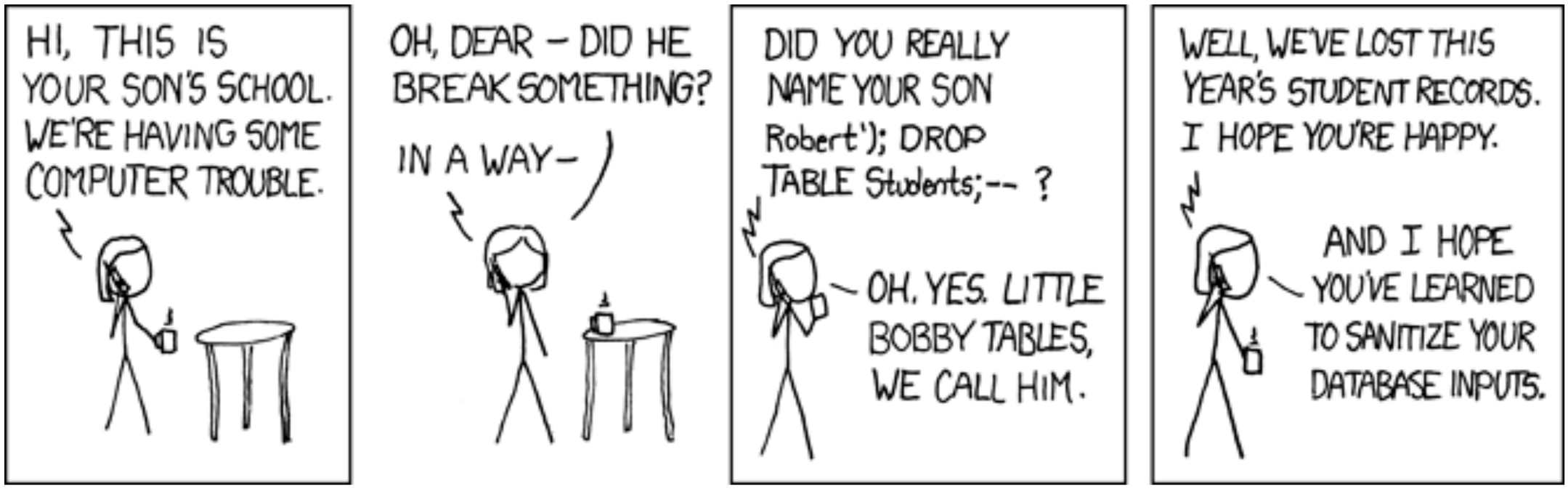
Learning SQL for Database Intrusion Detection using Context-sensitive Modelling

