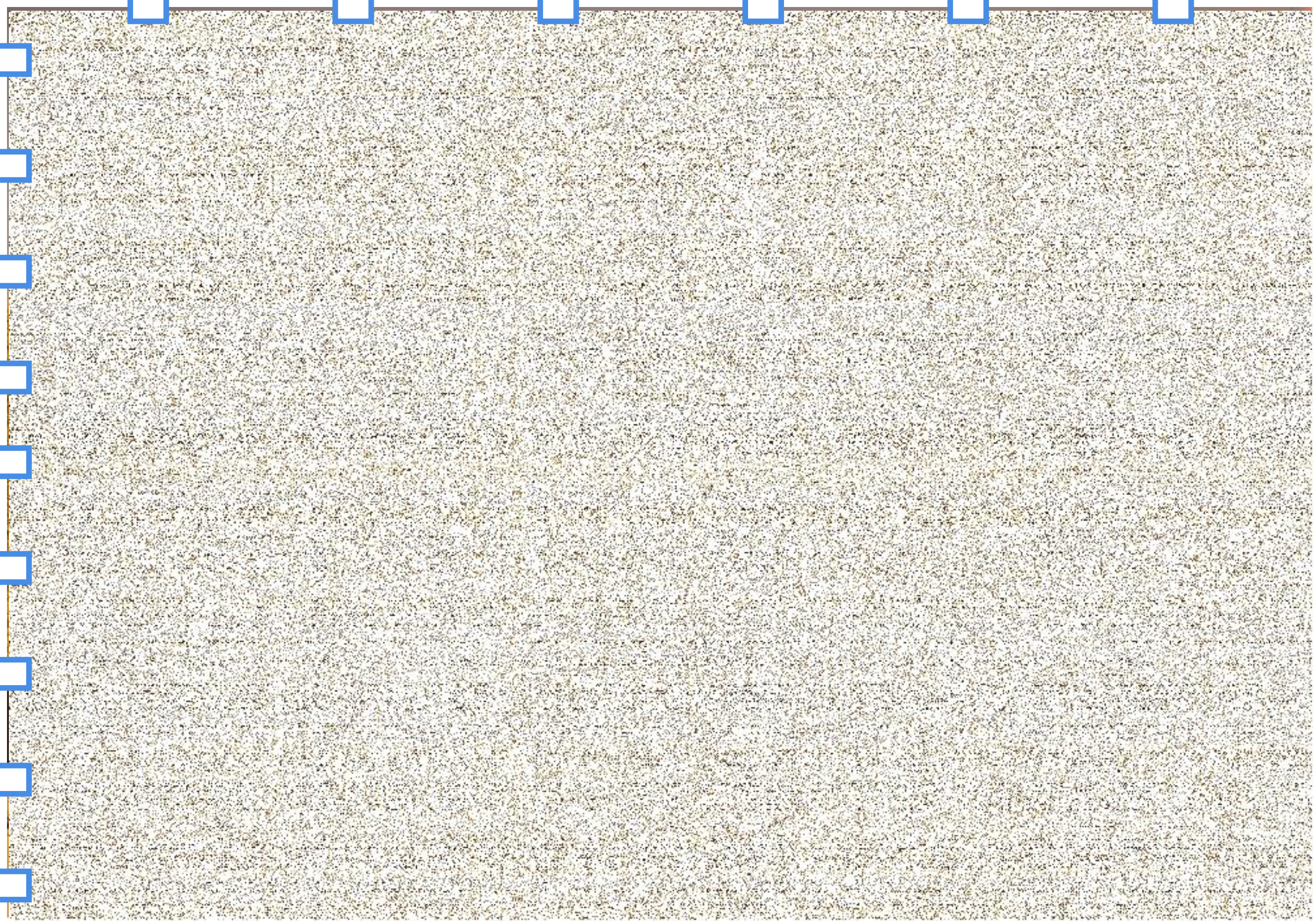
Phase 2: Random permutation of rows and columns



Phase 3: Determine the latent factors

99.05	101.5	5.99	1.50	99.96	100.0
98.30	93.45	8.00	4.98	100.0	99.71
92.37	114.7	2.98	1.01	99.27	101.5
97.91	94.18	3.97	2.02	99.35	101.3
103.4	103.7	7.97	4.03	99.60	100.8
101.4	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

6.0	1.5	5.94	1.53	99.05	101.5	5.99	1.50	99.96	100.0
8.0	5.0	7.86	4.67	98.30	93.45	8.00	4.98	100.0	99.71
3.0	1.0	2.77	1.15	92.37	114.7	2.98	1.01	99.27	101.5
4.0	2.0	3.91	1.88	97.91	94.18	3.97	2.02	99.35	101.3
8.0	4.0	8.27	4.15	103.4	103.7	7.97	4.03	99.60	100.8
[illegible]	[illegible]	9.13	3.02	101.4	100.6	9.02	2.97	100.3	98.95

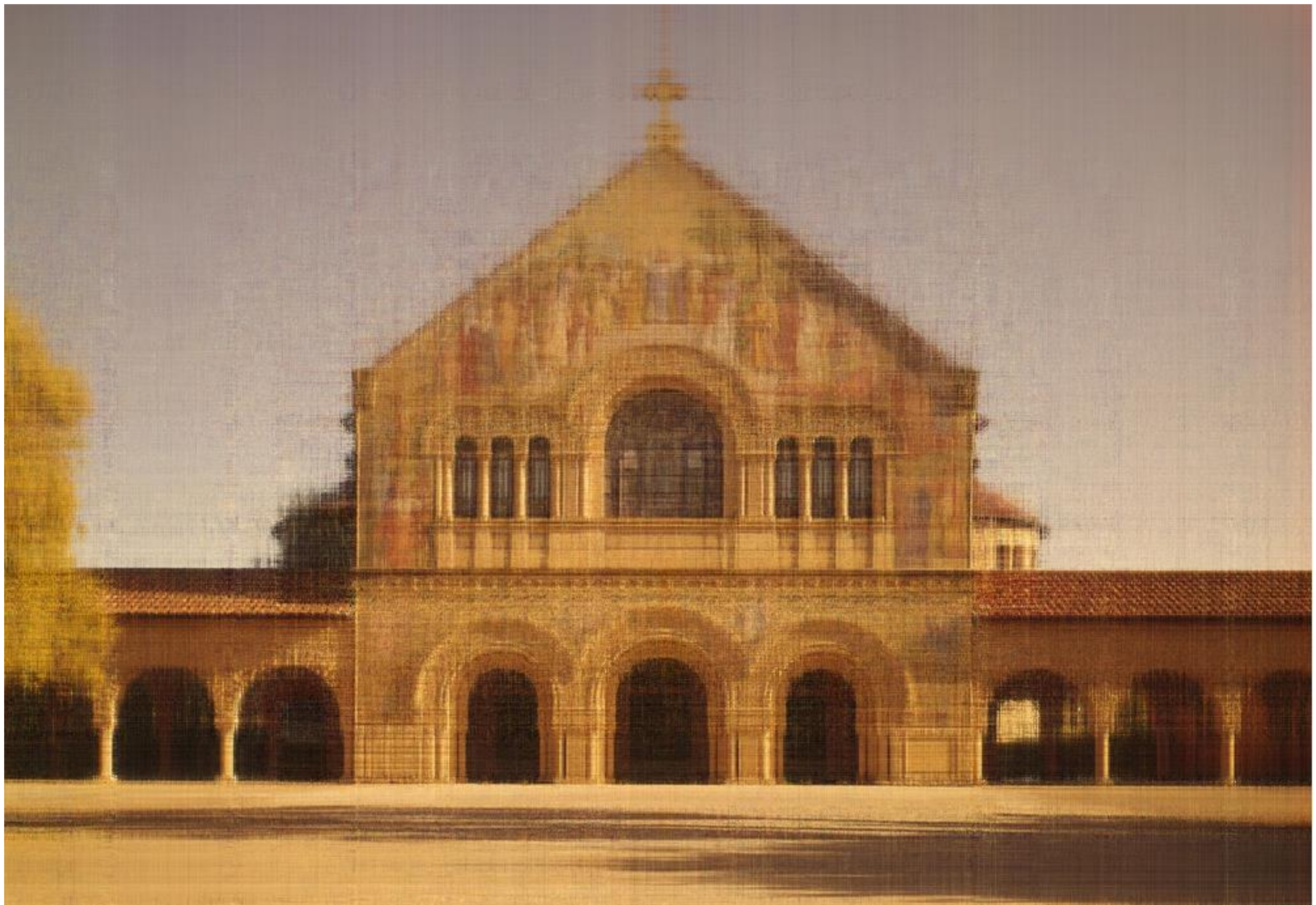


Phase 4: Reconstruction

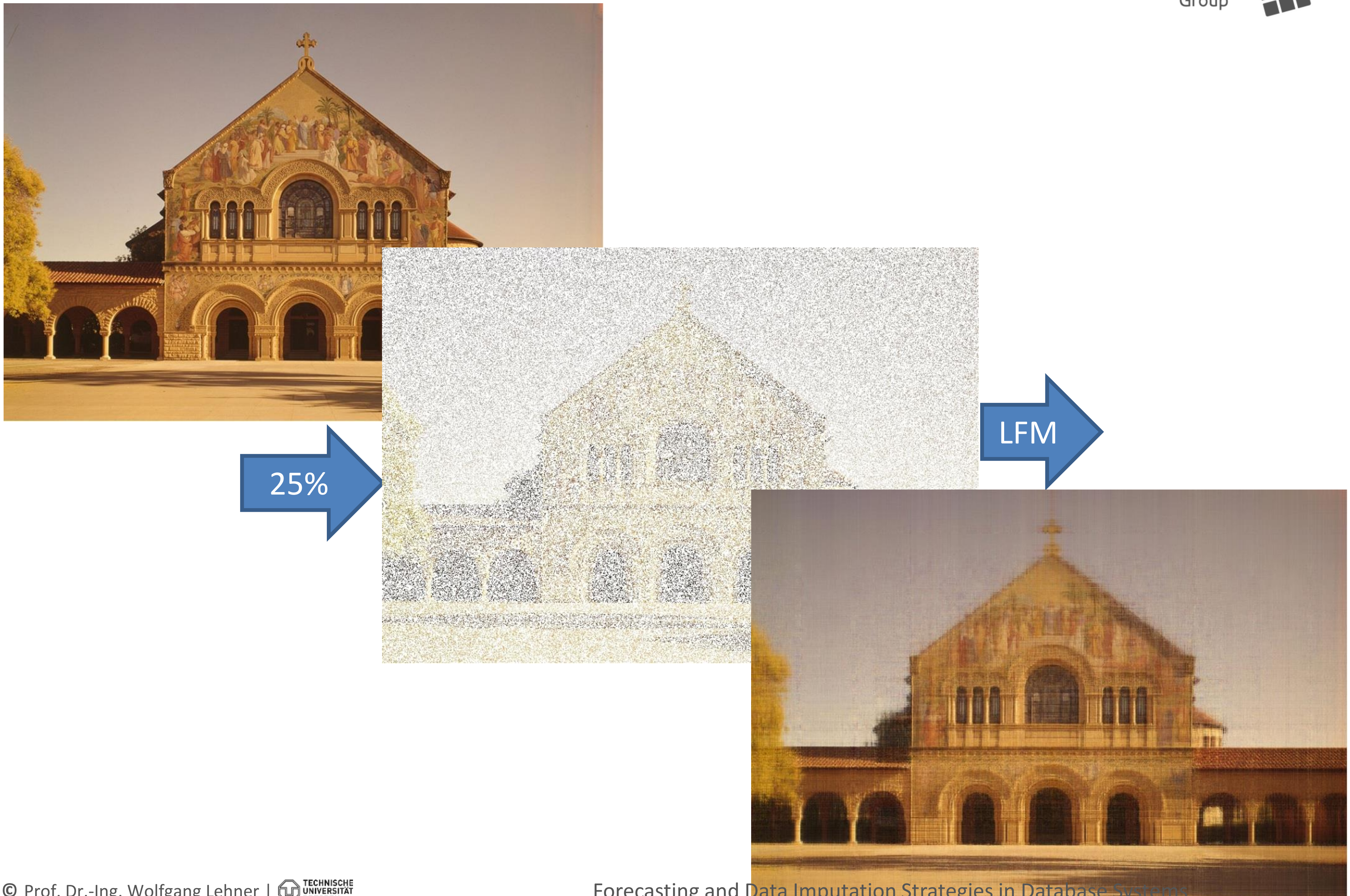
99.05	101.5	5.99	1.50	99.96	100.0
98.30	93.45	8.00	4.98	100.0	99.71
92.37	114.7	2.98	1.01	99.27	101.5
97.91	94.18	3.97	2.02	99.35	101.3
103.4	103.7	7.97	4.03	99.60	100.8
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	98.95

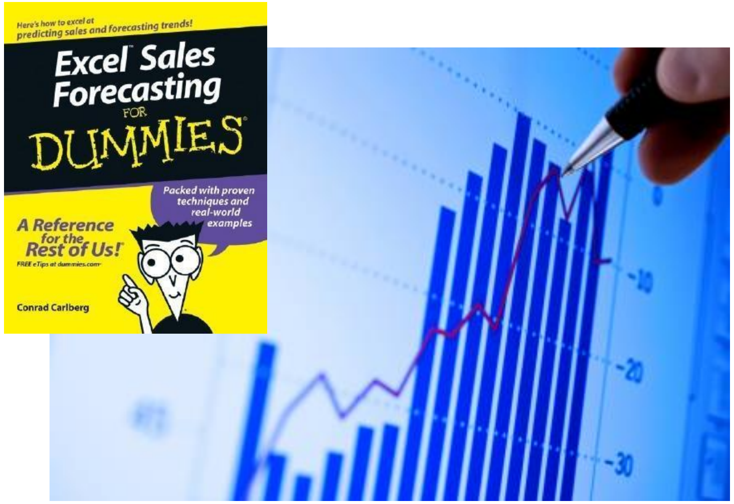
100.0	99.71	101.5	101.3	100.8	98.95
99.96	100.0	99.27	99.35	99.60	100.3
1.50	4.98	1.01	2.02	4.03	2.97
5.99	8.00	2.98	3.97	7.97	9.02
101.5	93.45	114.7	94.18	103.7	100.6
99.05	98.30	92.37	97.91	103.4	101.4
1.53	4.67	1.15	1.88	4.15	3.02
5.94	7.86	2.77	3.91	8.27	[illegible]
1.5	5.0	1.0	2.0	4.0	[illegible]
6.0	8.0	3.0	4.0	8.0	[illegible]

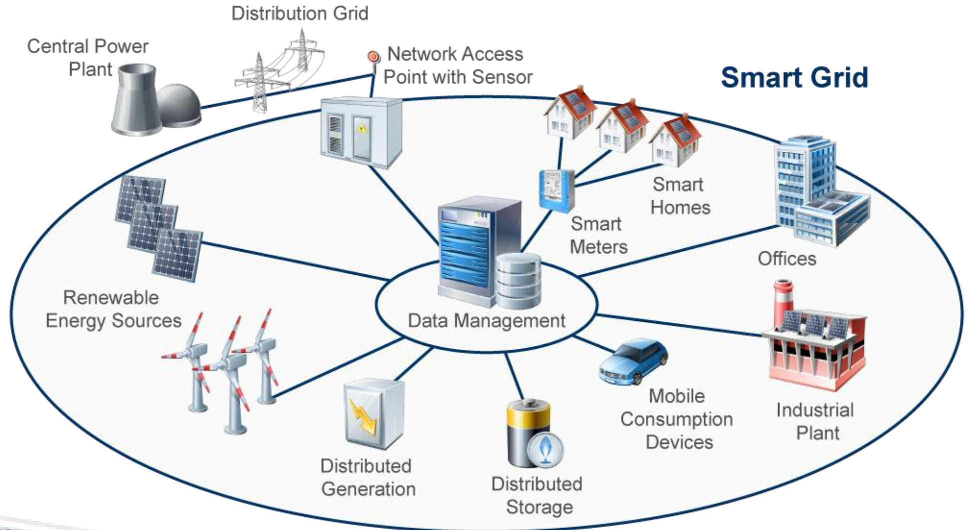
Phase 5: Final Result Generation



> The Experiment ...







➔ Multi-Dimensional Time Series Data

> Forecast Models

"It's tough to make predictions,
especially about the future."
-- Mark Twain

Given

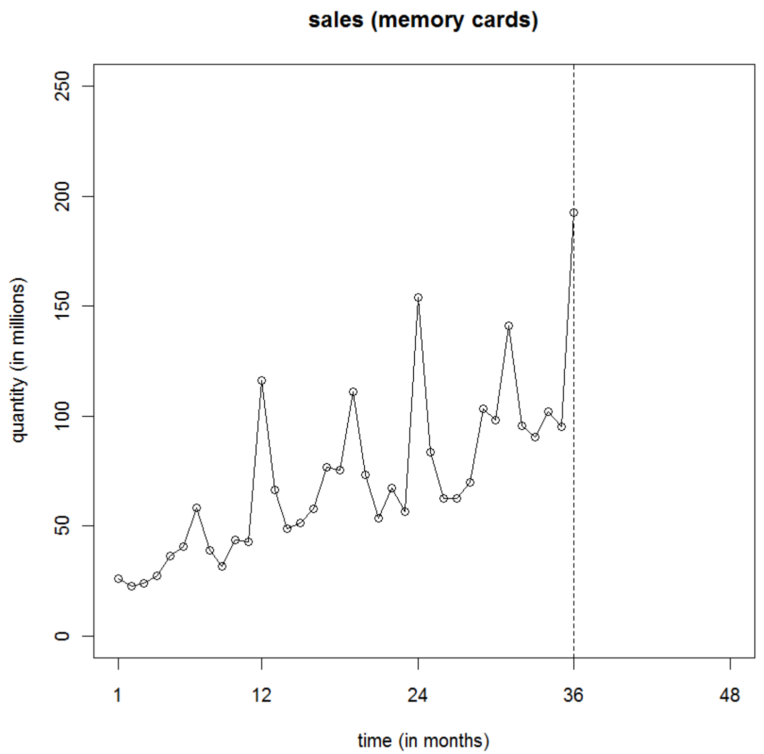
- Time series with numerical values as training data

Goal

- Predict future values for arbitrary future point in times (forecast horizon)
- Include trend and seasonality

Applications

- Planning of sales and budget
- Price development
- Inventory, manufacturing
- Climate, weather, environment
- Economic indicators
- Stocks



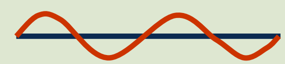
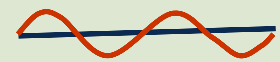
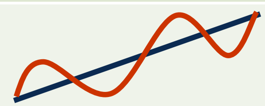
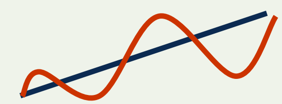
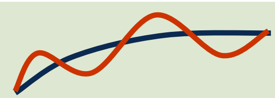
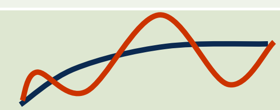
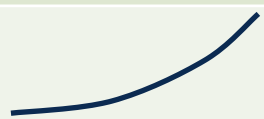
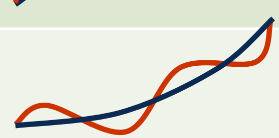
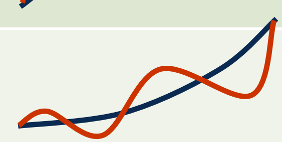
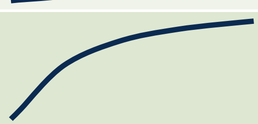
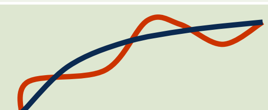
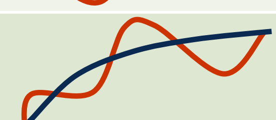
> Forecast Models

Diverse Time Series Data

- Find forecast model that minimizes the forecast error

Example: Exponential Smoothing Framework

- 30 variants (additional additive or multiplicative errors)

Trend/Season	No Season	Add. Season	Mult. Season
No Trend			
Add. Trend			
Add. Damp. Trend			
Mult. Trend			
Mult. Damp. Trend			

> Agenda

Mathematical Foundations

In-DBMS Time Series Forecasting

- Flash-Forward Query Project

Flash Forward Query

Query Processing and Optimization

Model Configuration Advisor

- Forecasting the Data Cube
- Selection of Model Configurations

Notification-based Forecast Queries

Beyond Forecast Models

- Model Refinement

Mathematical Foundations

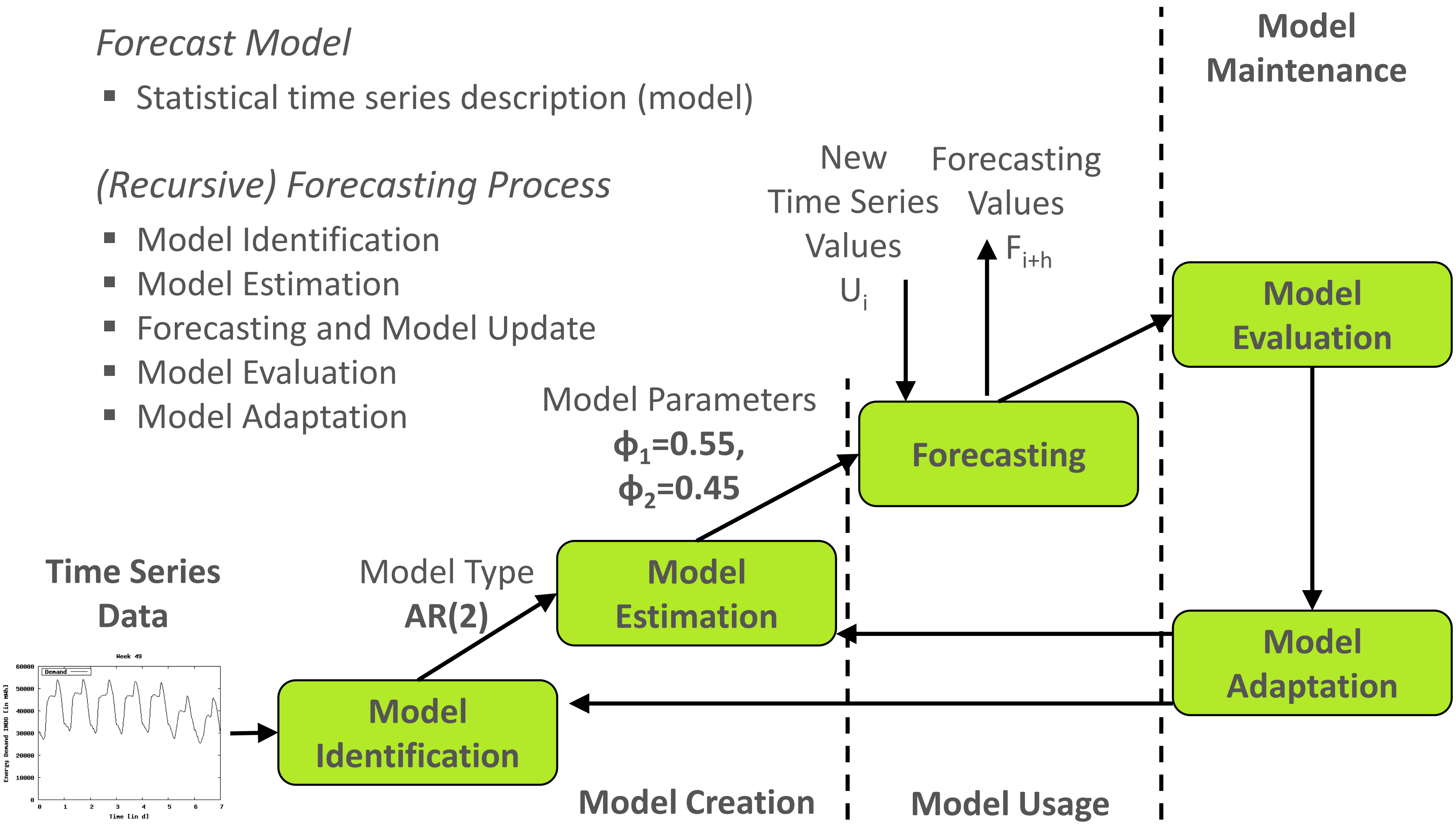
> Model-Based Forecasting

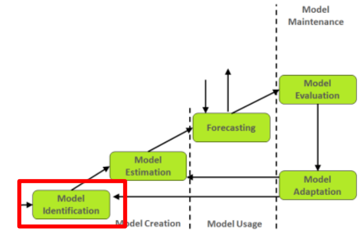
Forecast Model

- Statistical time series description (model)

(Recursive) Forecasting Process

- Model Identification
- Model Estimation
- Forecasting and Model Update
- Model Evaluation
- Model Adaptation





Model Identification

Forecast Model Types / Classes

Base Forecast Models → **Domain-Specific Extensions**

Domain-Specific Extensions (White-Box)

- **HWT** (Single-Equation)
- **EGRV** (Multi-Equation)

Base Forecast Models (Gray-Box / Black-Box):

(Auto)Regression

- **AR**
- **MA**
- **ARMA**
- **ARIMA**
- **SARIMA**
- **ARMAX**
- **MLR** (Multiple Linear Regression)

Exponential Smoothing

- **SESM** (Single Exponential Smoothing)
- **DESM** (Double Exponential Smoothing)
- **TESM / HoltWinters** (Triple Exponential Smoothing)

Machine Learning

- **BN** (Bayesian Networks)
- **SVM** (Support Vector Machines)
- **SVR** (Support Vector Regression)
- **ANN** (Artificial Neural Networks)

White-Box | Gray-Box | Black-Box

> Model Estimation

Problem

- Instantiate a forecast model w.r.t. meta model and training data set

Example

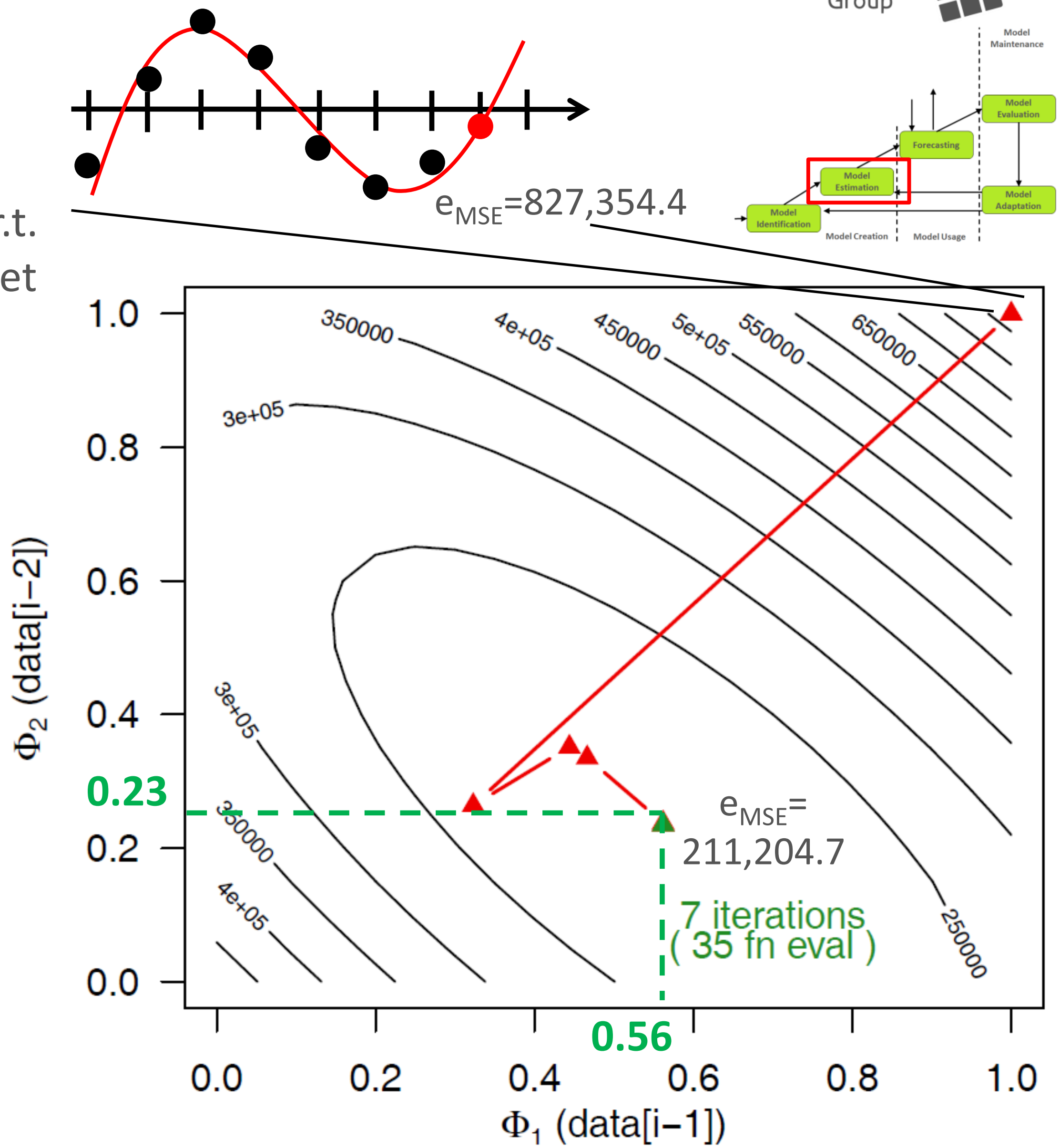
- Forecast Model Type **AR(2)**:

$$\hat{y}_t = \phi_1 \cdot y_{t-1} + \phi_2 \cdot y_{t-2}$$

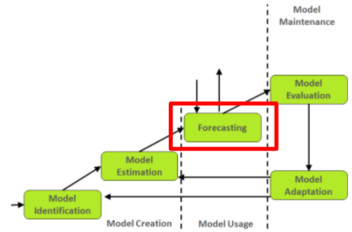
- Error Metric: **MSE**

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- Parameter Estimator **L-BFGS-B**



> Model Usage

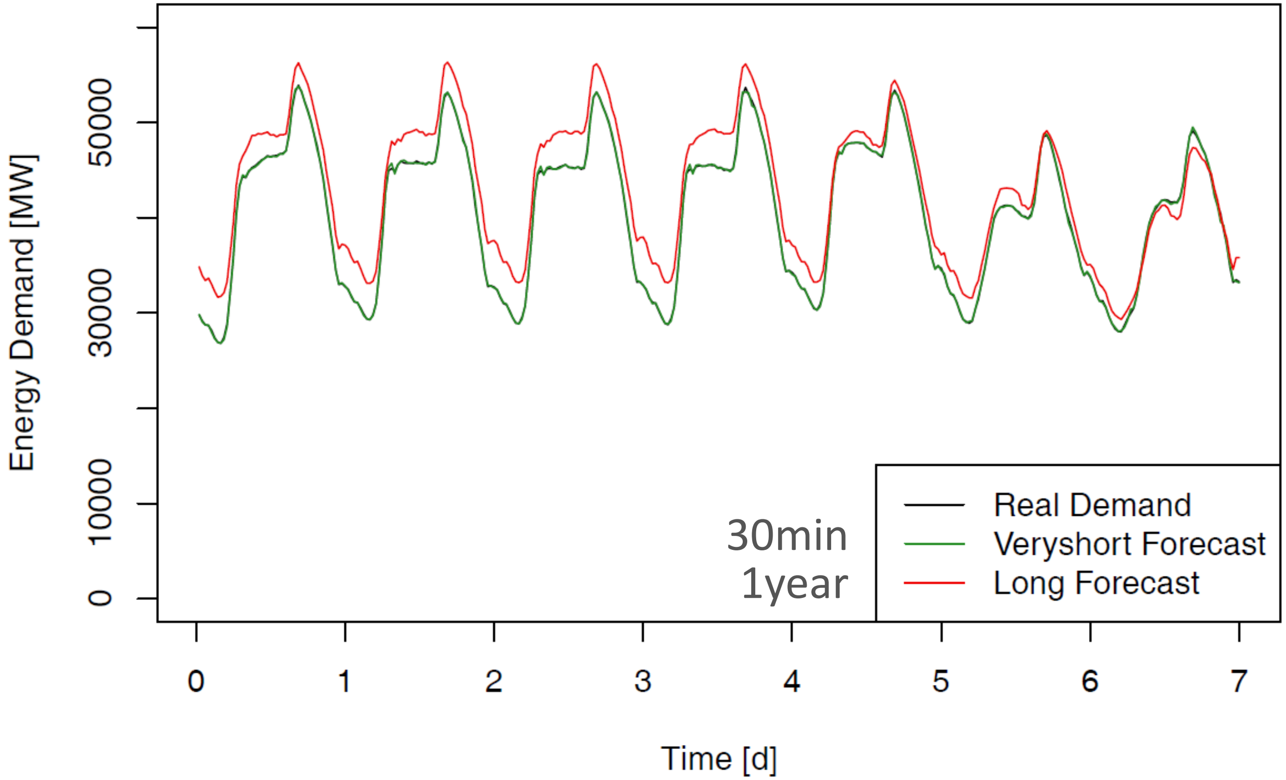


Forecasting

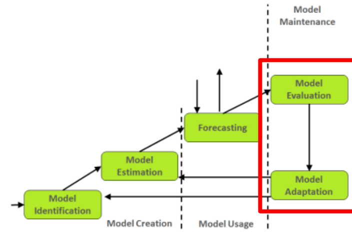
- Use the estimated forecast model $\hat{y}_t = 0.56 \cdot y_{t-1} + 0.23 \cdot y_{t-2}$
- Create h forecast values (forecast horizon)
- Update model state for new measurements (e.g., exponential smoothing)

Example Forecast (EGRV)

- SMAPE_{vshort} **=0.0021**
- SMAPE_{long} **=0.0755**



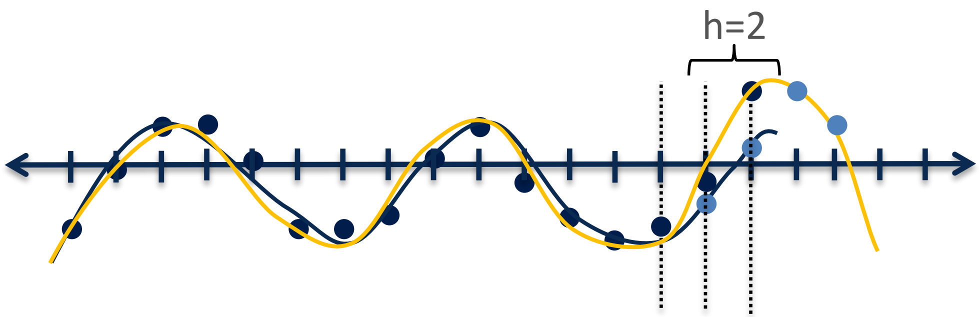
> Model Maintenance



Model Evaluation

- Goal: Trigger model adaptation only if necessary

- Fixed Interval Techniques (# updates, time interval)
- Continuous Evaluation Techniques (threshold, on-demand)



Model Adaptation

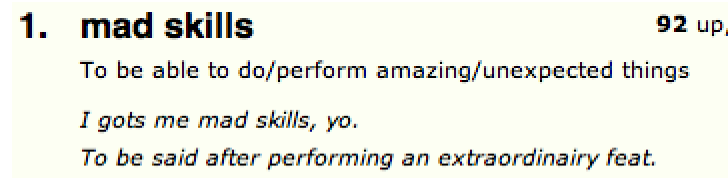
- Goal: Adapt the forecast model to the changed time series (if necessary)

- Model Re-Identification
- Model Re-Estimation (old model as start point)

In-DBMS Time Series Forecasting

Flash Forward Query

> MAD Skills



1. mad skills 92 up,

To be able to do/perform amazing/unexpected things

I gots me mad skills, yo.
To be said after performing an extraordinairy feat.

Magnetic

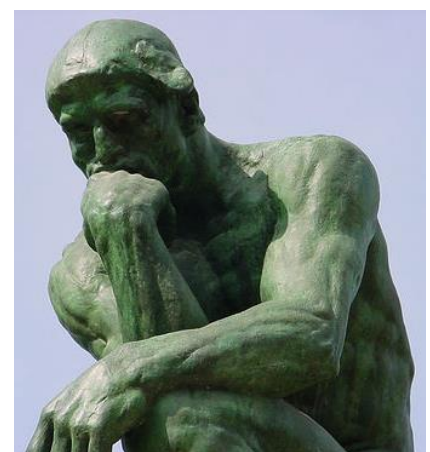
- **„Attract data and practitioners“**
- Use all available data sources independent of their quality

Agile

- **„Rapid iteration: ingest, analyze, productionalize“**
- Continuous and rapid evolution of physical and logical structures
- ELT (Extraction, Loading, Transformation)

Deep

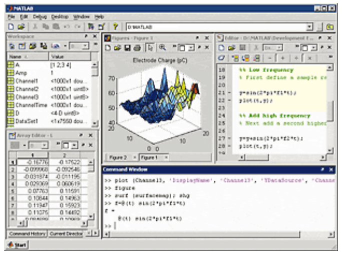
- **„Sophisticated analytics in Big Data“**
- Extended algorithmic runtime environment
- Ad-hoc advanced analytics and statistics

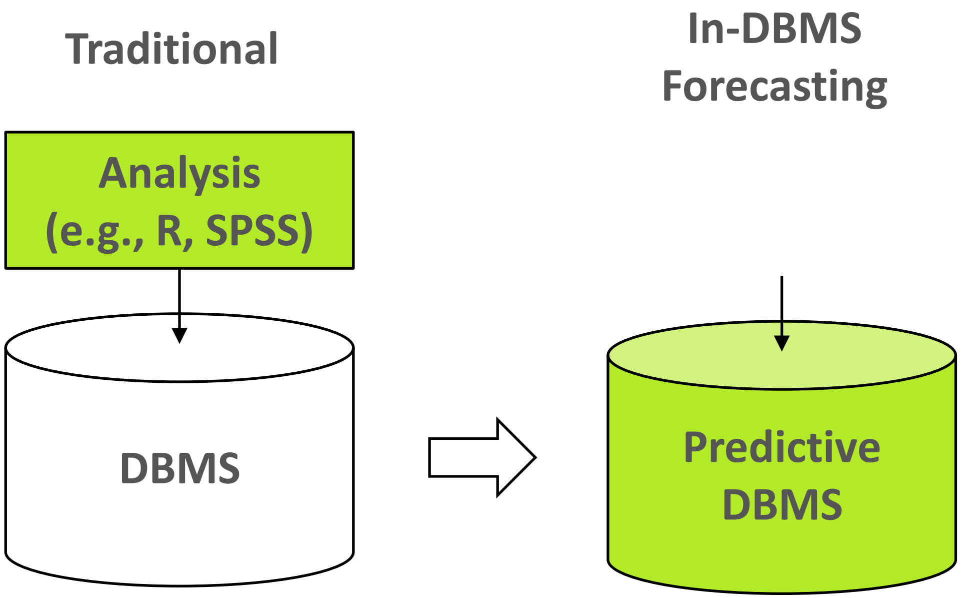


> Integrated Data Analytics

Integration of advanced analytics into scalable database management systems

- Traditionally forecasting is performed manually in external statistical systems
- Support of transparent and automatic in-DBMS forecasting

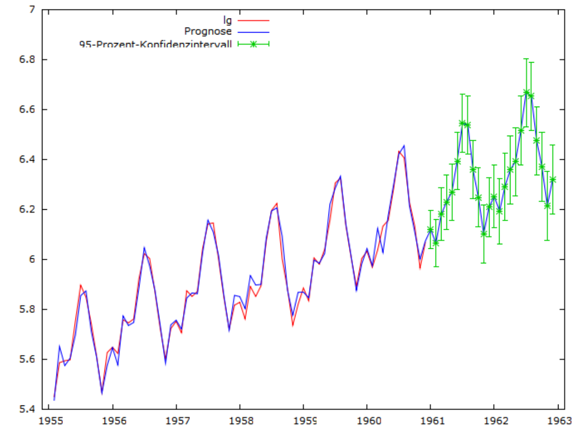




> Integrated Forecasting

Model-based Prediction

- We employ (classical) time series models (e.g. exponential smoothing, ARIMA)
- Used to
  - Explain time series from it's history
  - And—possibly—from exogenous inputs



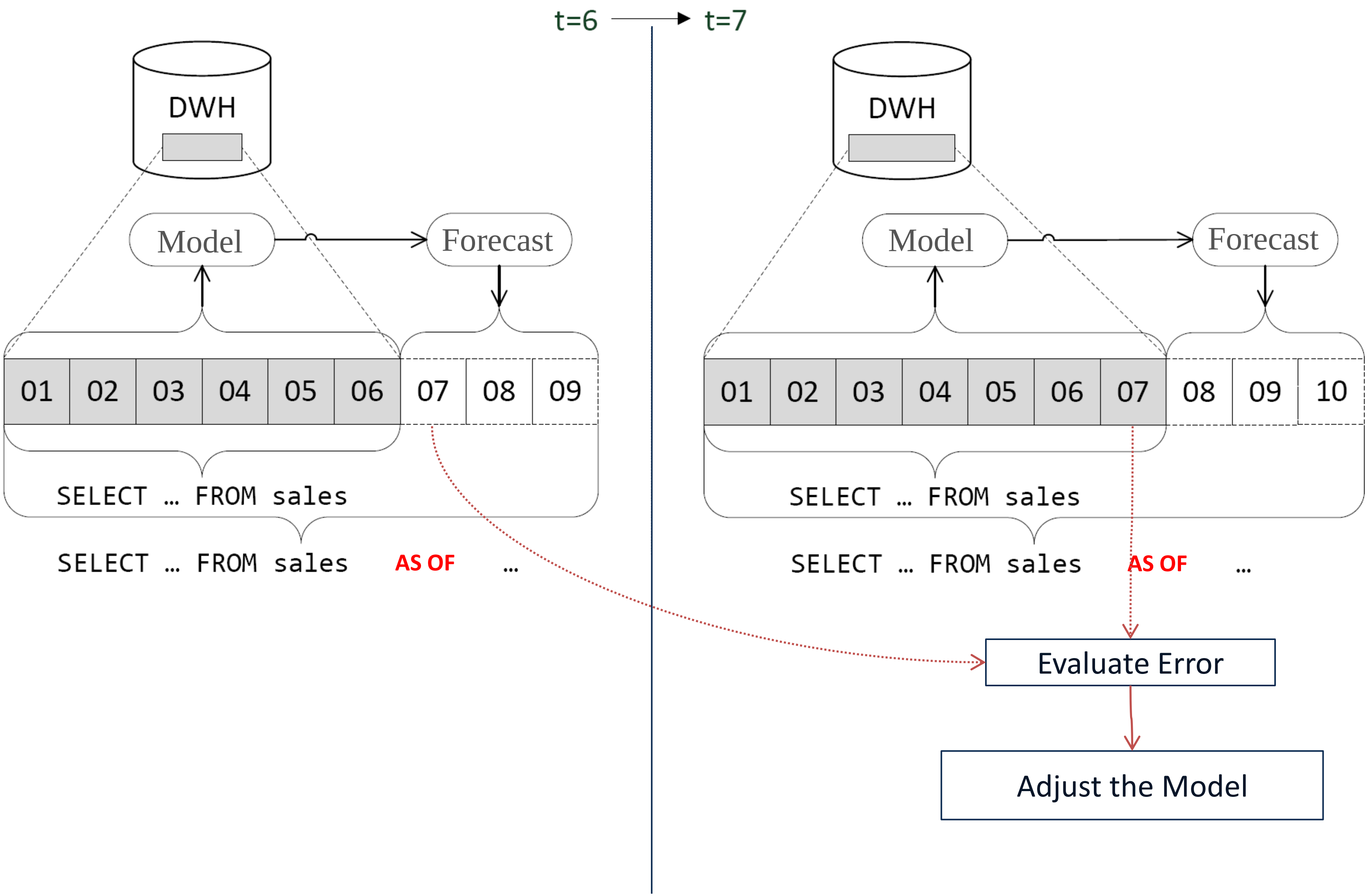
Related Work

- Customized functions with proprietary languages
  - SQL Server 2012: ARIMA, autoregressive trees
  - Oracle: exponential smoothing, non-linear regression
- Bi-directional communication
  - Reuse existing statistical tools (e.g. R)
  - SAP HANA, Oracle, IBM Netezza
- Model-based views