Martin Apel, Christian Bockermann, Michael Meier

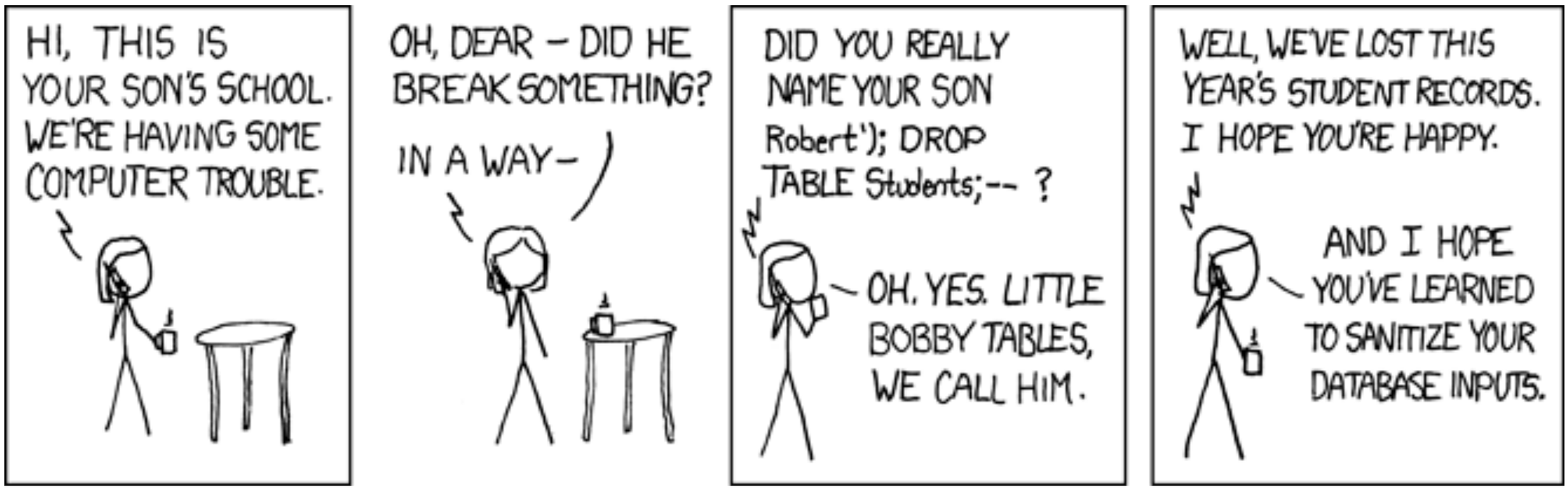
technische universität
dortmund

Fakultät Informatik
Lehrstuhl für
Künstliche Intelligenz

The joke that should not be...



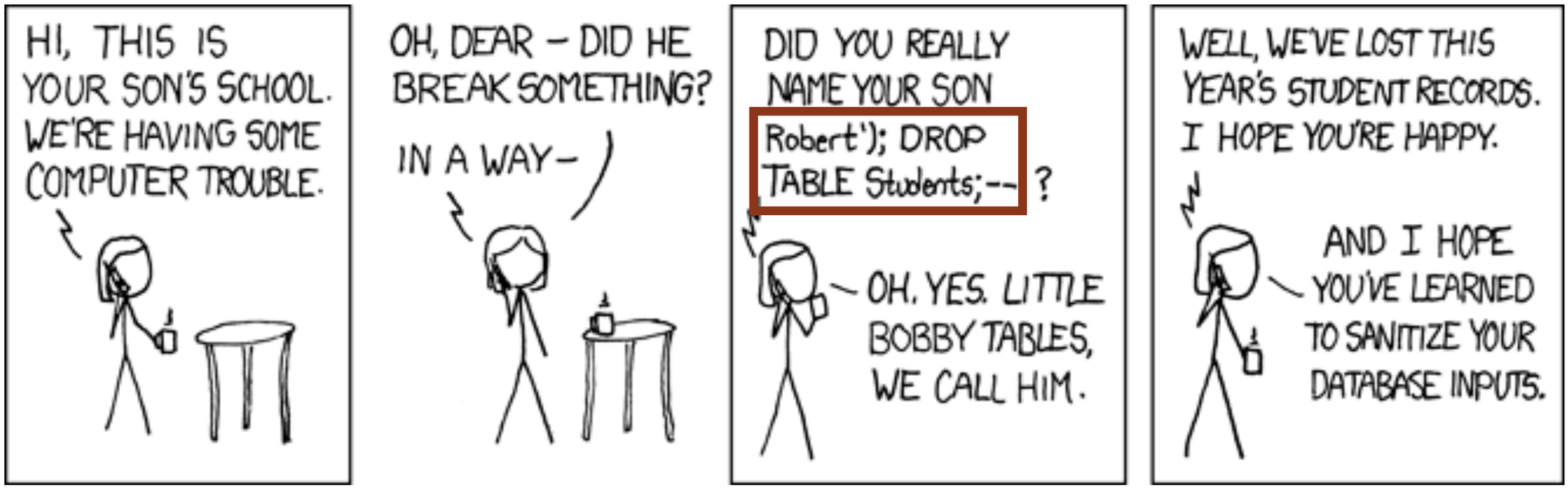
The joke that should not be...




```
$name = $_POST['name'];
  // $name = "Robert'); DROP TABLE Students; --"

$insert = "INSERT INTO STUDENTS VALUES ('$name');";
```