Key Elements

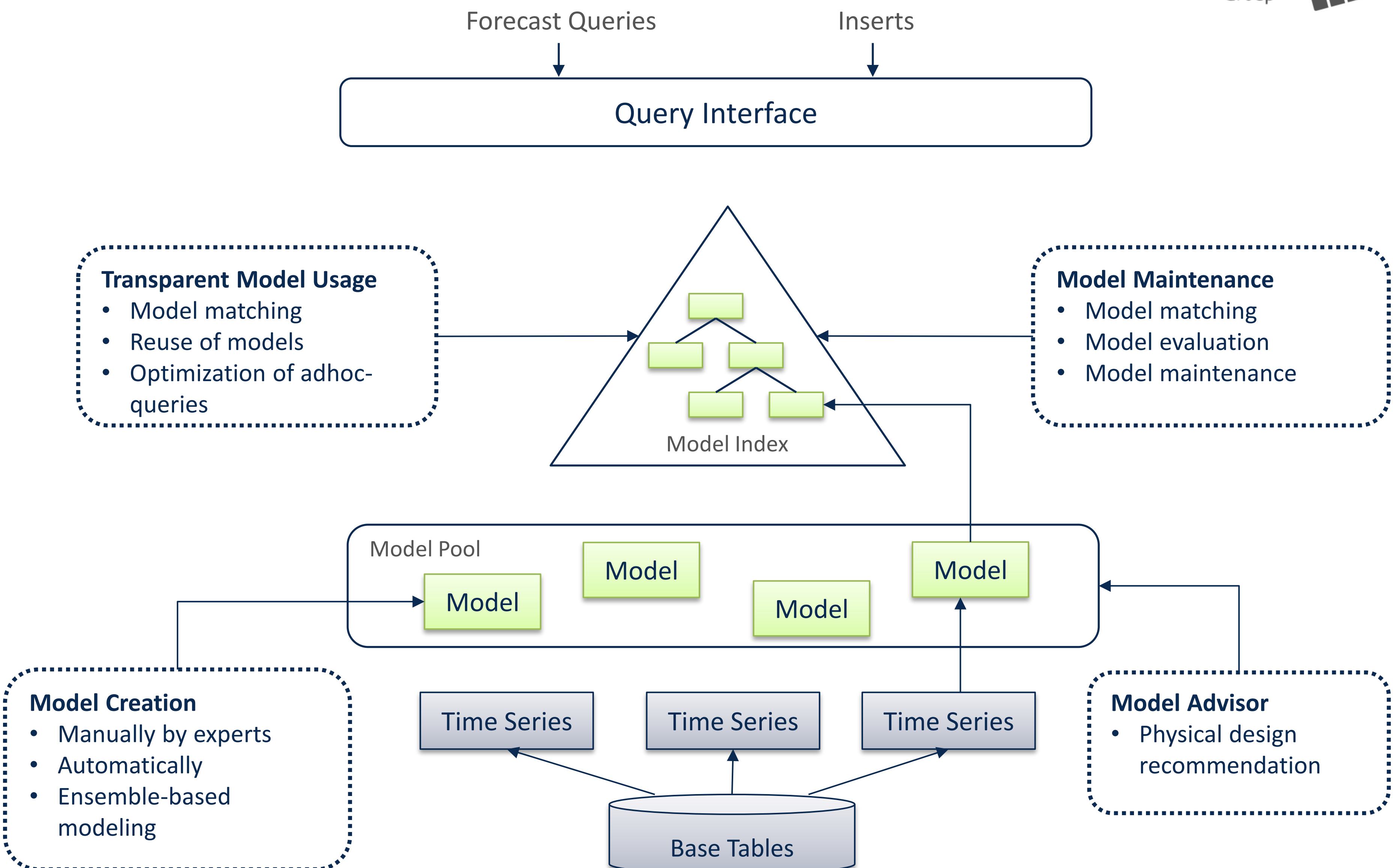
- Declarative querying
- Automatic model creation
- Automatic model maintenance
- Forecast model advisor

> Scenario

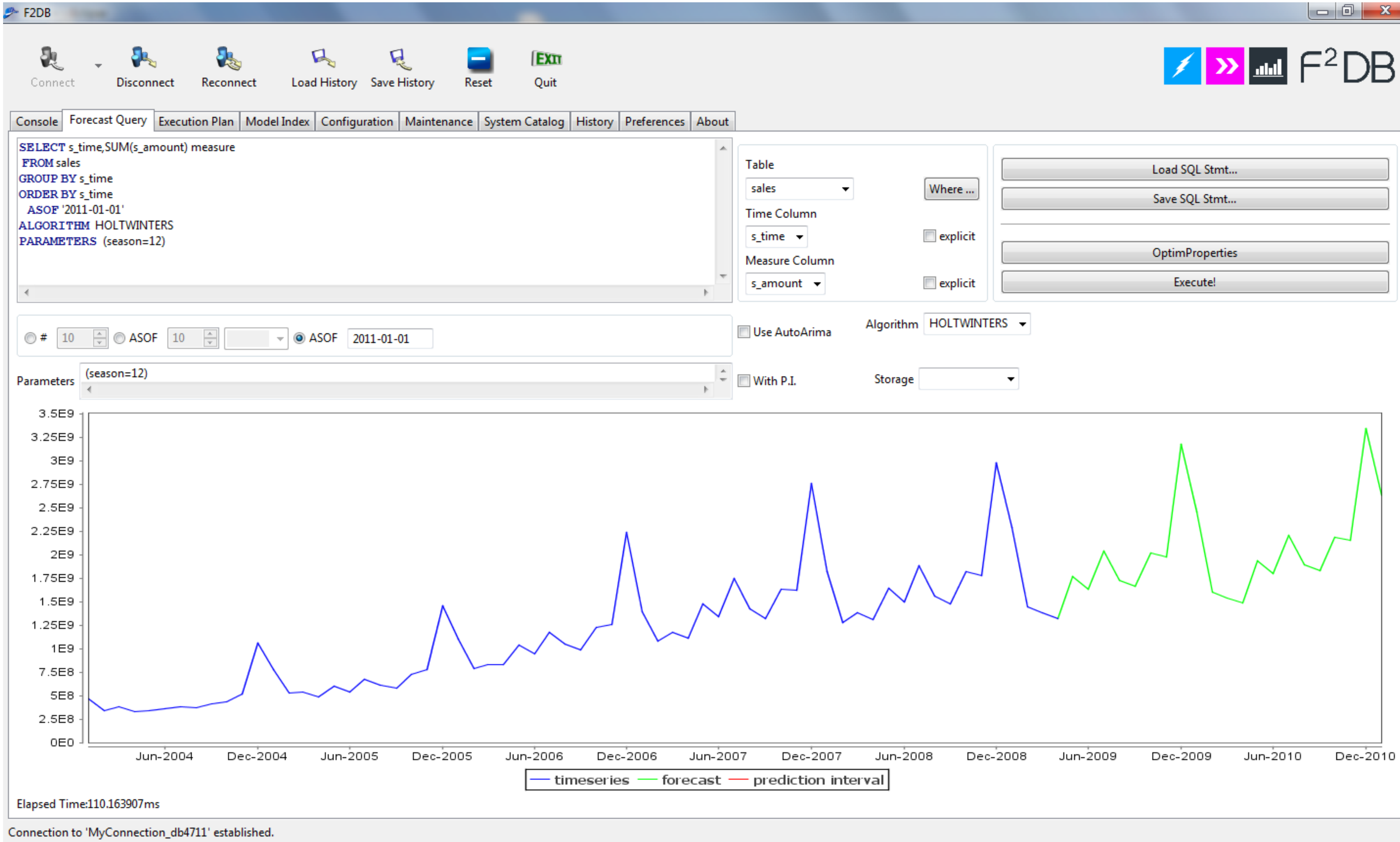


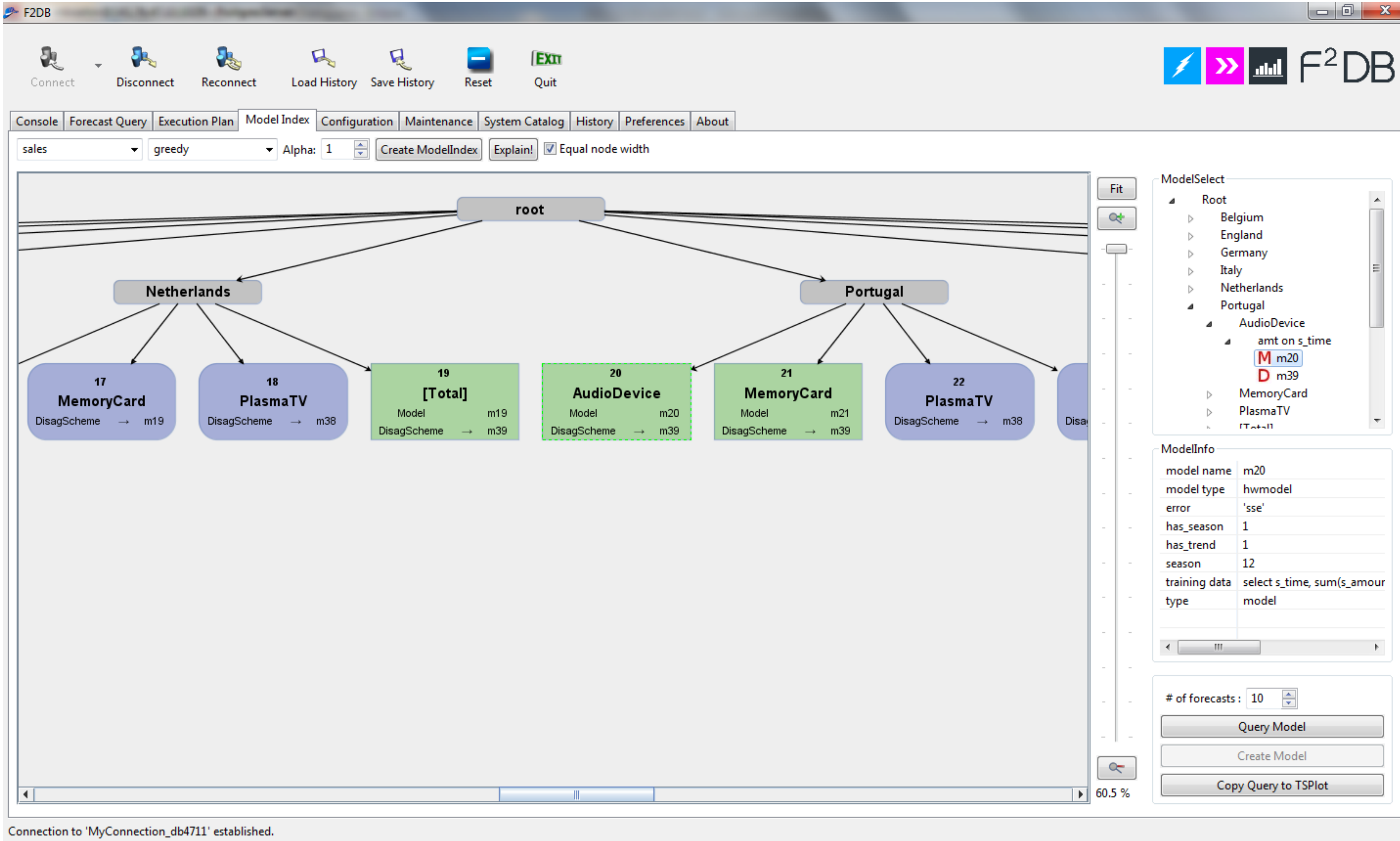
t=6
t=7
DWH
Model
Forecast
01 02 03 04 05 06 07 08 09
SELECT … FROM sales
SELECT … FROM sales AS OF …
DWH
Model
Forecast
01 02 03 04 05 06 07 08 09 10
SELECT … FROM sales
SELECT … FROM sales AS OF …
Evaluate Error
Adjust the Model

> System Architecture



> F2DB


F2DB
Connect
Disconnect
Reconnect
Load History
Save History
Reset
Quit
F2DB
Console
Forecast Query
Execution Plan
Model Index
Configuration
Maintenance
System Catalog
History
Preferences
About
SELECT s_time,SUM(s_amount) measure
FROM sales
GROUP BY s_time
ORDER BY s_time
ASOF '2011-01-01'
ALGORITHM HOLTWINTERS
PARAMETERS (season=12)
Table
sales
Where ...
Time Column
s_time
explicit
Measure Column
s_amount
explicit
Load SQL Stmt...
Save SQL Stmt...
OptimProperties
Execute!
10 ASOF 10 ASOF 2011-01-01
Use AutoArima
Algorithm HOLTWINTERS
Parameters (season=12)
With P.I.
Storage
3.5E9
3.25E9
3E9
2.75E9
2.5E9
2.25E9
2E9
1.75E9
1.5E9
1.25E9
1E9
7.5E8
5E8
2.5E8
0E0
Jun-2004 Dec-2004 Jun-2005 Dec-2005 Jun-2006 Dec-2006 Jun-2007 Dec-2007 Jun-2008 Dec-2008 Jun-2009 Dec-2009 Jun-2010 Dec-2010
timeseries
forecast
prediction interval
Elapsed Time:110.163907ms
Connection to 'MyConnection_db4711' established.


F2DB
Connect
Disconnect
Reconnect
Load History
Save History
Reset
Quit
F2DB
Console
Forecast Query
Execution Plan
Model Index
Configuration
Maintenance
System Catalog
History
Preferences
About
sales
greedy
Alpha: 1
Create ModelIndex
Explain!
Equal node width
root
Netherlands
Portugal
17 MemoryCard DisagScheme → m19
18 PlasmaTV DisagScheme → m38
19 [Total] Model m19 DisagScheme → m39
20 AudioDevice Model m20 DisagScheme → m39
21 MemoryCard Model m21 DisagScheme → m39
22 PlasmaTV DisagScheme → m38
Fit
60.5 %
ModelSelect
Root
Belgium
England
Germany
Italy
Netherlands
Portugal
AudioDevice
amt on s_time
M m20
D m39
MemoryCard
PlasmaTV
ModelInfo
model name m20
model type hwmodel
error 'sse'
has_season 1
has_trend 1
season 12
training data select s_time, sum(s_amour
type model
# of forecasts : 10
Query Model
Create Model
Copy Query to TSPlot
Connection to 'MyConnection_db4711' established.

Query Processing and Optimization

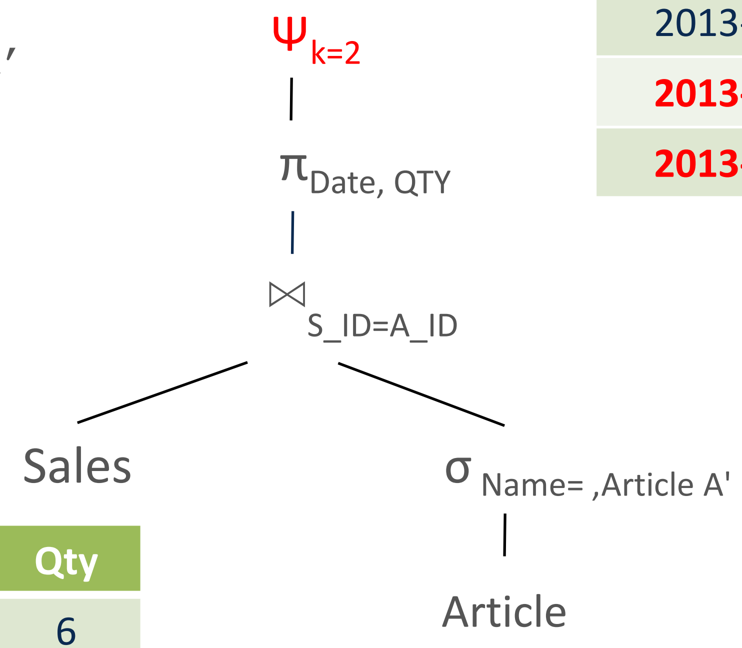
SQL Forecast Query

```
SELECT  Date, Quantity
  FROM  Sales, Article
  WHERE S_ID = A_ID AND
        Name = 'Article A'
  AS OF 2013-05-02
```

Date	Qty
2013-04-28	6
2013-04-29	5
2013-04-30	7
2013-05-01	**6**
2013-05-02	**6.5**

Query Plan

- Forecast operator Ψ



...	S_ID	Date	Qty
...	1	2013-04-28	6
...	1	2013-04-29	5
...	2	2013-04-30	1
...	1	2013-04-30	7

A_ID	Name	...
1	Article A	...
2	Acticle B	...

Selection invariance

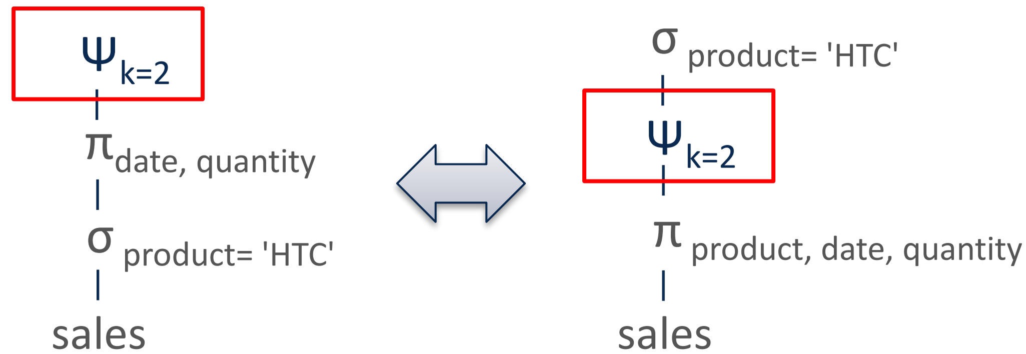
- Condition filtering out whole clusters

Projection invariance

- If neither time, measure or cluster attribute

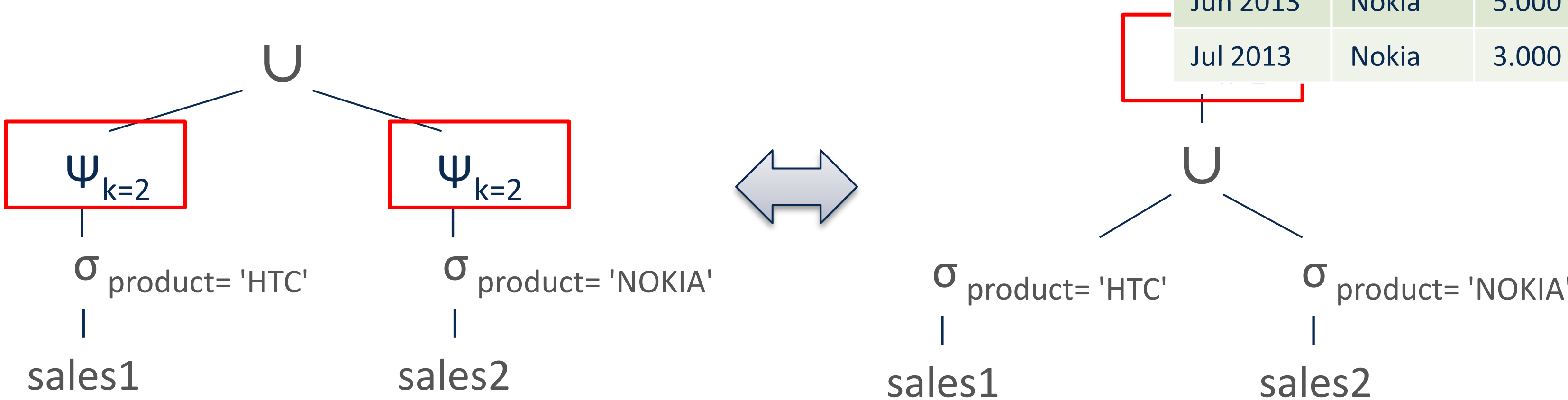
Union invariance

- On different relations



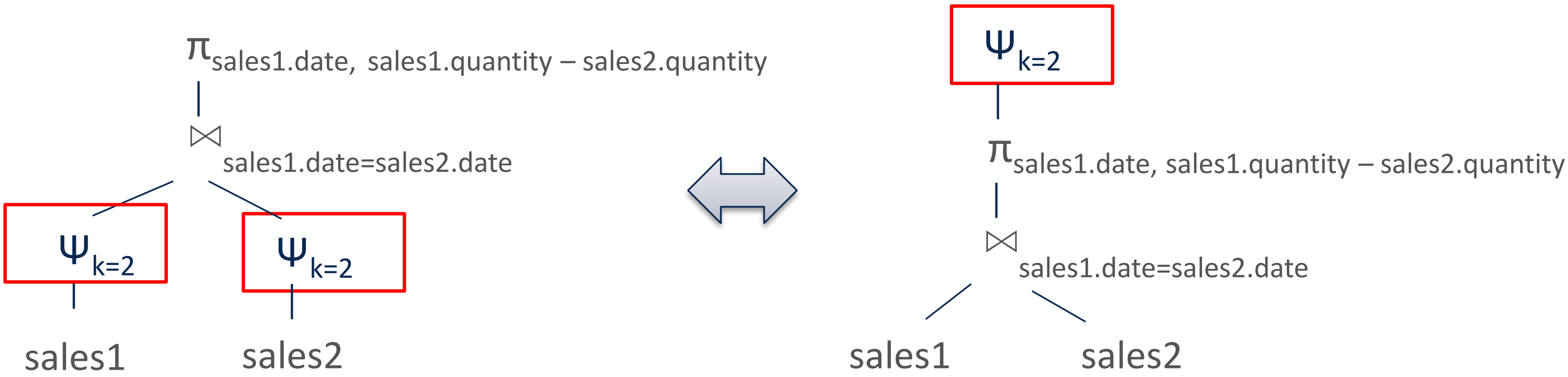
➔ Influence

date	product	quantity
Jun 2013	HTC	36.000
Jul 2013	HTC	38.000
Jun 2013	Nokia	5.000
Jul 2013	Nokia	3.000



Restructuring might also influence accuracy

- E.g. join, aggregation



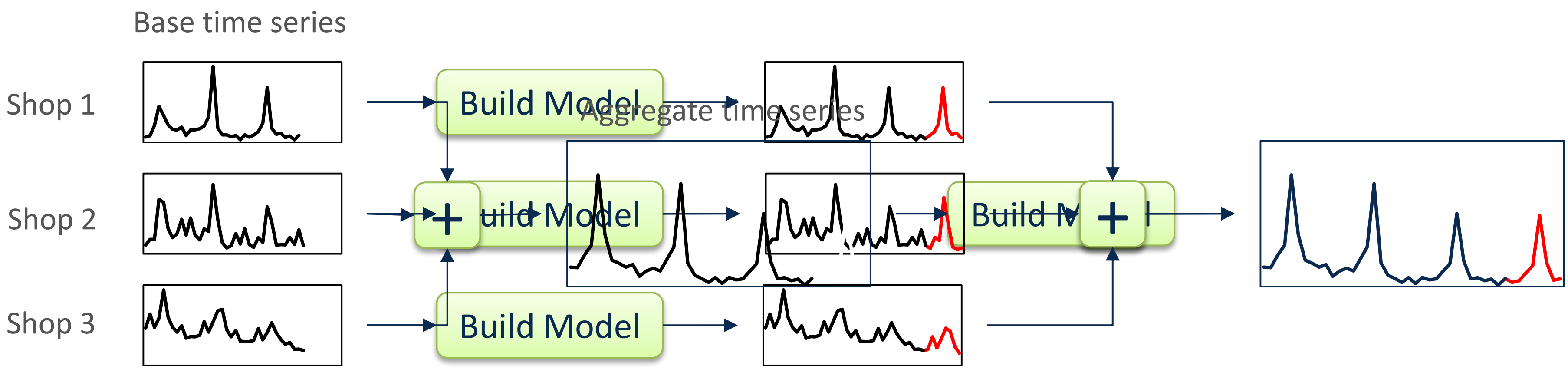
Expensive model creation

Expensive join

➔ Cost model required: accuracy and runtime

> Example: Aggregation Queries

How to compute aggregation forecast queries?



	Runtime	Accuracy
	⇧	⇩
	⇩	⇧
	⇨	⇧

> Example: Aggregation Queries

		t-1	t		t+1	
base time series	…	8	9	→	10	Model
	…	48	49			
	…	8	9	→	10	Model
aggregate	…	64	67	→	ŷ?	$= \alpha \sum Forecast$

Uniform Estimation $\alpha = \frac{\#\ base\ series}{\#\ base\ series\ in\ sample}$ $\frac{3}{2} \cdot 20 = 30$

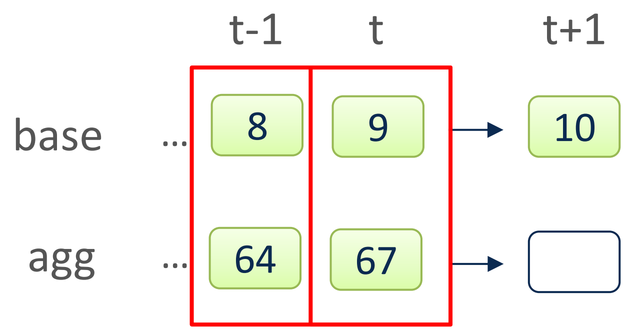
Estimation with Historical Ratios

$$ratio = \frac{base\ series}{aggregate\ series} \Rightarrow \alpha = \frac{1}{\sum ratios} \qquad \frac{1}{\frac{9}{67}+\frac{9}{67}} = 3.7 \cdot 20 = 74$$

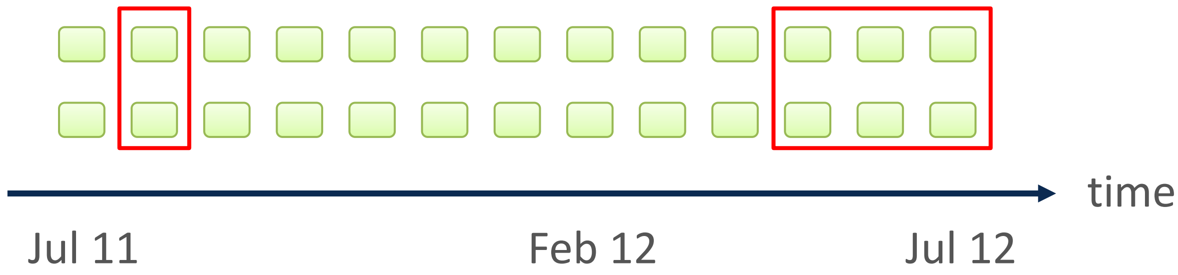
> Example: Aggregation Queries

Calculation of Historical Ratios

- Different approaches possible
  - Simple averages
  - Lagged proportions
  - …
- Seasonality of data is important



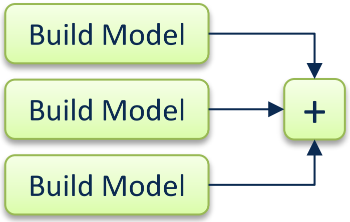
Combined strategy: mixture of past ratios and ratio one season ago

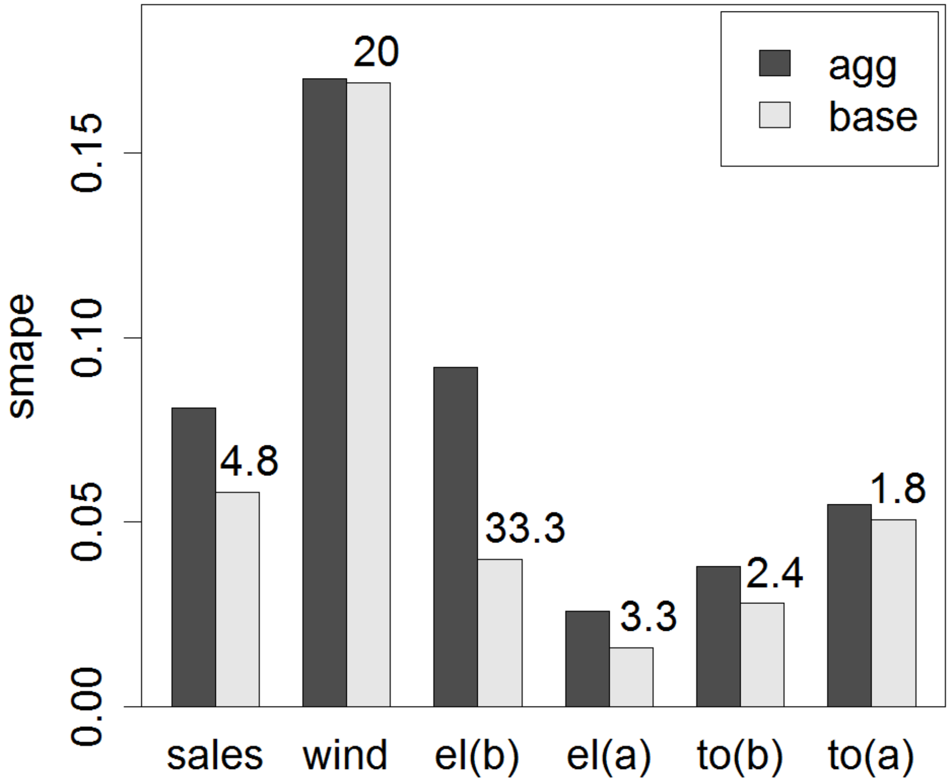


agg

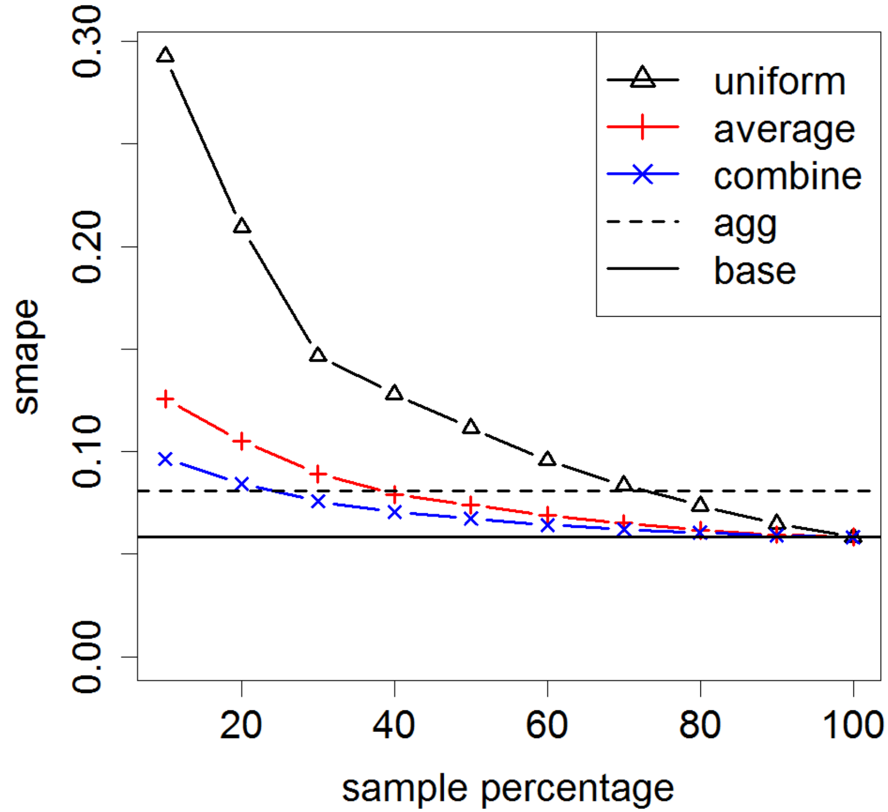

+
Build Model

base


Build Model
Build Model
Build Model
+

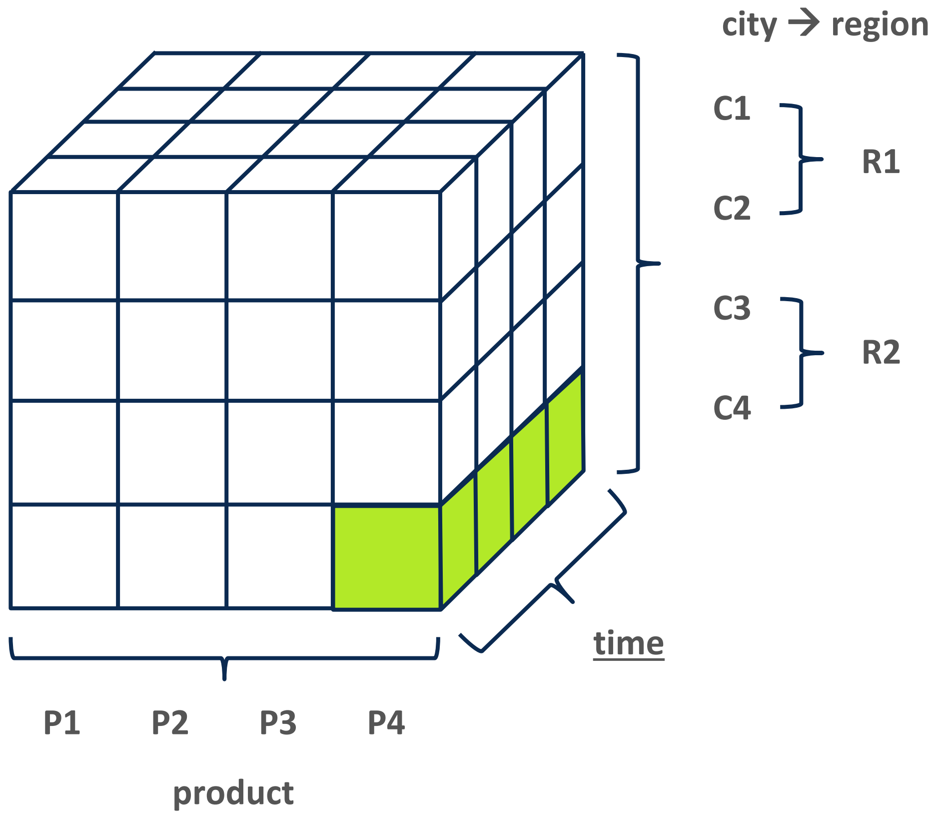

smape
0.00
0.05
0.10
0.15
agg
base
20
4.8
33.3
3.3
2.4
1.8
sales
wind
el(b)
el(a)
to(b)
to(a)

Aggregation - Sales

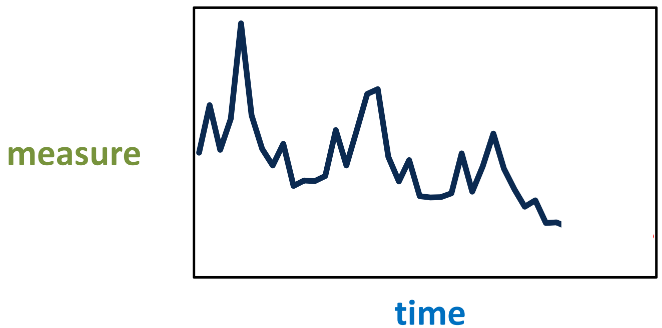


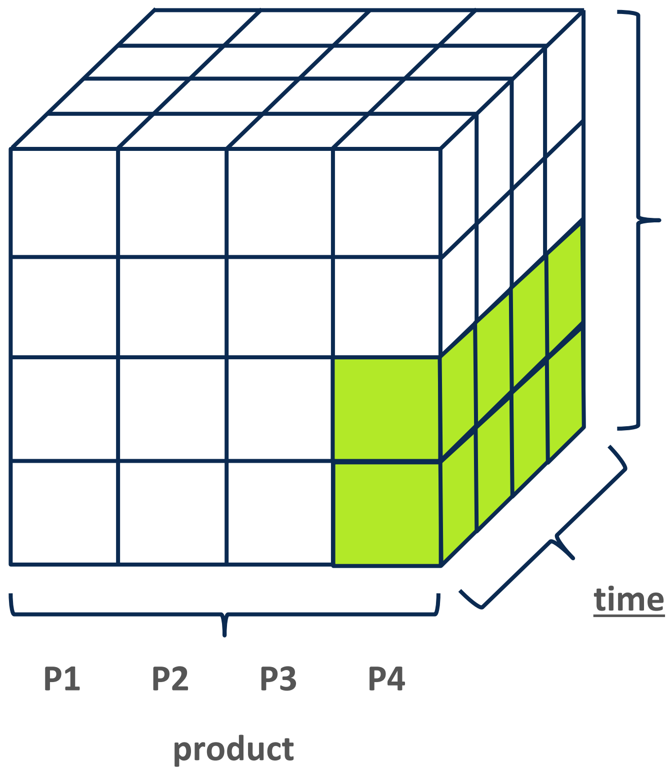
Model Configuration Advisor

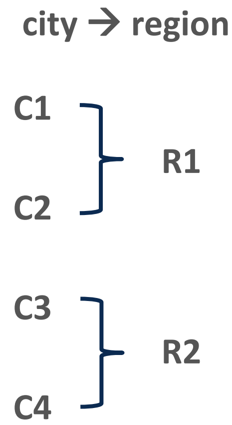
Forecasting the Data Cube



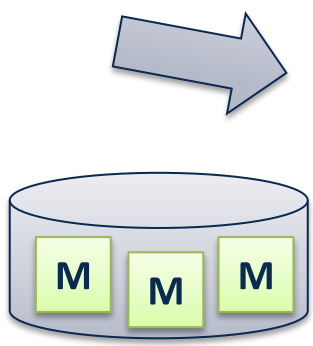

```
SELECT   time, measure
FROM     facts
WHERE    product = P4
AND      city = C4
AS OF    now() + 1 day
```

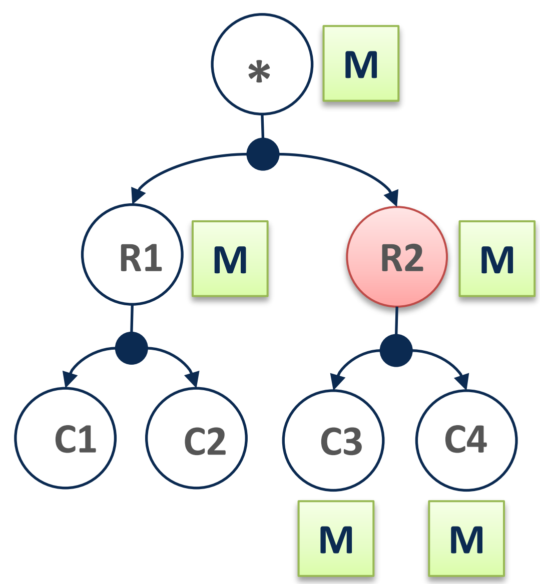




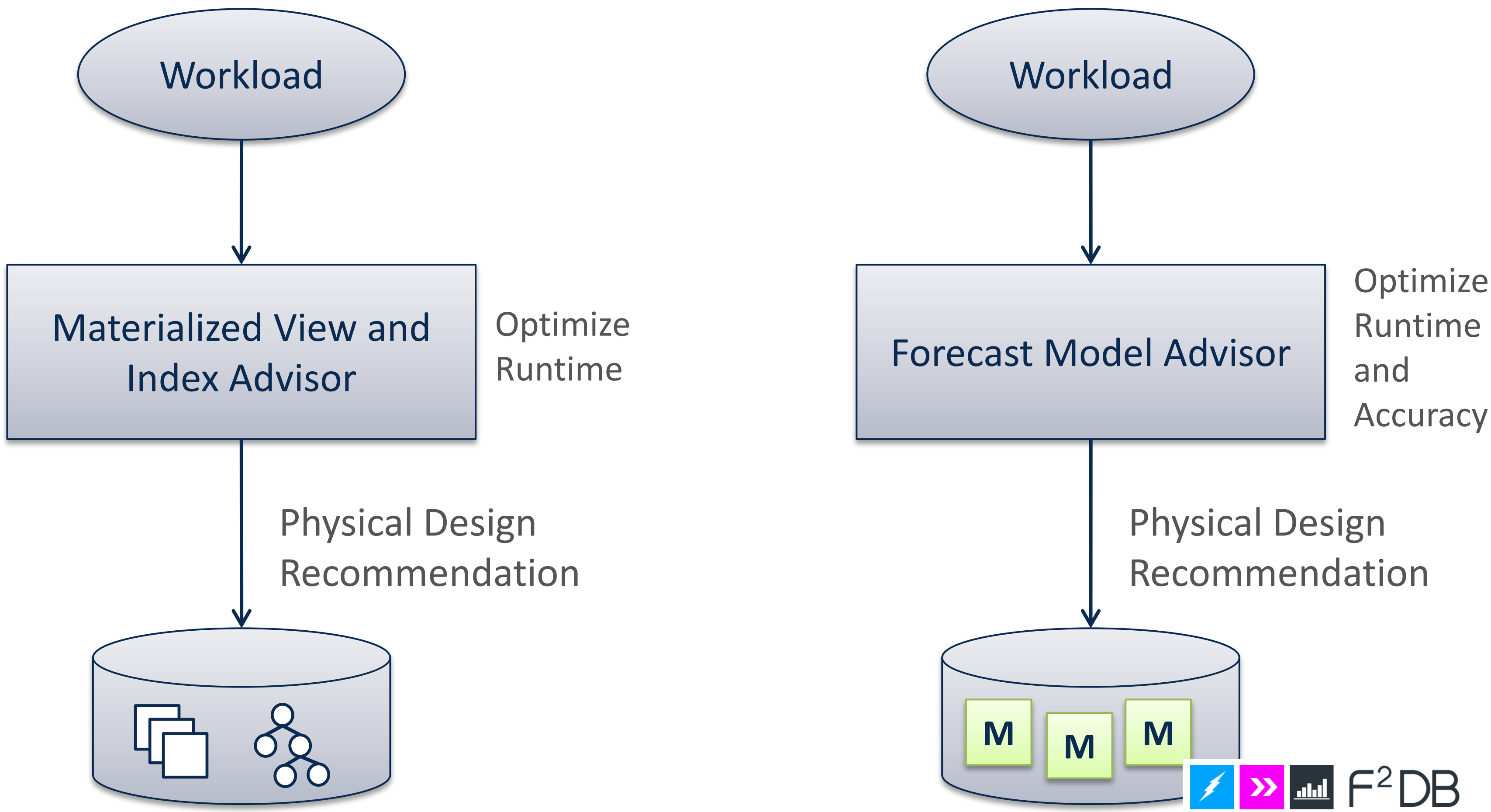



```
SELECT    time, SUM(measure)
FROM      facts
WHERE     product = P4
AND       region = R2
GROUP BY  time
AS OF     now() + 1 day
```