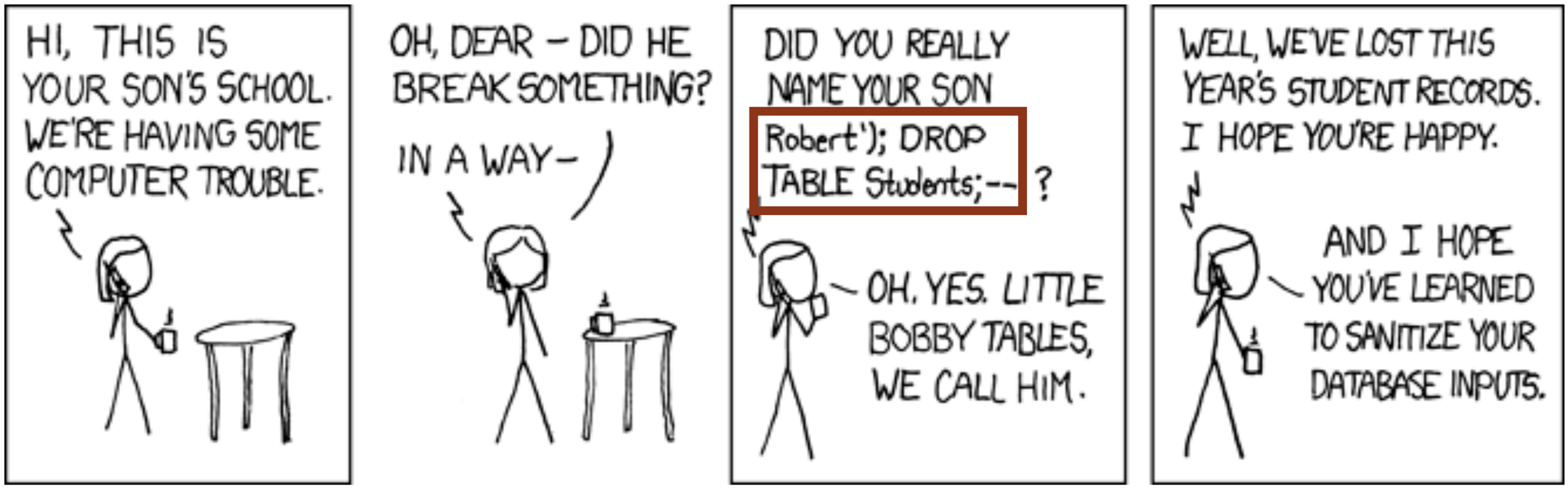
The joke that should not be...




```
$name = $_POST['name'];
  // $name = "Robert'); DROP TABLE Students; --"

$insert = "INSERT INTO STUDENTS VALUES ('$name');";
```