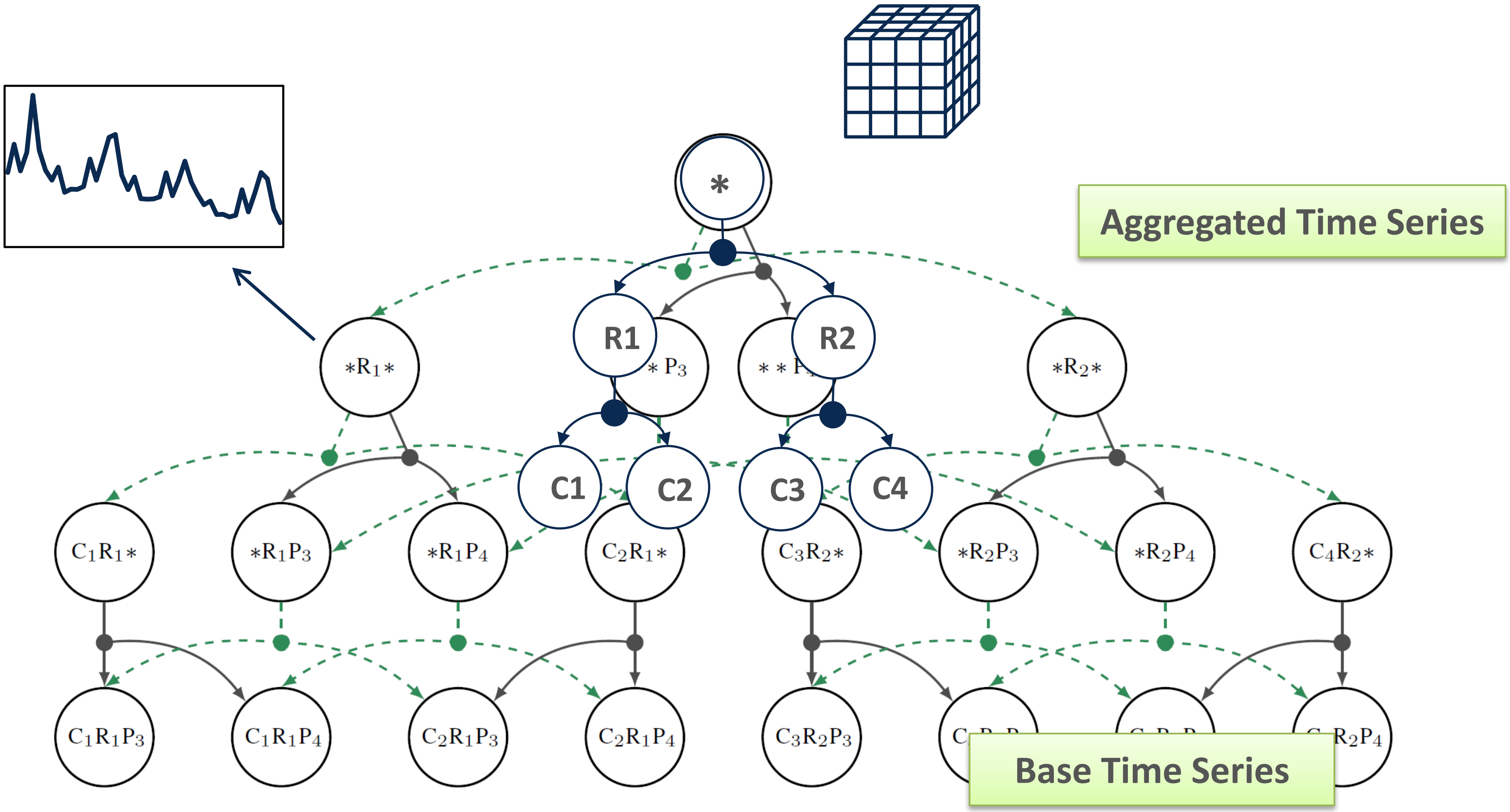


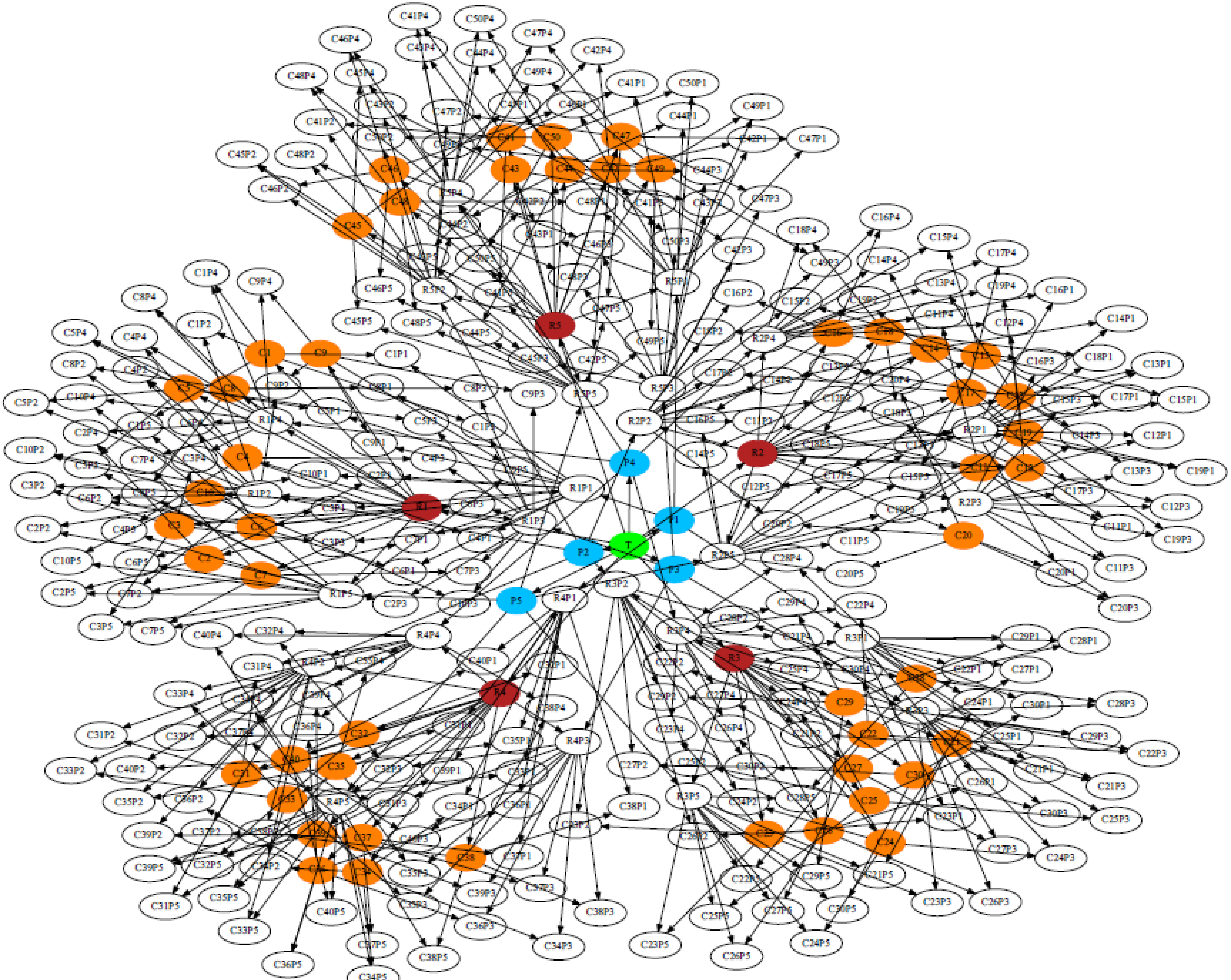


Traditional Database Systems *Statistical Database Systems*

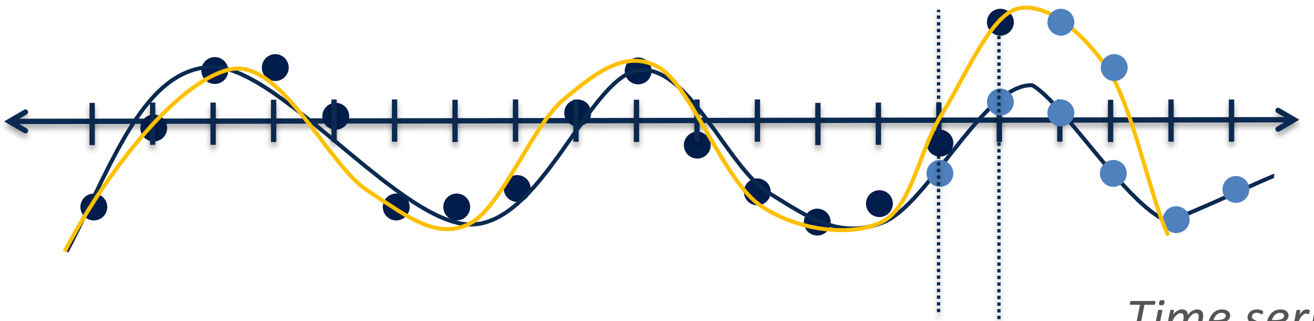


Conceptually, we organize the aggregation possibilities as a directed time series hyper graph





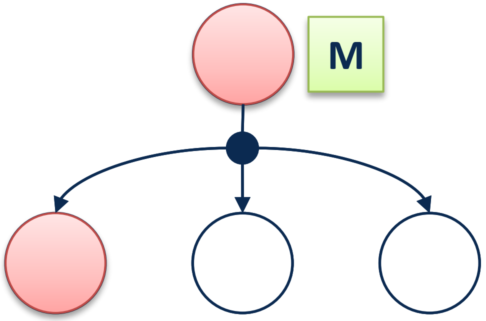
Each node (or time series) may be associated with a forecast model



Time series methods

- Exponential Smoothing
- ARIMA

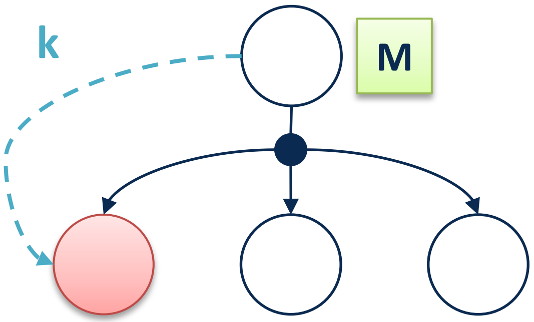
A query describes one or several nodes in the hyper graph

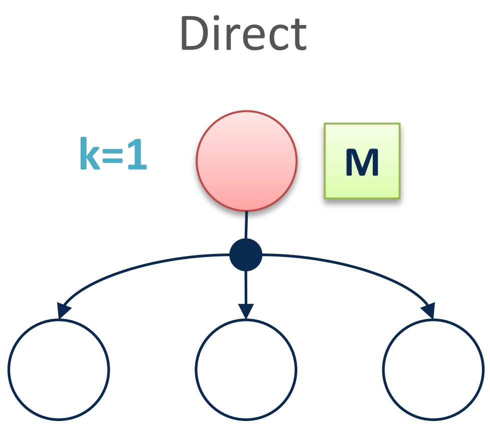


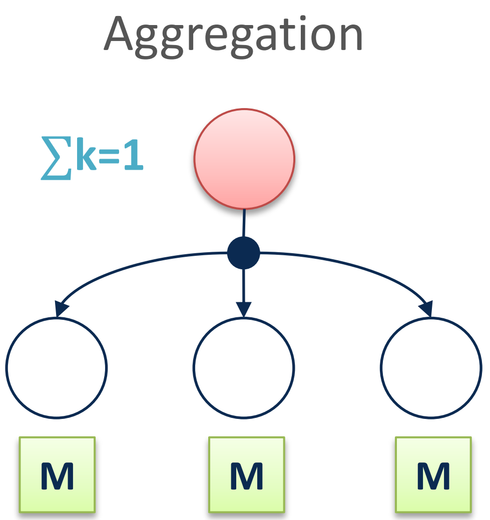
Forecast values of a node can be computed by any nodes in the graph

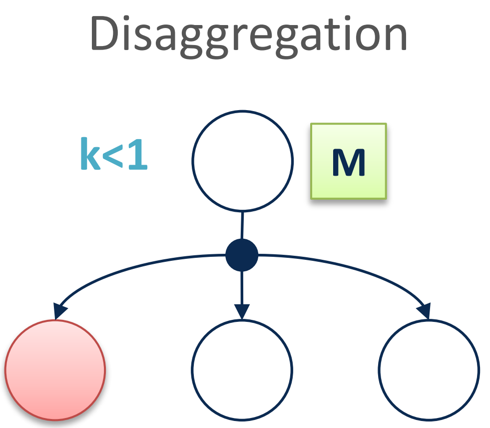
Derivation weight k

- Based on history of source and target time series





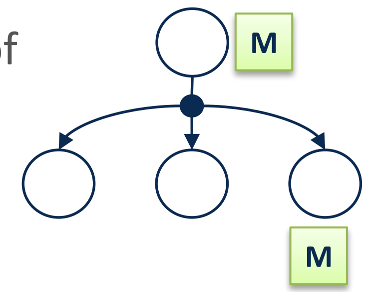




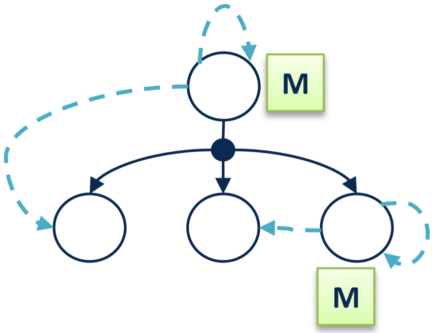
> Evaluation of Model Configurations

Model configuration

Assignment of models

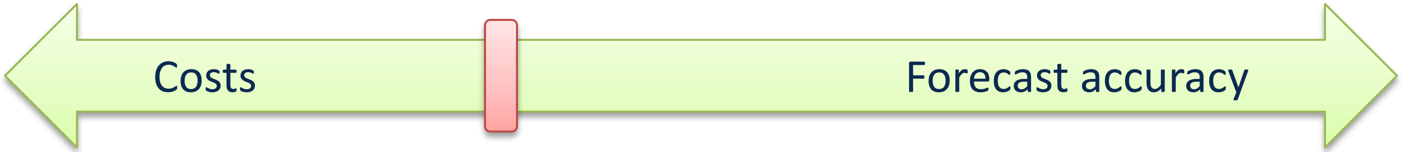


+

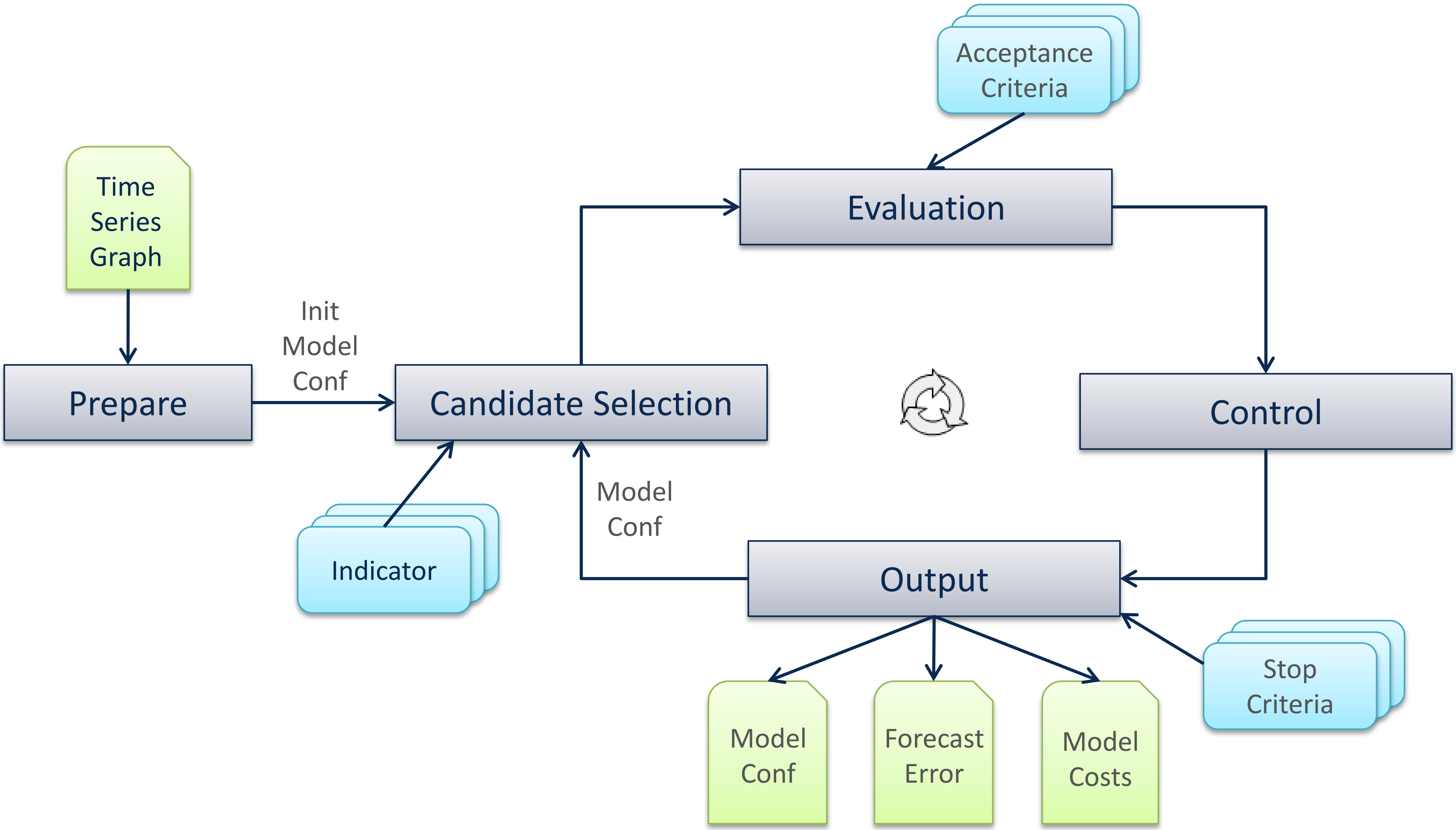


Assignment of derivation schemes

Configuration evaluation



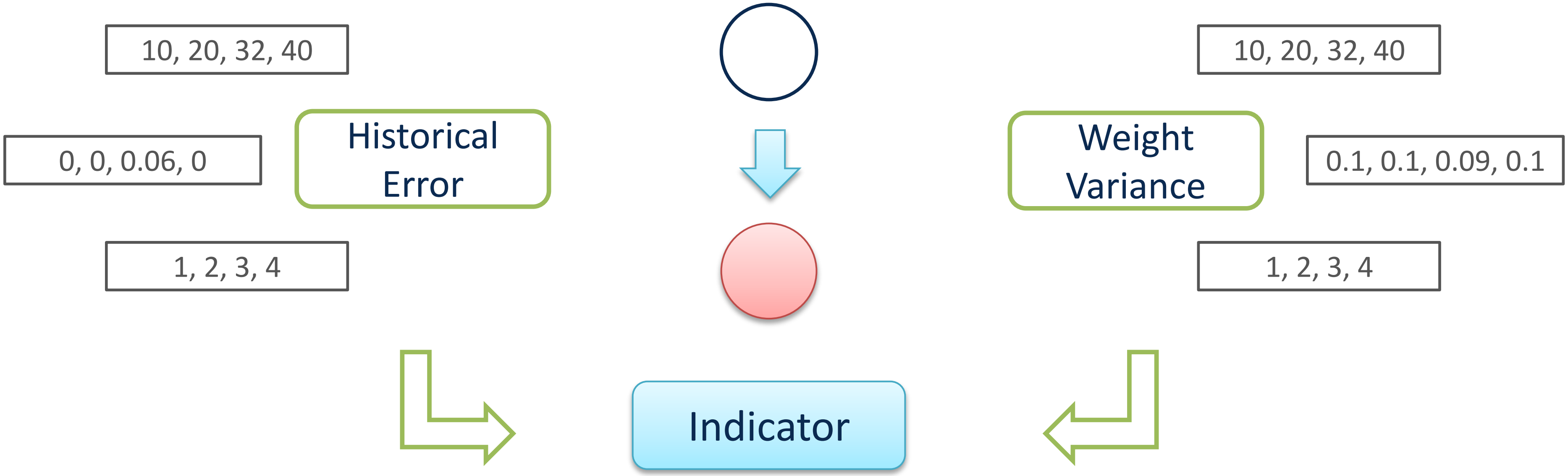
> Overview Configuration Advisor



> Indicators

Heuristically indicate the expected benefit of a model at a node (without building the model)

- Focus on time series relationships
- Measure to specify the derivation error between two nodes
- Low indicator value →low error (*good derivation*)
- High indicator value → high error (*poor derivation*)



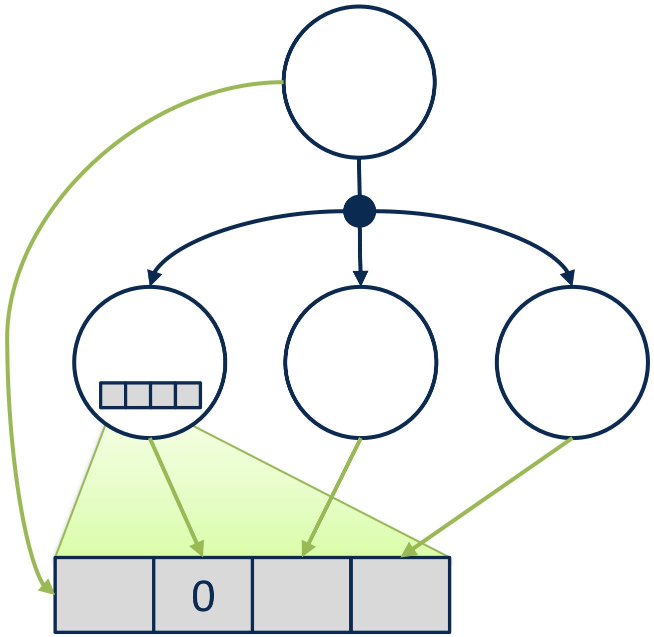
> Indicators

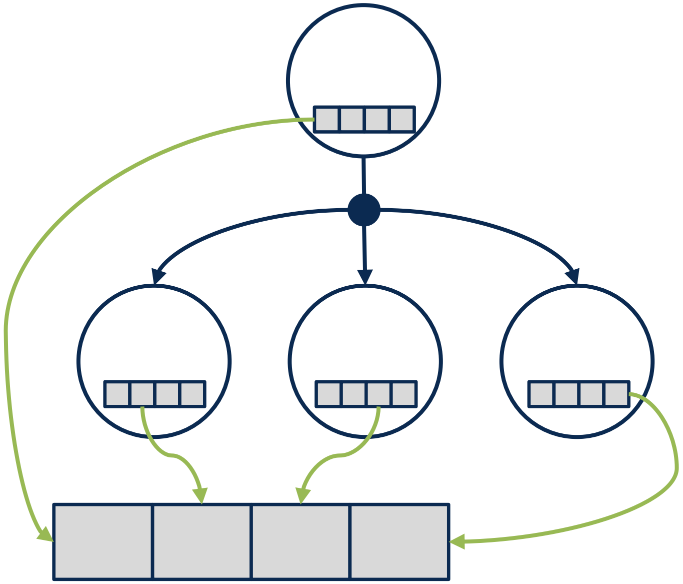
Local indicator arrays

→ *Derivation errors of one node*

Global indicator array

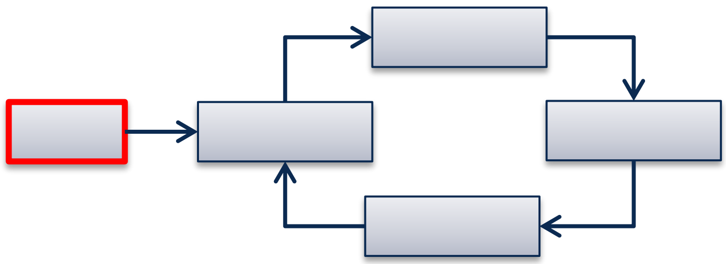
→ *Minimum over all local arrays*



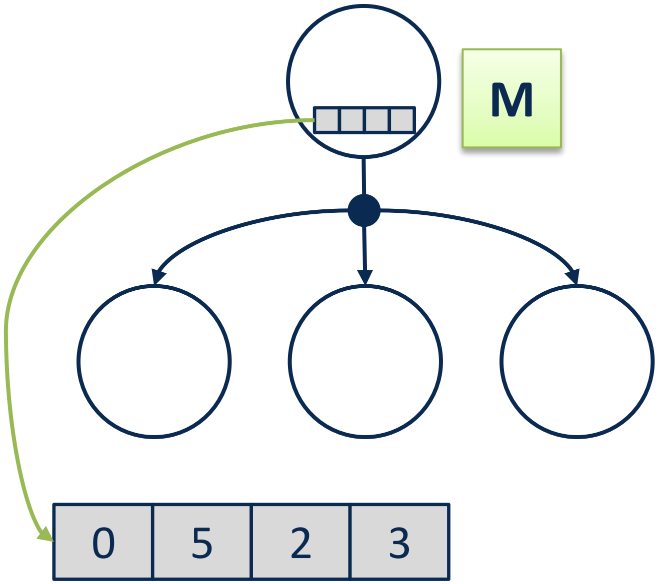


MIN

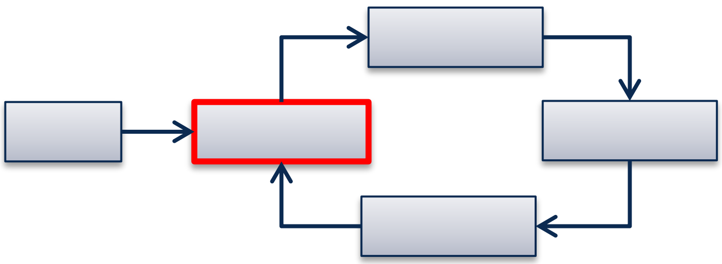
> Selection of Model Configurations



Select start configuration



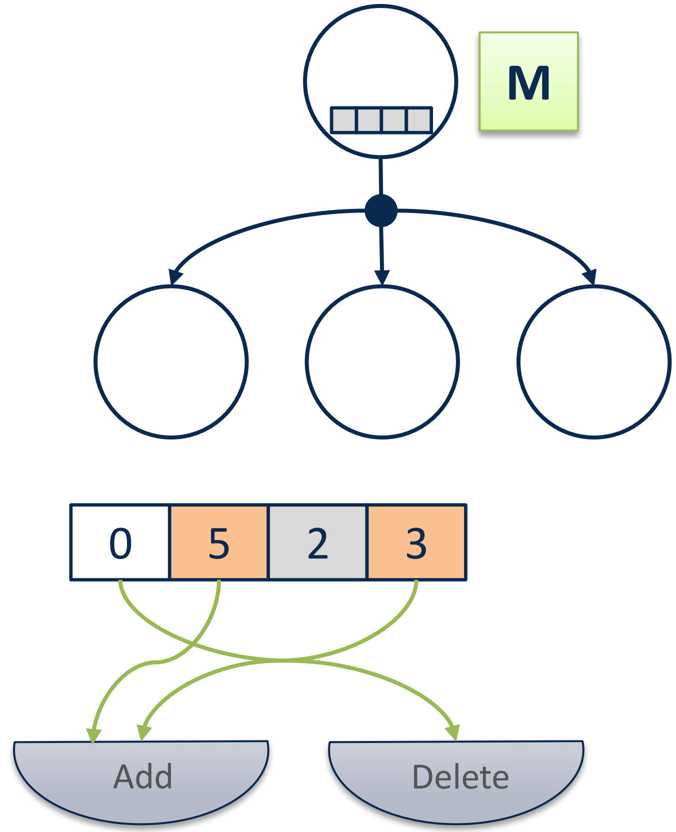
> Selection of Model Configurations



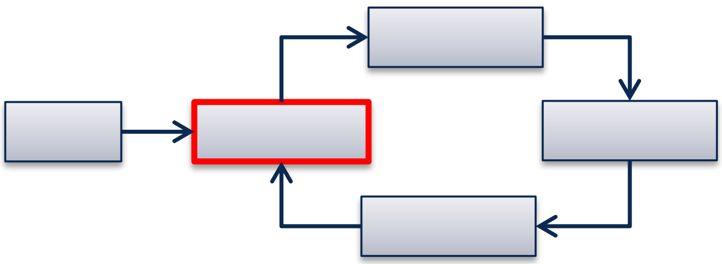
Select start configuration

Candidate selection

- Preselection



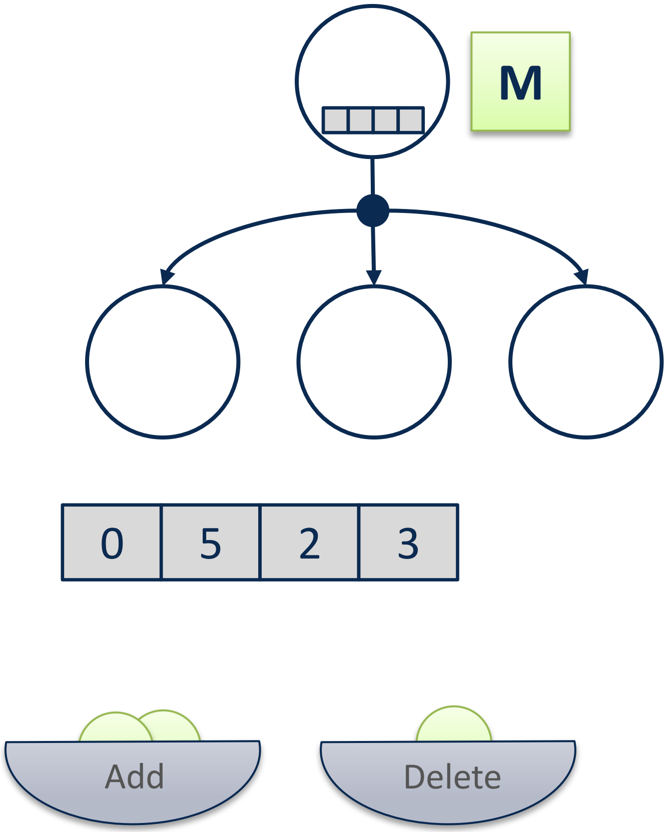
> Selection of Model Configurations



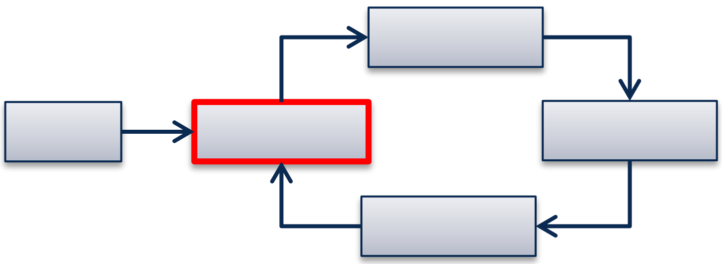
Select start configuration

Candidate selection

- Preselection



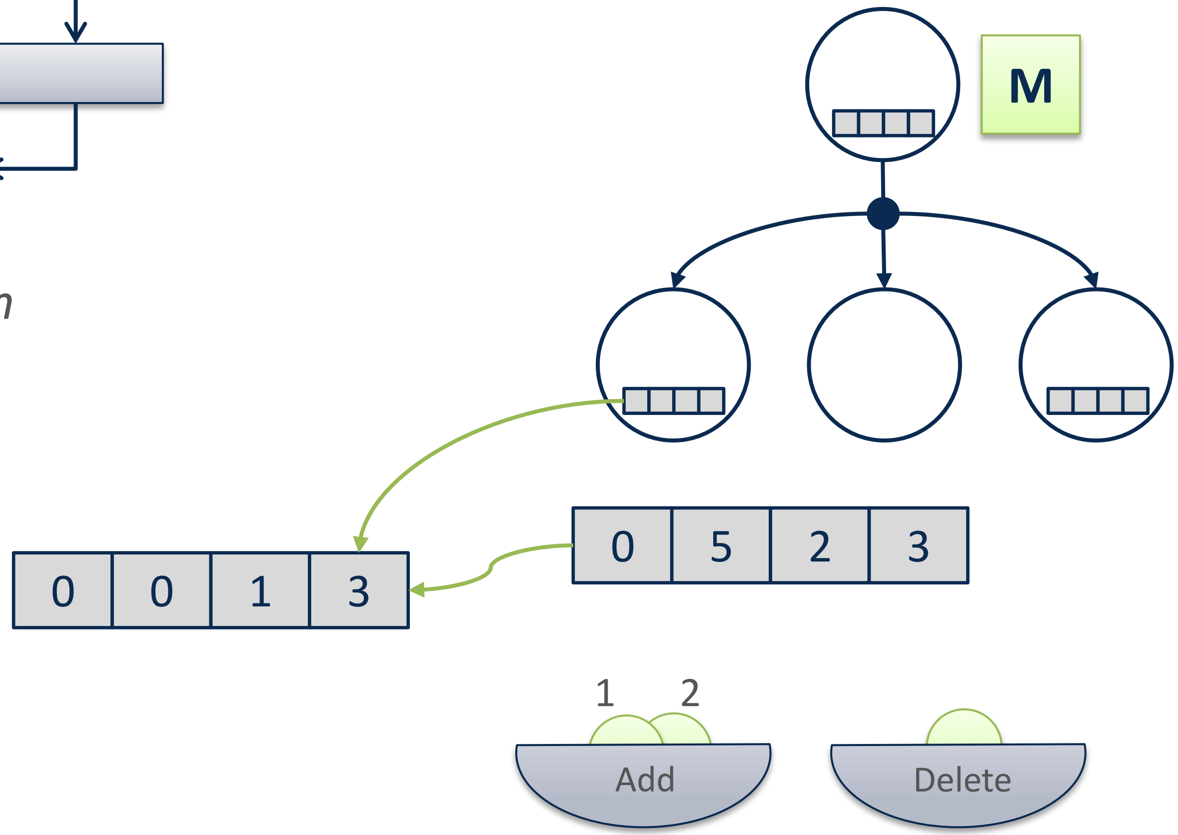
> Selection of Model Configurations



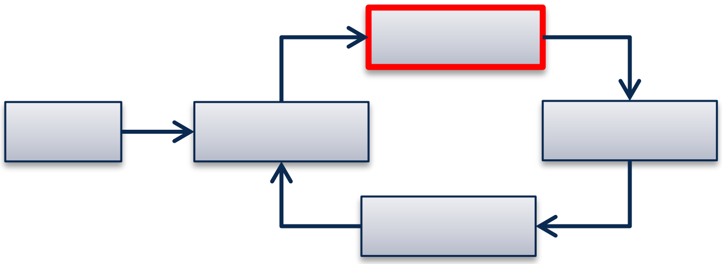
Select start configuration

Candidate selection

- Preselection
- Ranking



> Selection of Model Configurations



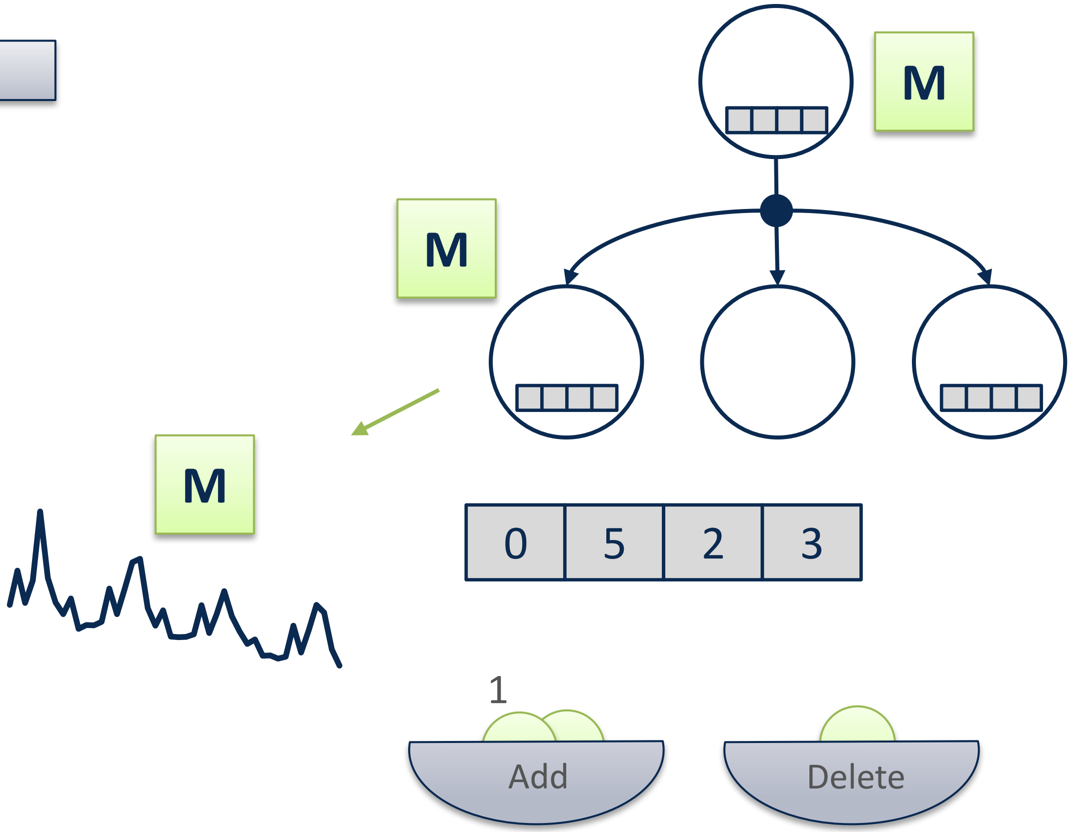
Select start configuration

Candidate selection

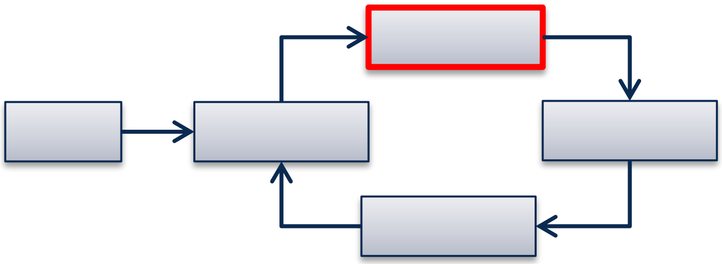
- Preselection
- Ranking

Evaluation

- Model creation
- Acceptance



> Selection of Model Configurations



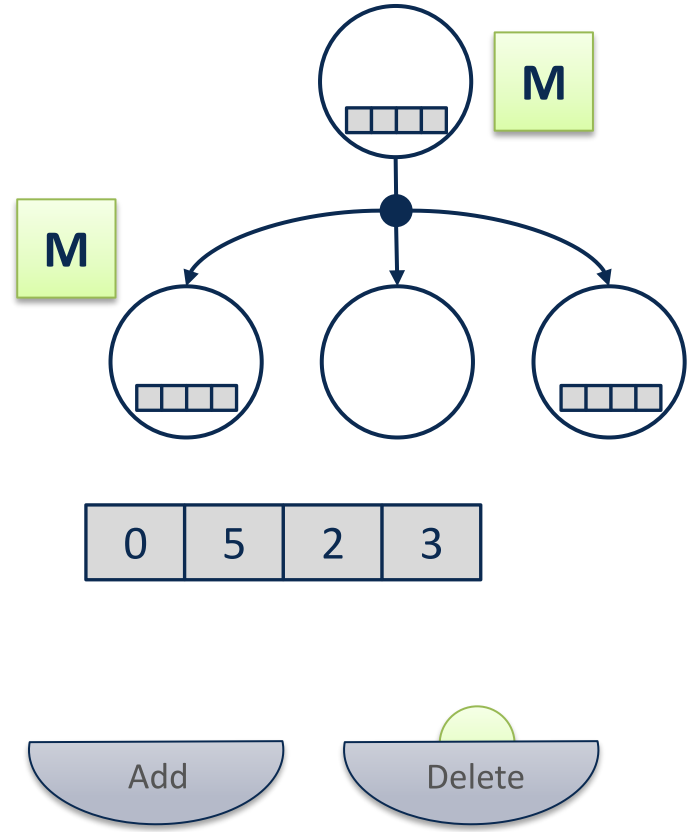
Select start configuration

Candidate selection

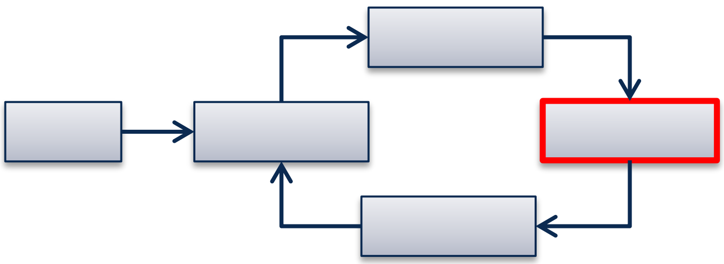
- Preselection
- Ranking

Evaluation

- Model creation
- Acceptance



> Selection of Model Configurations



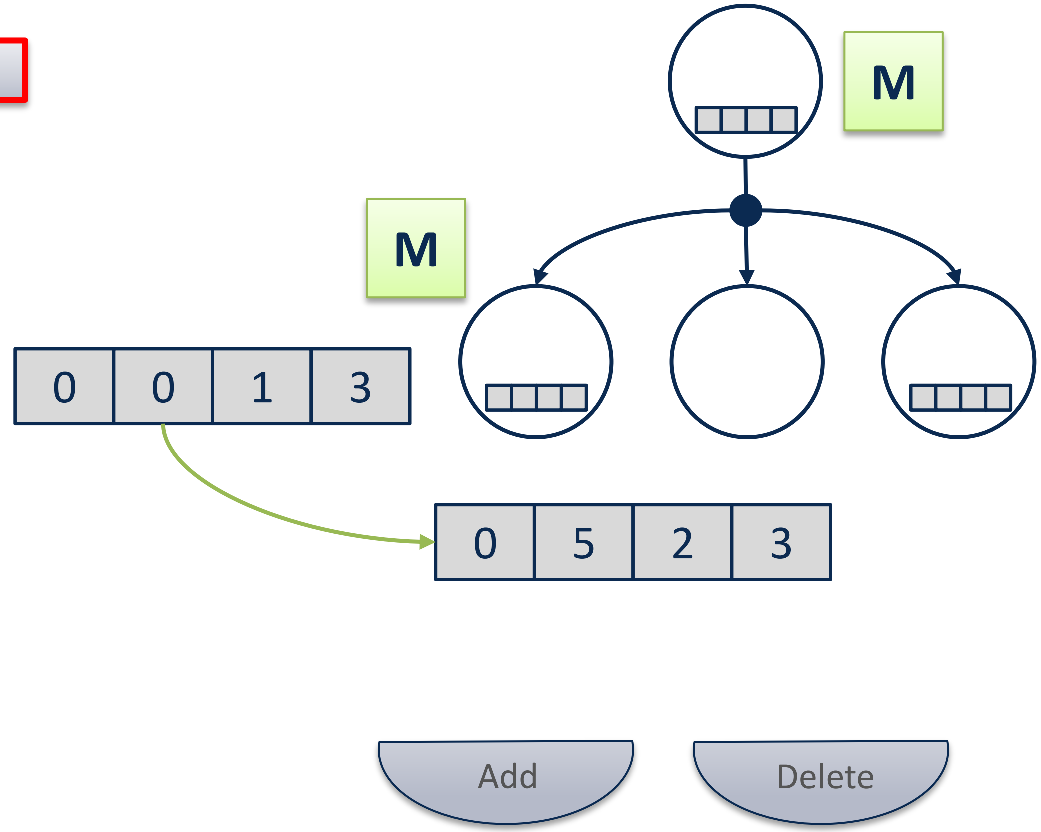
Select start configuration

Candidate selection

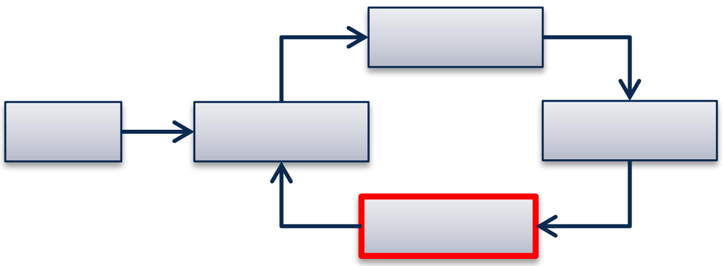
- Preselection
- Ranking

Evaluation

- Model creation
- Acceptance



> Selection of Model Configurations



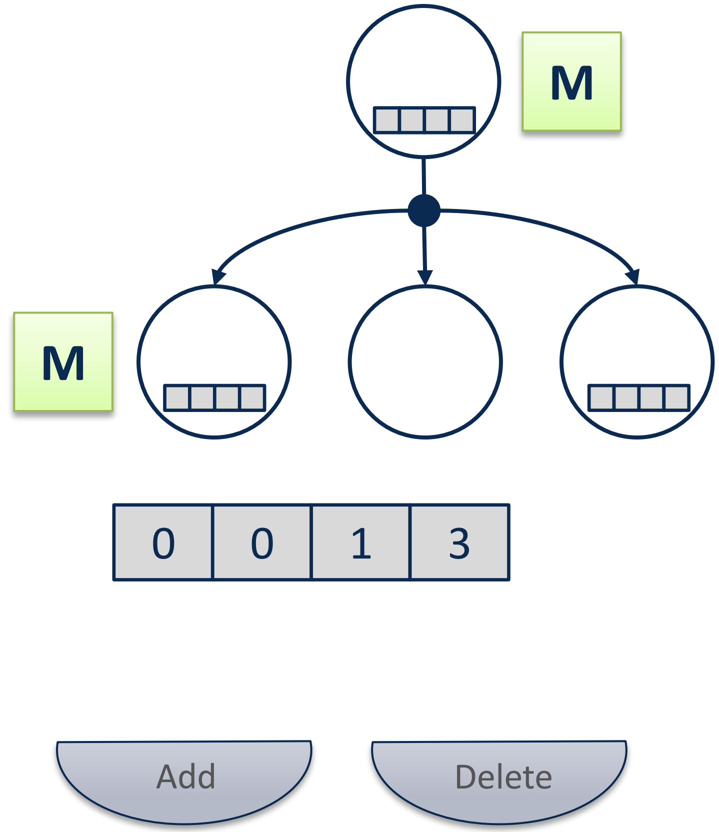
Select start configuration

Candidate selection

- Preselection
- Ranking

Evaluation

- Model creation
- Acceptance

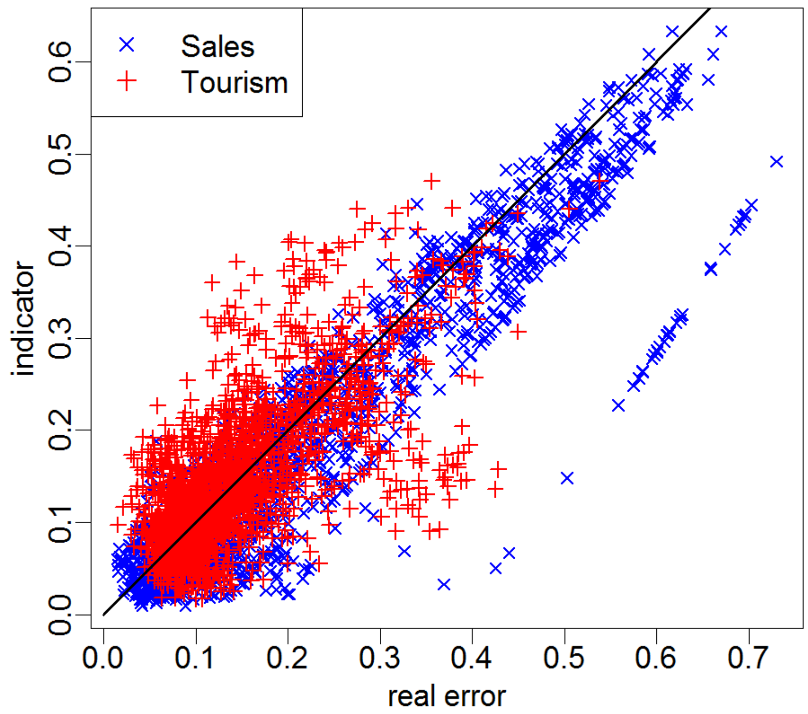


Model costs: 2

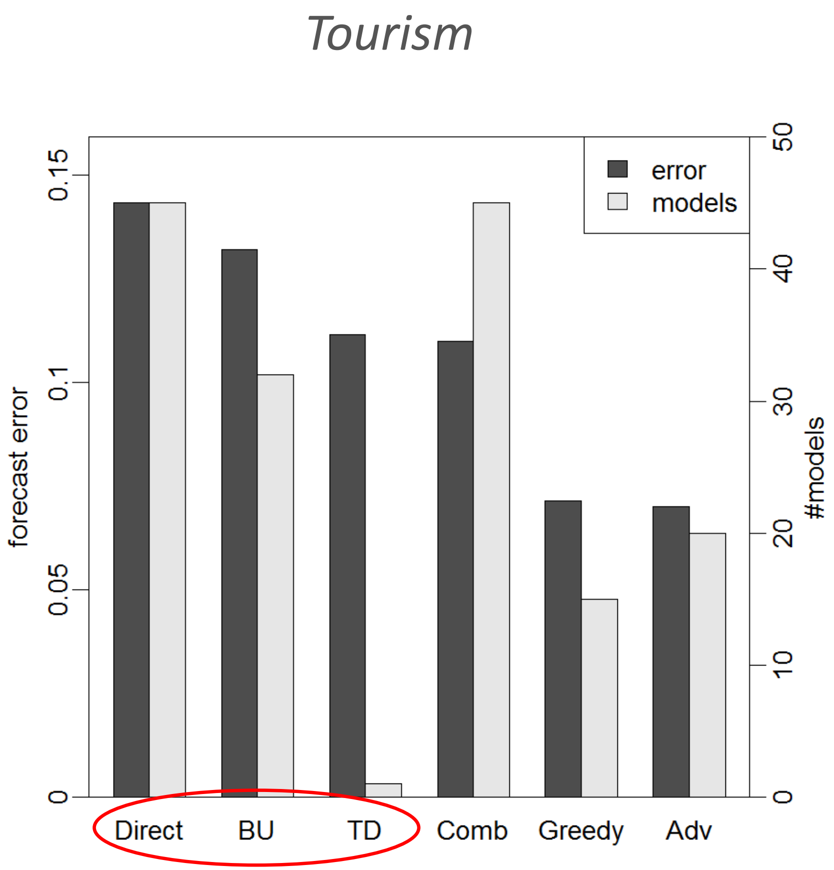
Forecast error: 10 %

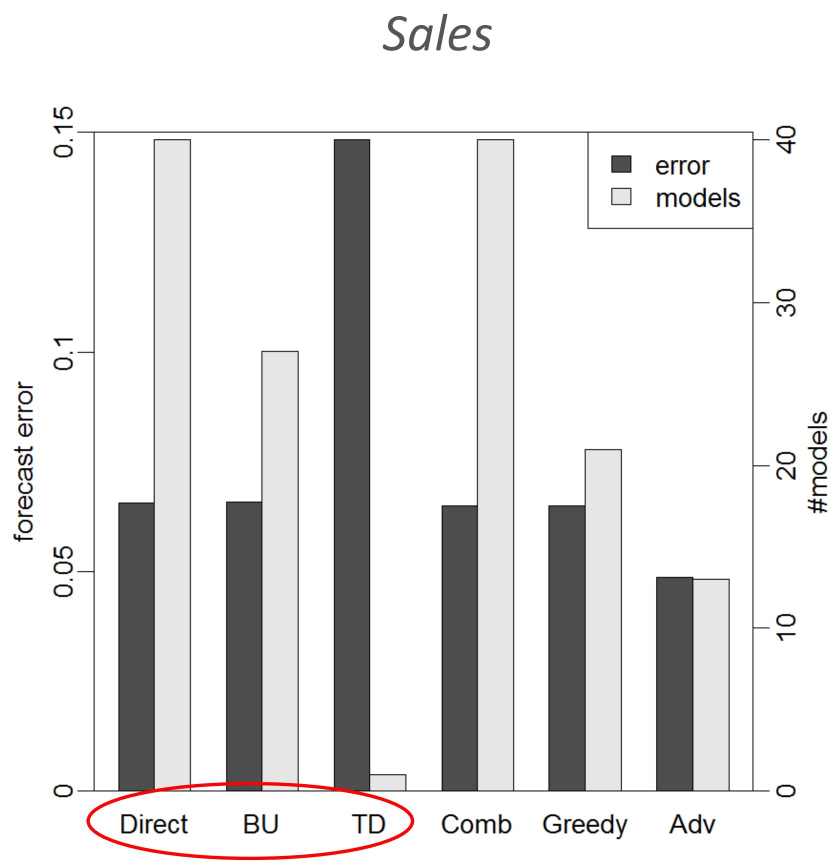
> Indicator Accuracy

Correlation between indicators and real forecast error

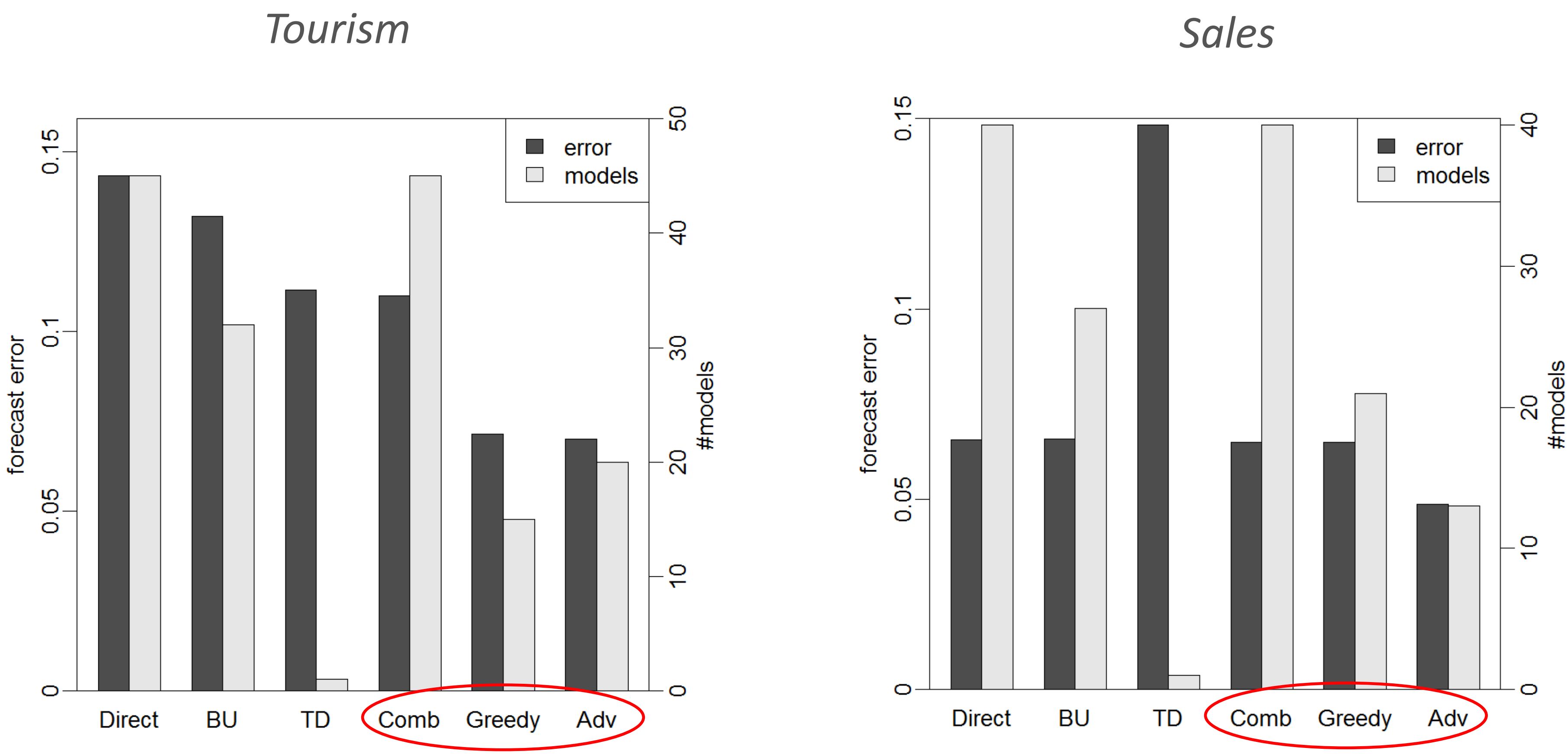


> Accuracy Analysis



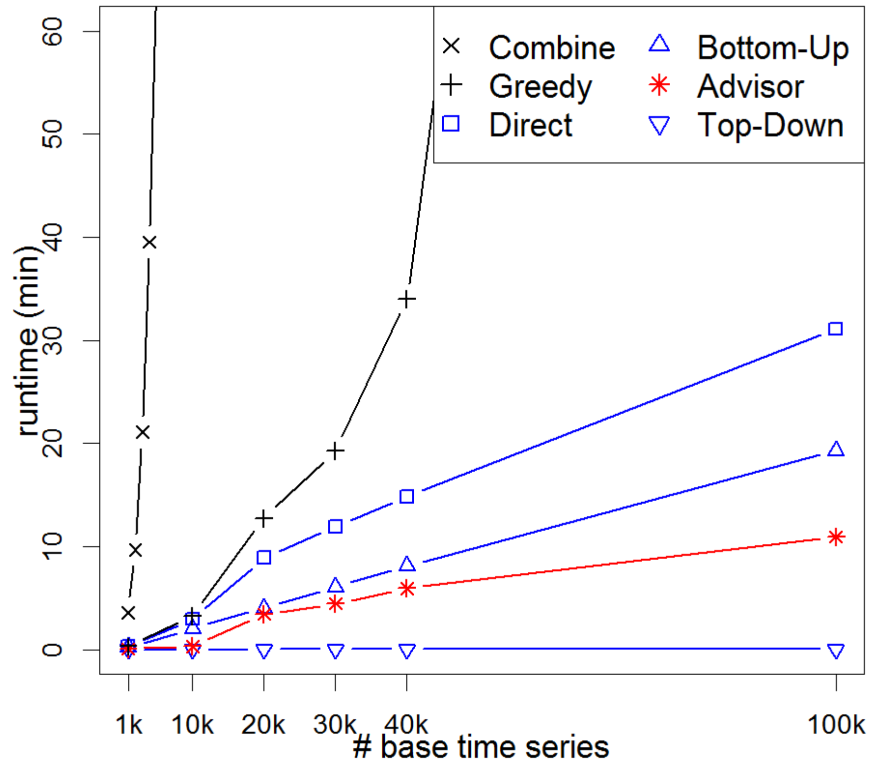