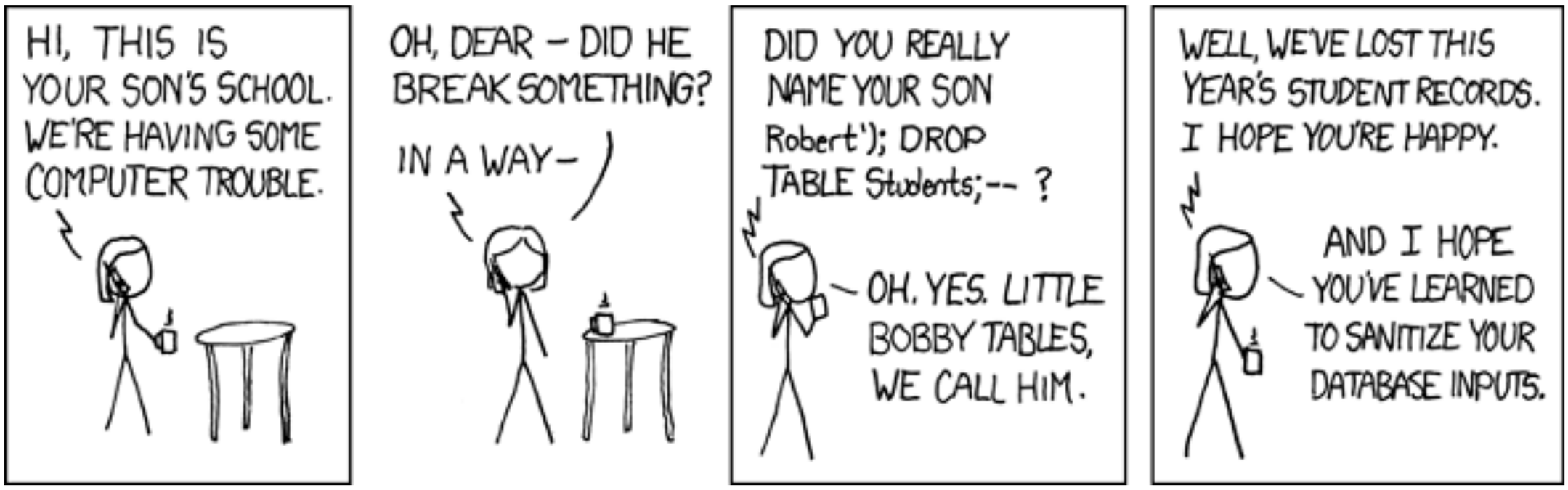
The joke that should not be...




```
$name = $_POST['name'];
  // $name = "Robert'); DROP TABLE Students; --"

$insert = "INSERT INTO STUDENTS VALUES ('$name');";
```

The joke that should not be...



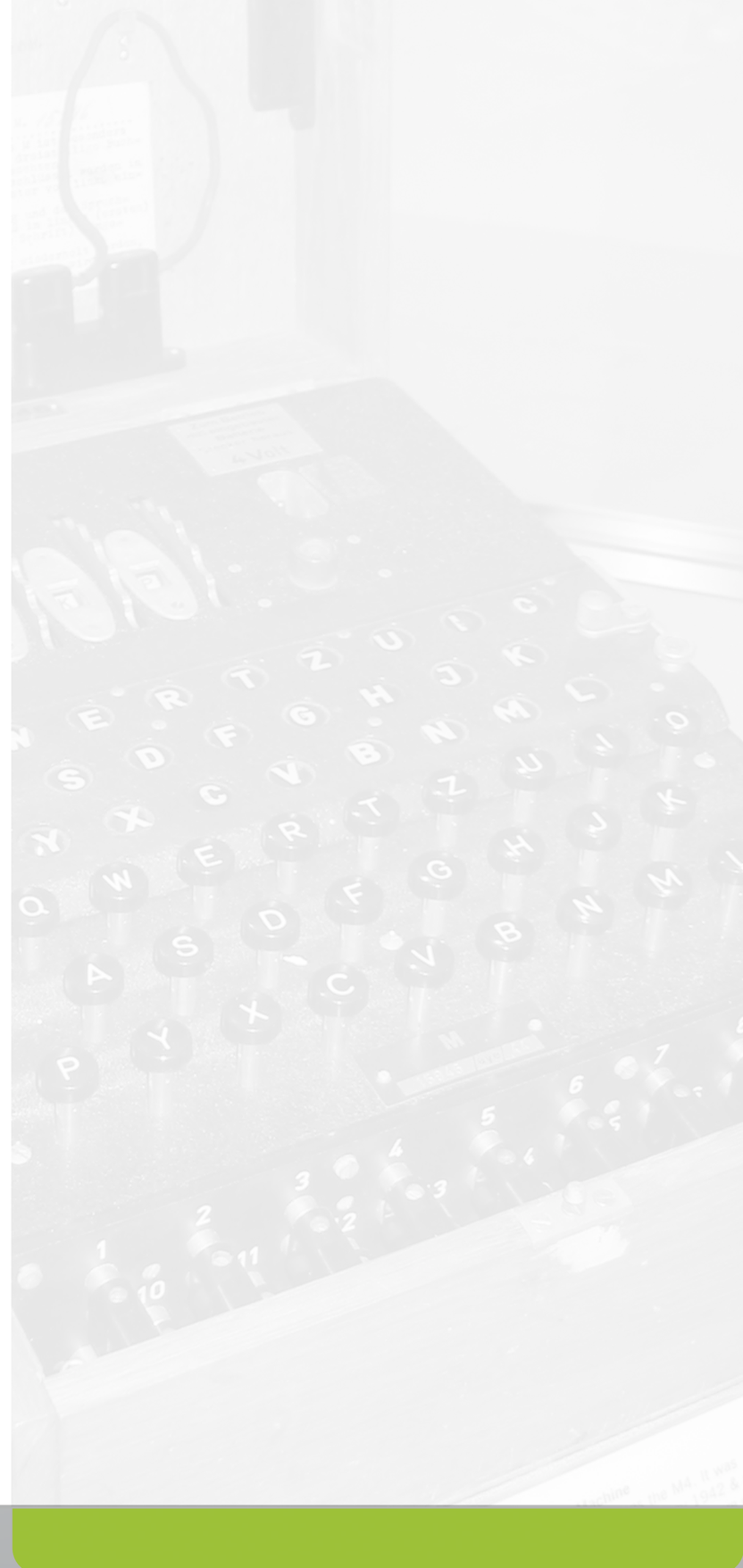

```
$name = $_POST['name'];
  // $name = "Robert'); DROP TABLE Students; --"

$insert = "INSERT INTO STUDENTS VALUES ('$name');";
```