Static approaches – data independent



Dynamic approaches – empirical selection

Scalability



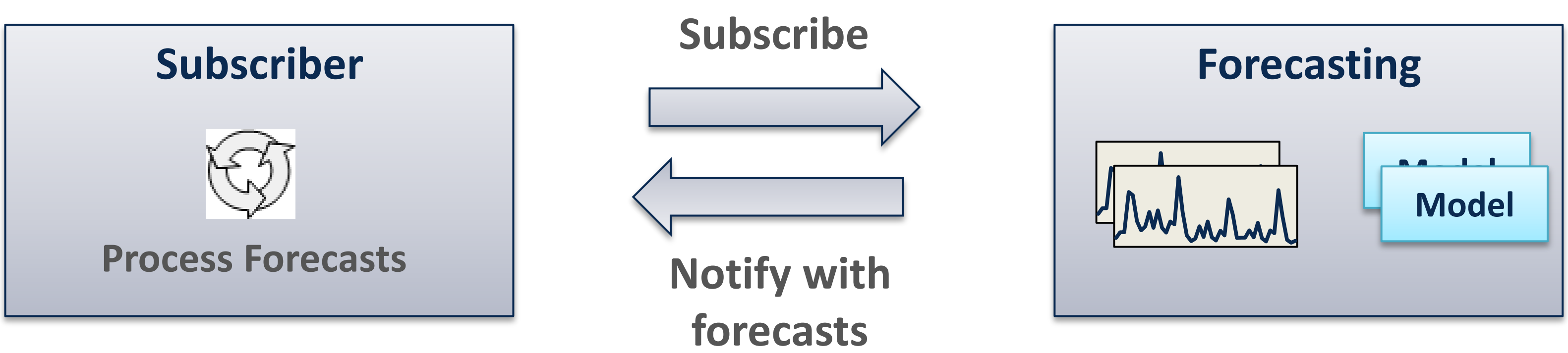
Subscription-Based Forecast Queries

Energy data management systems

- Provide stable energy supply while including larger amounts of renewable energy
- **Continously** require forecasts of energy demand and supply

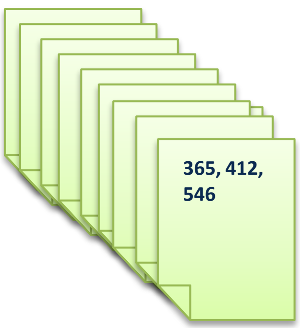
MIRABEL energy
Balancing energy supply and demand

Subscription-based forecast queries



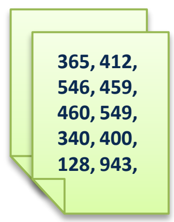
When to notify subscriber?

After each new real value …



→ Many notifications
→ **High subscriber costs**

As less as possible …



→ Long messages
→ Low accuracy
    → Resend messages
→ **High subscriber costs**

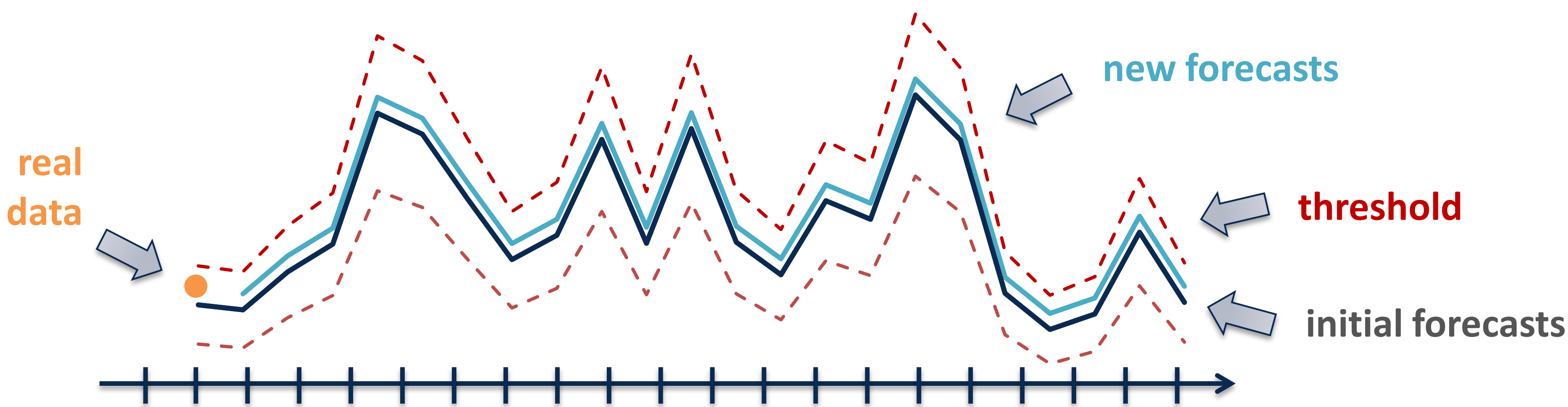
Optimal …

→ Reduce number of notifications and notification length

Parameters

- Time series description
- Minimum continous forecast horizon
- Accuracy threshold

```
SELECT    datetime, production
FROM      ts_powerproduction
WHERE     type = „wind“
FORECAST  3 hours
THRESHOLD 0.1
```



> Processing Model

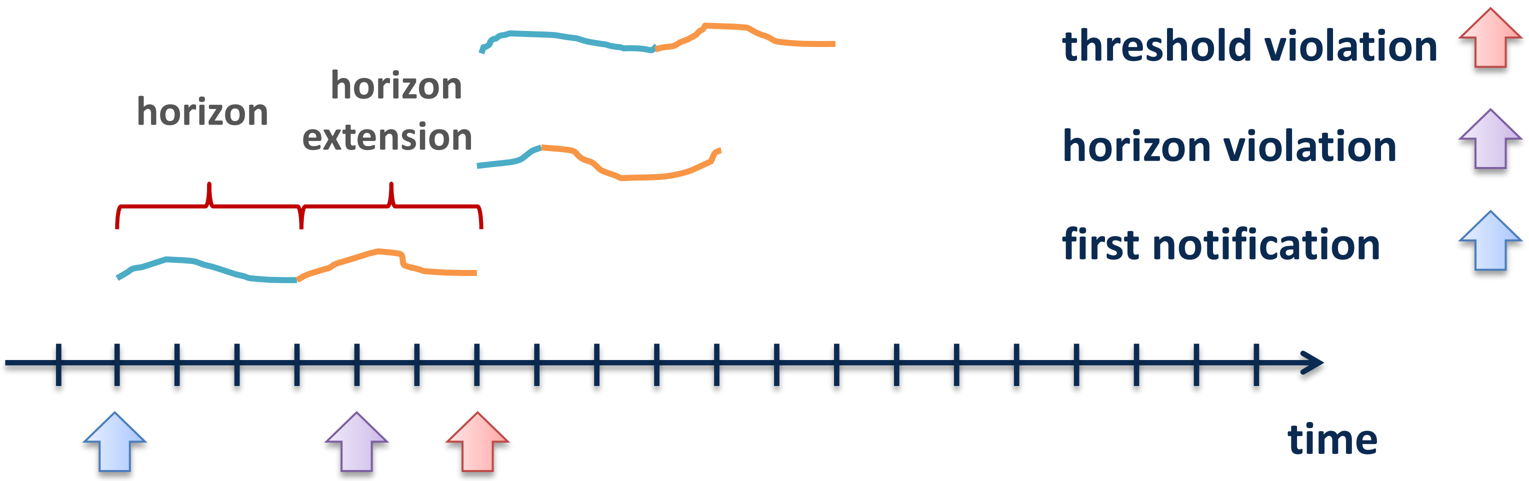
```
SELECT      datetime, production
FROM        ts_powerproduction
WHERE       type = „wind“
FORECAST    3 hours
THRESHOLD   0.1
```

Horizon Violation

- When? Subscriber has less than minimum horizon
- What? Send missing values + horizon extension

Threshold Violation

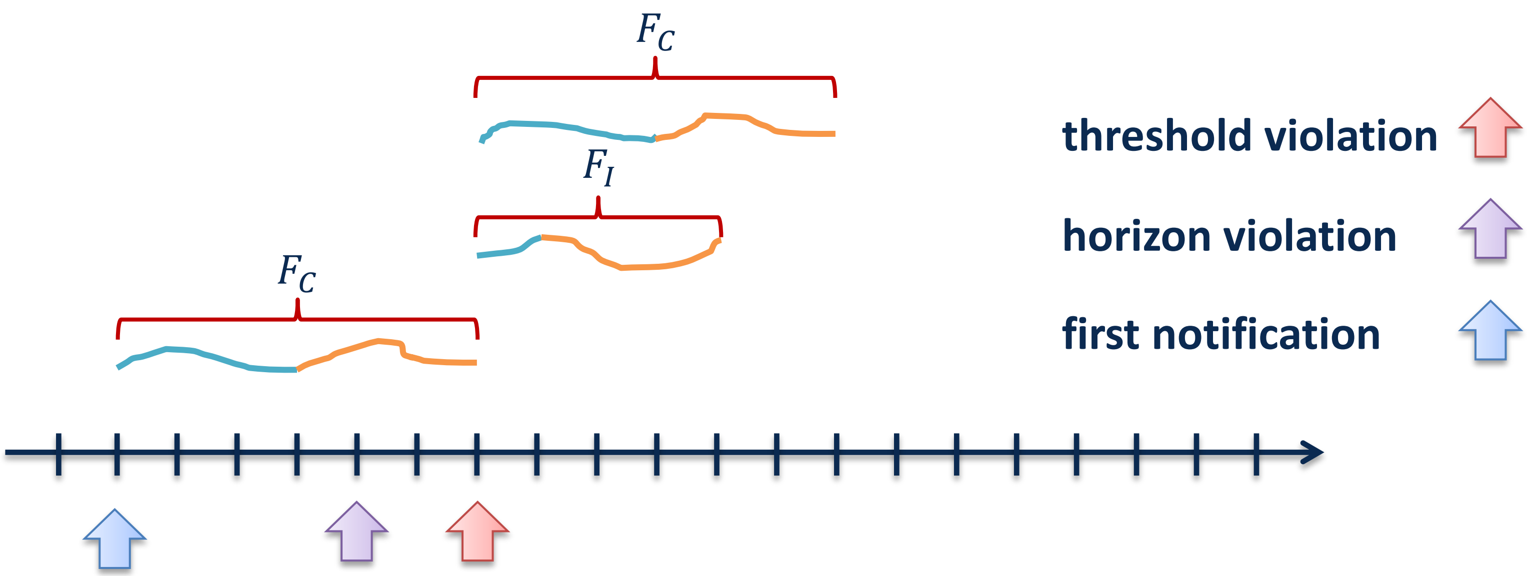
- When? Threshold is violated
- What? Resend all values



> Subscriber Cost Model

Processing costs of the subscriber

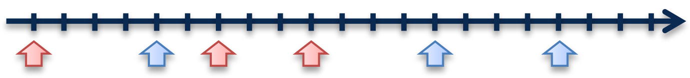
- Analytically known or learned function
- Depend on the forecast horizon
- Complete costs F_C
  - Complete restart of processing
- Incremental costs F_I
  - Processing of additional values



> Optimization Goal

Assume we know …

- The sequence of threshold + horizon violations
- The subscriber cost functions F_C and F_I

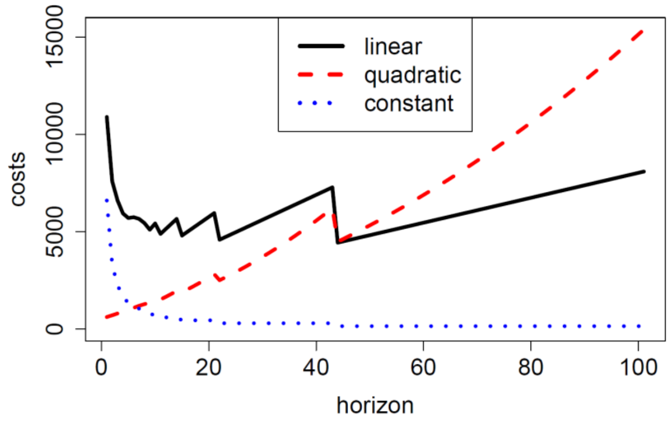


Subscriber costs over subscription lifetime

- Sum over all F_C and F_I

Optimization Goal

- Find forecast horizon that minimizes total costs
- Depends on subscriber cost function
- Depends on forecast accuracy



How to get threshold violations?

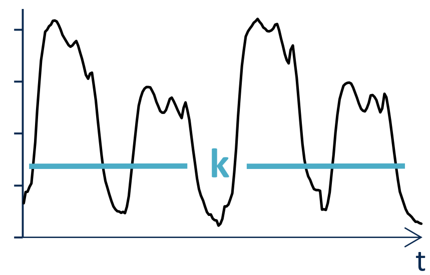
⇨ Analyze past to predict future

> Computation Approaches

Core Idea: Calculate best forecast horizon using our cost model on the time series history

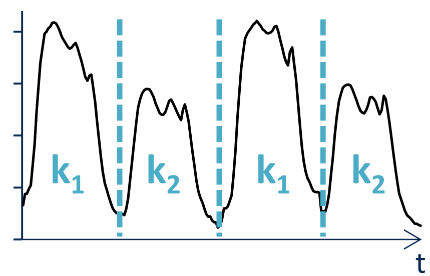
Offline – Static

- One forecast horizon over whole lifetime



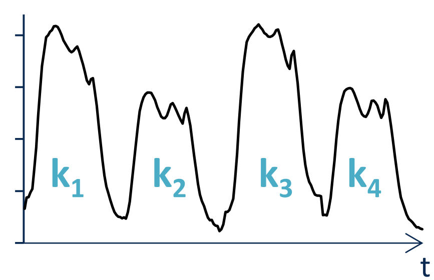
Offline – Dynamic

- Adapts to periodic changes of time series accuracy
- Sequence of forecast horizons for time slices



Online

- Adapts to arbitrary changes in data or cost functions
- Continously adapts forecast horizon



> Experimental Setting

Real-world energy demand and supply data sets

- National Energy Demand
- Household Energy Demand
- National Wind Supply

Subscriber cost functions

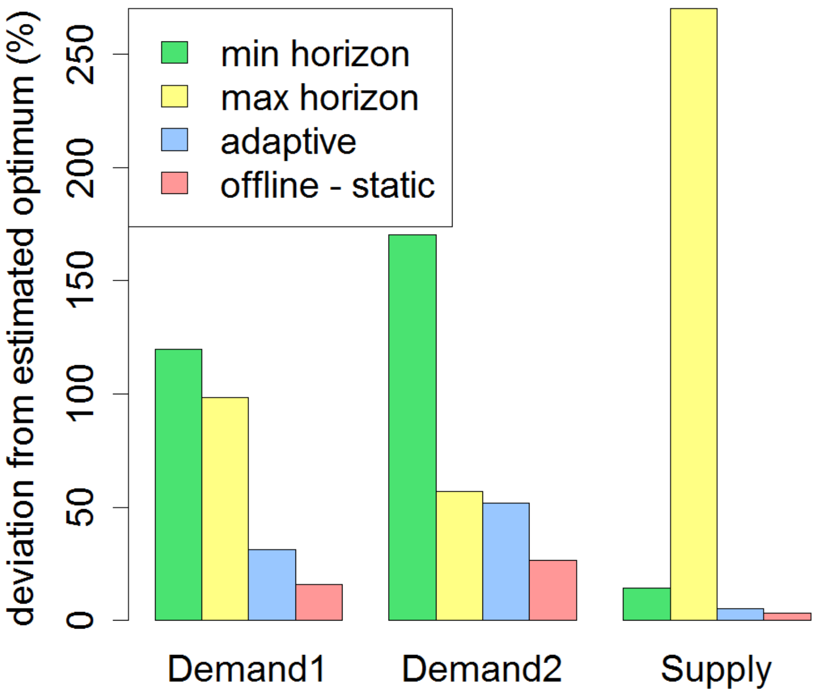
- Synthetic linear function
- Real world cost function (obtained from MIRABEL)

Forecast Methods

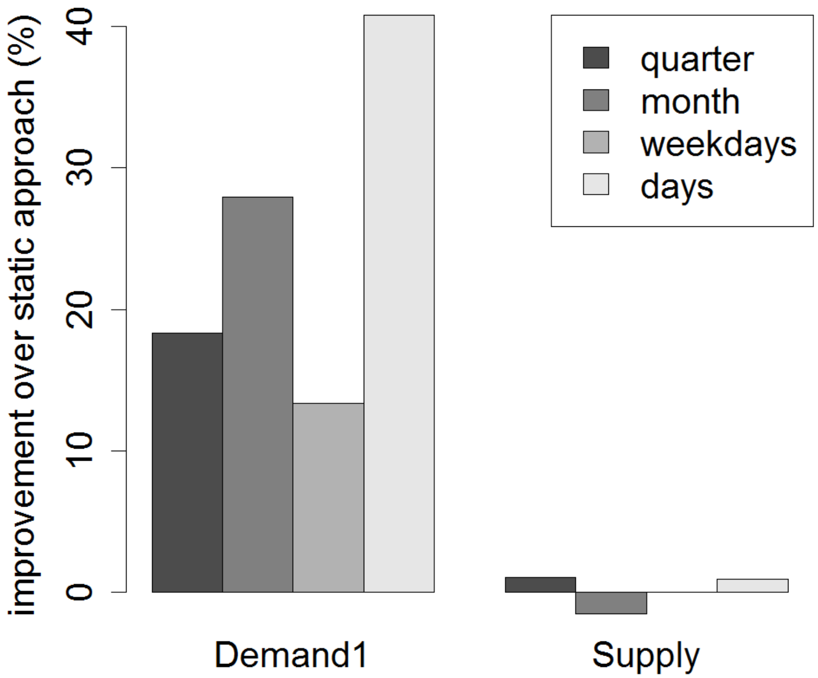
- Tailor-made for short-term energy forecasting
- Extension of exponential smoothing

Comparison of Approaches

- Fixed subscription parameters and linear cost function



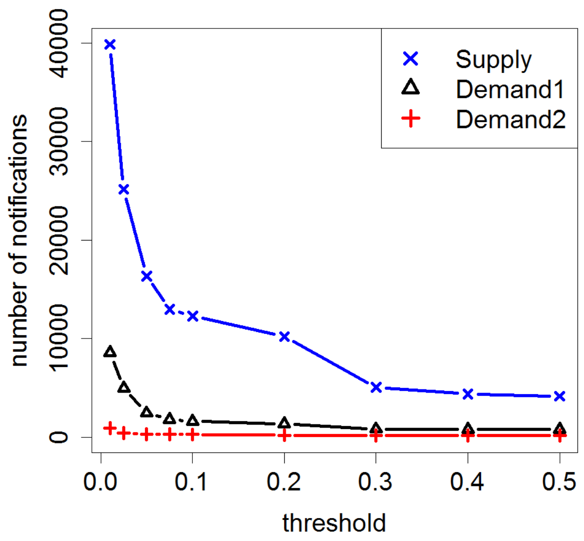
Evaluation of Time Slice Approach



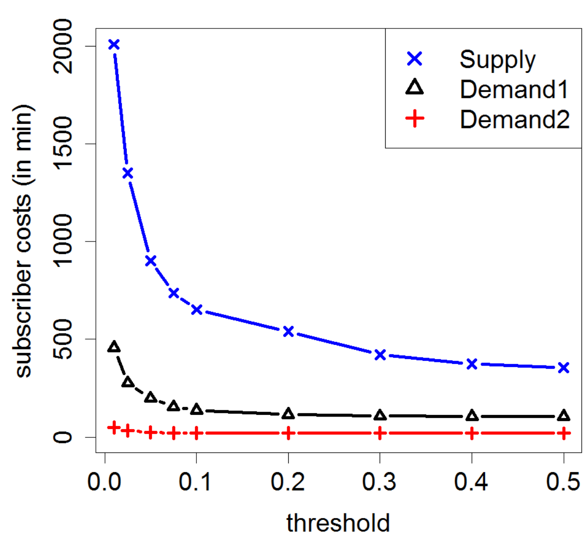
Influence of subscriber threshold

- Relationship between number of notifications, subscriber costs and runtime
- Real-world cost function

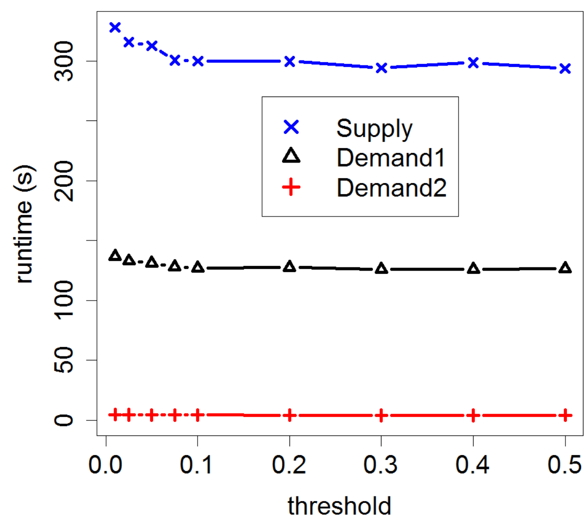
number of notifications



subscriber costs

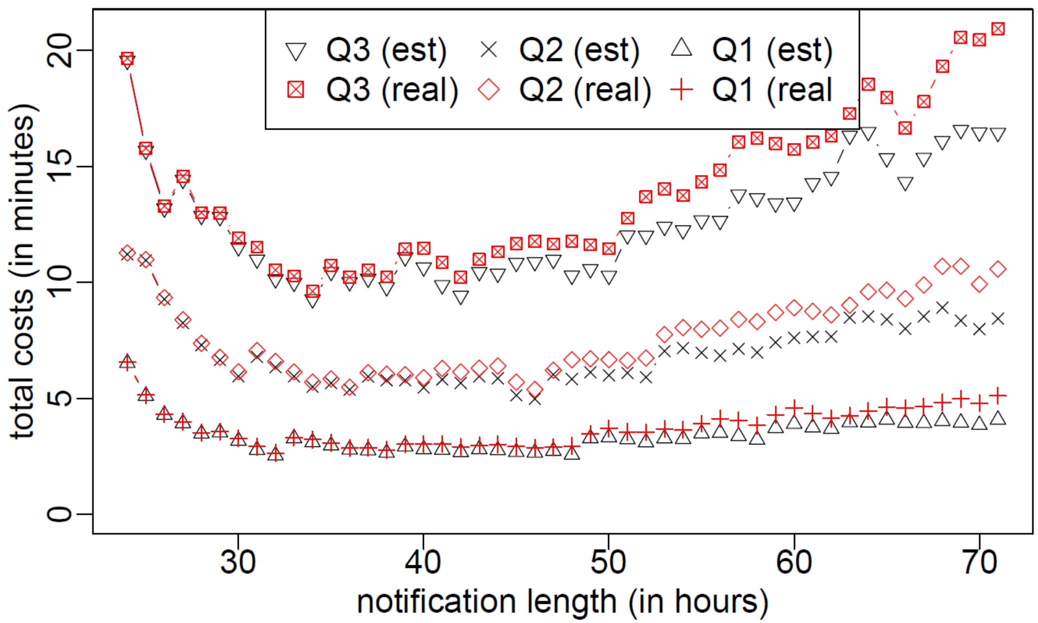


runtime



Cost Model Validation

- Estimated costs vs. real costs
- Real-world cost function
- Queries with increasing complexity (Q1, Q2, Q3)



Beyond Forecast Models

Towards a Model-based Database System

> FIRA^2 - Approach

Forecast

- Base approach
- Predict missing data from history
- Requires no known data

Impute

- Exploit local patterns
- Infer missing data from similar units
- Requires adequate set of known data

Refine

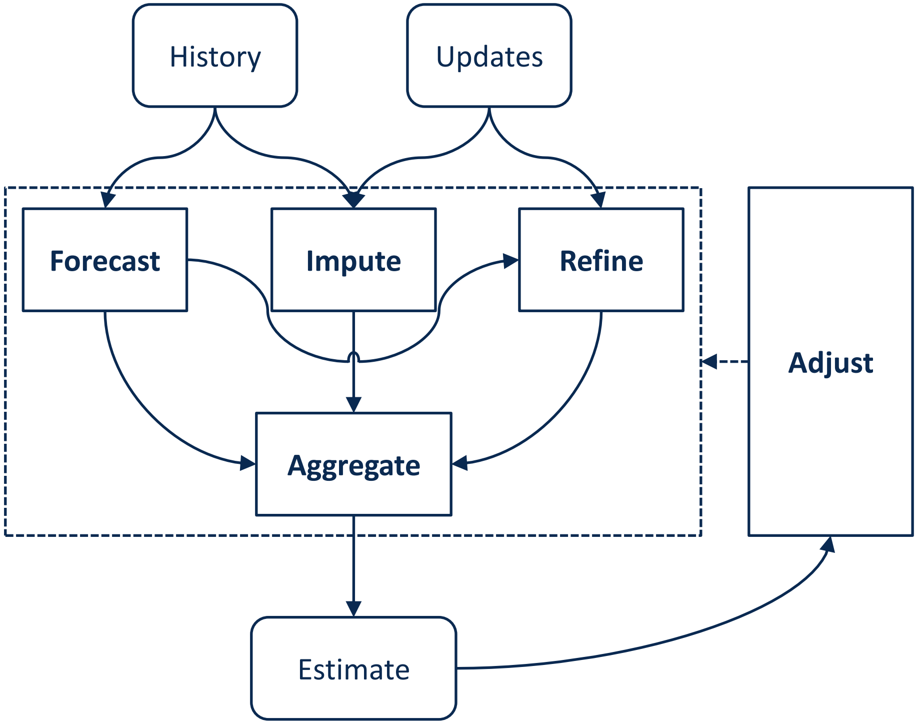
- Detect local and global shifts
- Infer error → yields new forecasts
- Requires few known data

Aggregate

- Calculate aggregate (e.g. report)

Adjust

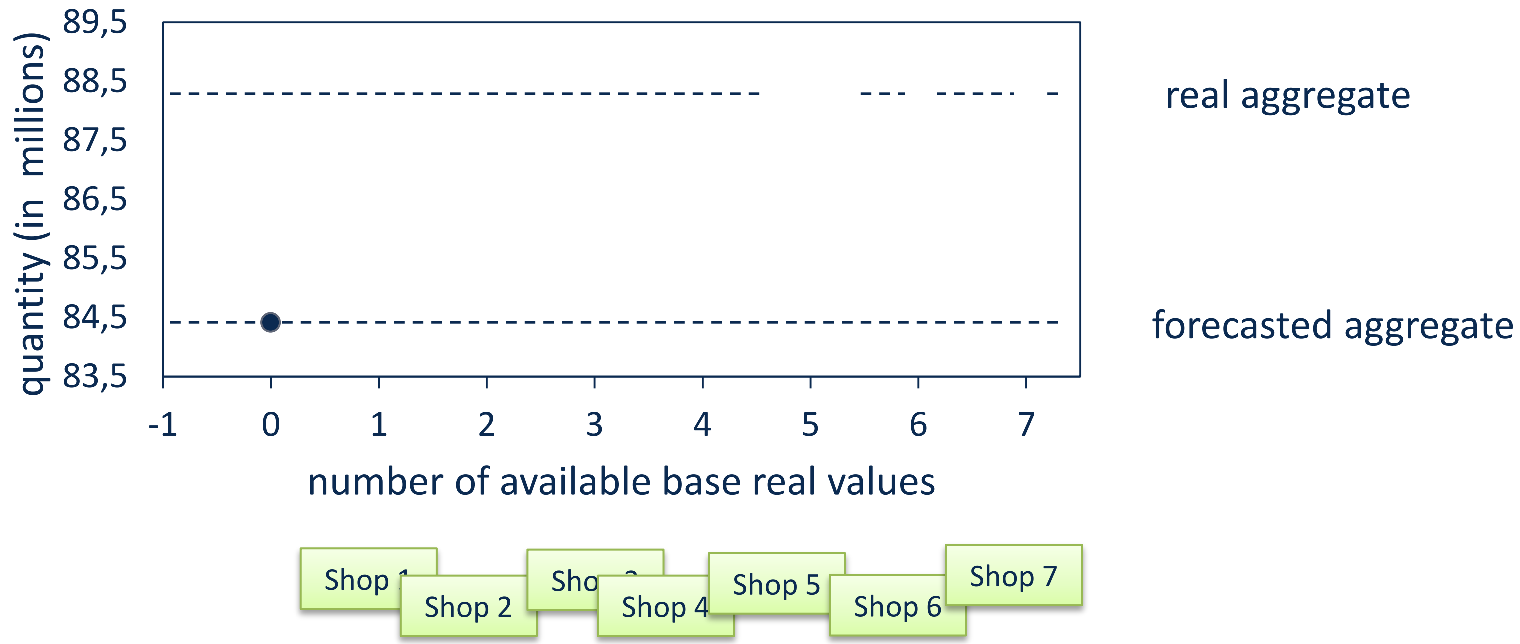
- Maintain models and synopses
- Optimize accuracy of estimates



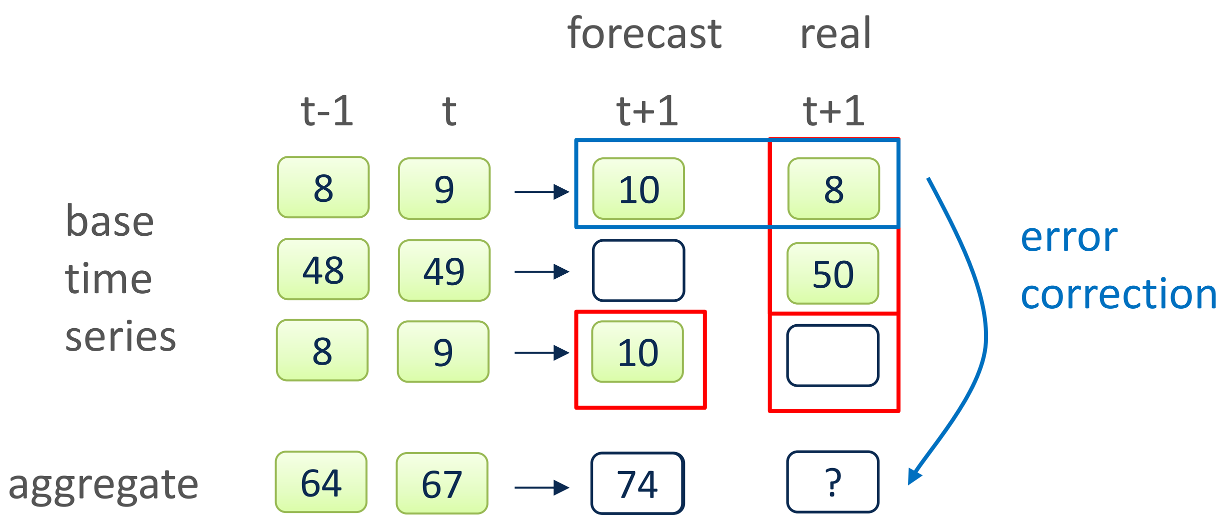
How to include new real data?

Data delivery may be late

- There might be missing data for the last period
- Reports still have to be generated
- Estimate missing data



> Refine

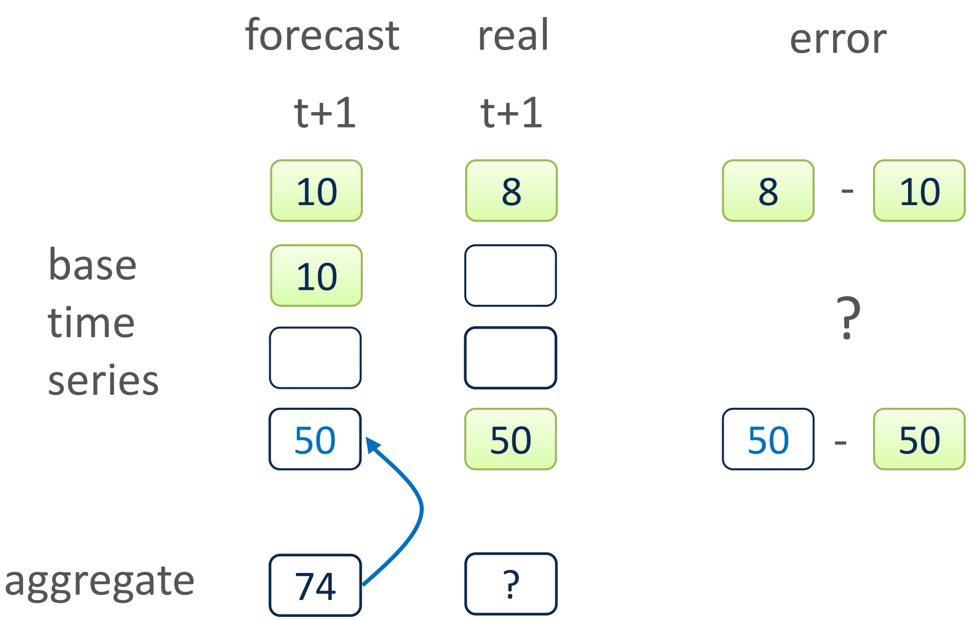


Refine I $\alpha \sum Real$

Refine II $\alpha \sum (Real + Forecast)$

Refine III $\alpha \sum (Real + Forecast) - \beta \sum Error$

Refine III: Estimation of forecast errors



Case 1

- Forecast and real value
- Calculate true model error

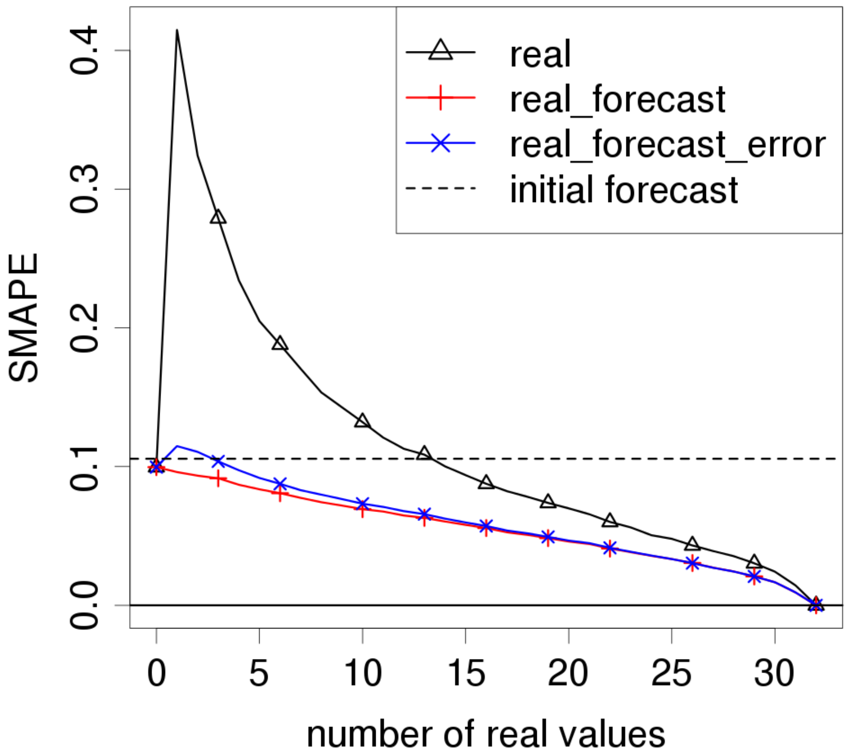
Case 2/3

- No real value
- No error calculation

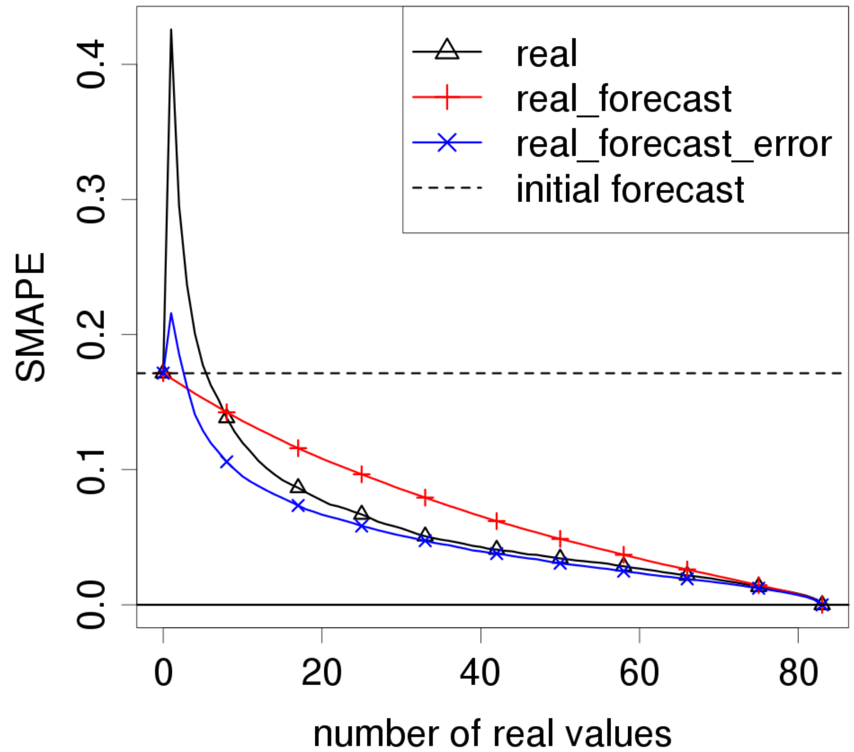
Case 4

- No forecast value
- Estimate model error

Refine – Sales



Refine – Wind production



> Conclusions

Flash Forward Query

FFQ project

- Provide forecasting as 1st class citizen within a database system
- Preserve logical and physical data independence (e.g. transparent model usage, transparent model maintenance, and model creation)
- Extend traditional processing and optimization techniques
- Apply concept of traditional index advisors to foreast models

Towards a model-based database system

- Data is increasingly inconsistent, incomplete and imprecise
- Extend concept of models to other use cases (missing data, uncertain data, data compression …)

Wolfgang Lehner

Forecasting and Data Imputation Strategies in Database Systems

11.07.2013