INSERT INTO STUDENTS VALUES ('Robert'); DROP TABLE Students; -- ');

Isn‘t SQL Injection rather old?

- Most attacks are focusing on XSS/client side script evaluation



Isn‘t SQL Injection rather old?

- Most attacks are focusing on XSS/client side script evaluation
- OWASP Top-10 list, still ranks SQL-Injection one of the major vulnerabilities:

1. Cross Site Scripting (XSS)
2. Injections Flaws (SQL-Injection,...)
3. Malicious File Execution (RFI)
4. Insecure Direct Object Reference...
5. ...

Isn‘t SQL Injection rather old?

- Most attacks are focusing on XSS/client side script evaluation
- OWASP Top-10 list, still ranks SQL-Injection one of the major vulnerabilities:

1. Cross Site Scripting (XSS)
2. Injections Flaws (SQL-Injection,...)
3. Malicious File Execution (RFI)
4. Insecure Direct Object Reference...
5. ...

Web Hacking Incident Database

Ofer Shezaf et.al., whid.xiom.com

SQL injection Hits Sensitive US Army servers

WHID 2009-40

245,000 records stolen from Orange France using SQL injection

WHID 2009-39

Kaspersky site breached using SQL injection, sensitive data exposed

WHID 2009-19

WHID 2009-20: BitDefender joins Kasperski on the Breached side

WHID 2009-20

Related Work - ID in Databases

- **Parse Tree Validation to prevent SQL-Injections**
  - Injected snippets do change overall structure of the query due to user-input
  - Comparing query structures BEFORE and AFTER inserting user-data
  - Comparison based on SQL parse tree
- **DIWeDa - Detecting Intrusions in Web-Databases**
  - Analysing SQL-Sessions as provided by DBMS
  - Queries generalized by keywords, replacing constants

Using Parse Tree Validation to Prevent SQL Injection Attacks
Gregory T. Buehrer, Bruce W. Weide, Paolo A.G. Sivilotti
SEM '05: Proceedings of the 5th international workshop on Software engineering and middleware, ACM, 2005

DIWeDa - Detecting Intrusions in Web-Databases
Alex Roichman, Ehud Gudes
Proceeedings of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security, 2008

Intrusion Detection in DB

- **Idea:** Using machine-learning methods for anomaly detection in SQL database systems
  - Monitor SQL query log
  - Create a „query set model“ for a specific (web-) application in training phase
  - Provide a detector for online-discovery of malicious SQL queries in DBMS after training phase
- **In this work:** Focusing on different SQL representations for learning process

Good ways to learn on SQL?

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

Good ways to learn on SQL?

```
SELECT * FROM USERS
    WHERE login = ‚cb‘
    AND pass = ‚secret‘
```

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

Good ways to learn on SQL?

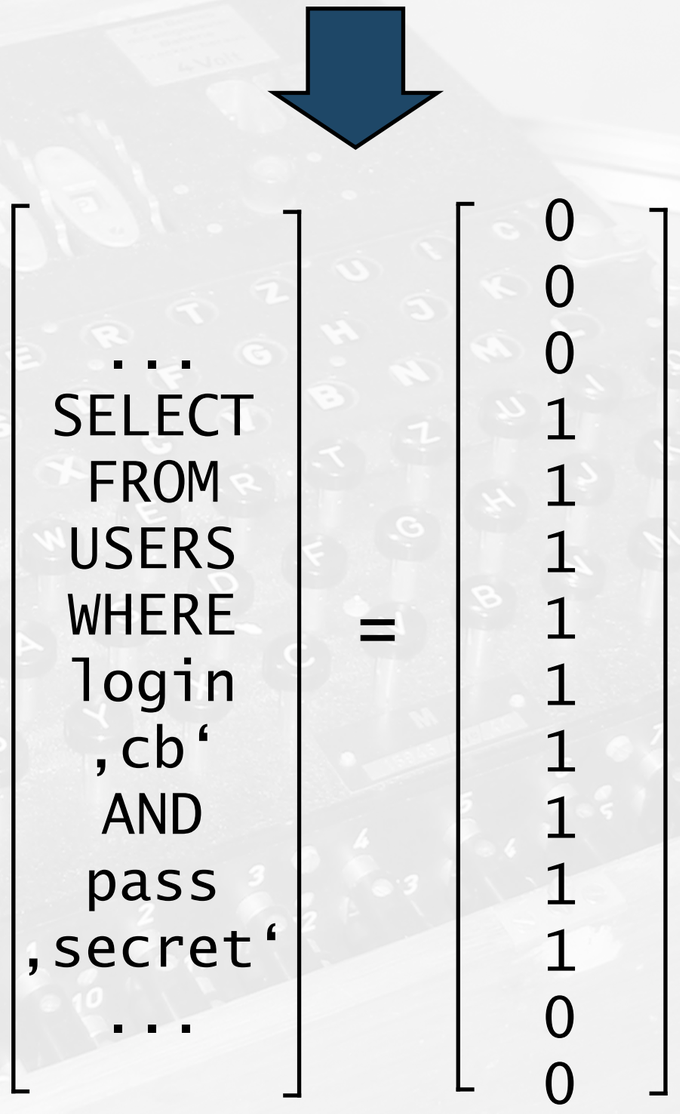
```
SELECT * FROM USERS
    WHERE login = ‚cb‘
    AND pass = ‚secret‘
```

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

Good ways to learn on SQL?

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

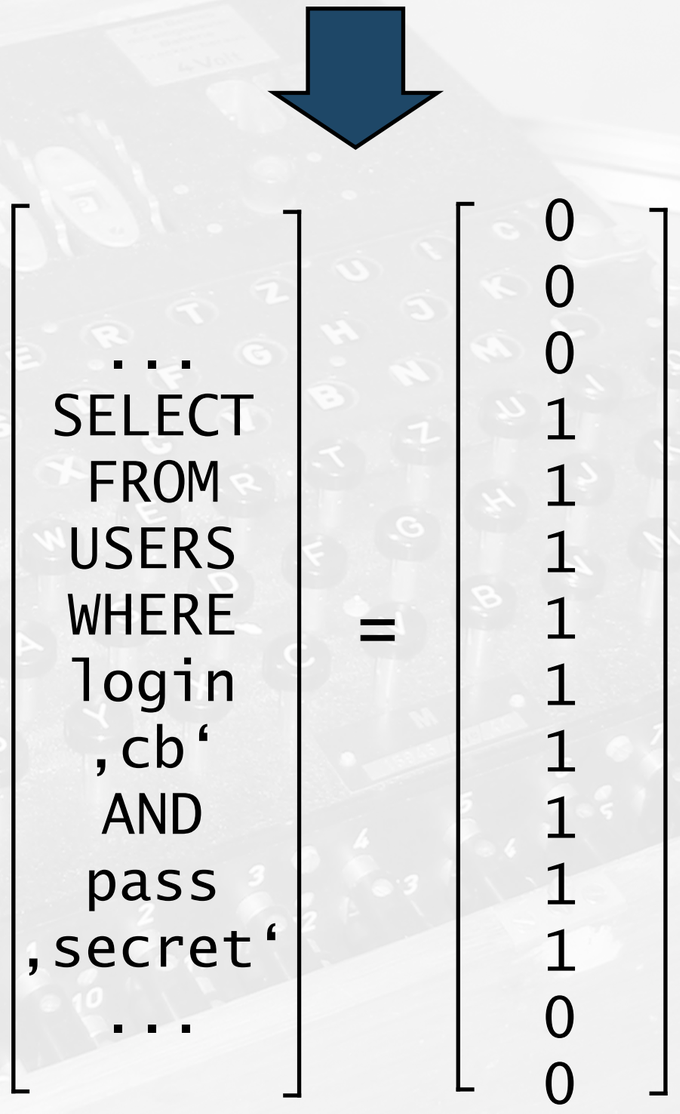
```
SELECT * FROM USERS
    WHERE login = ‚cb‘
    AND pass = ‚secret‘
```



Good ways to learn on SQL?

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

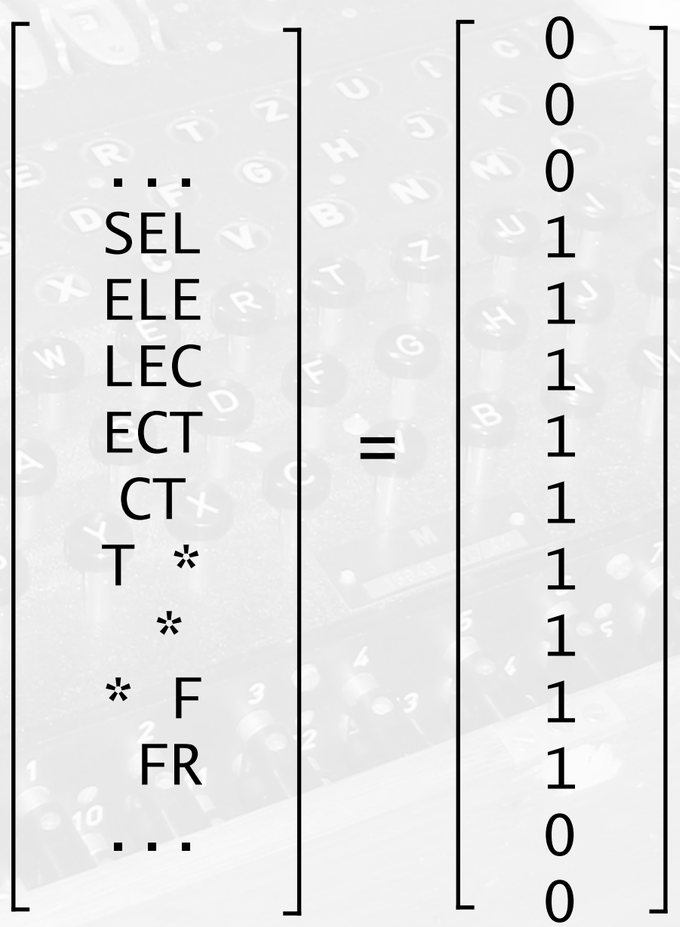
```
SELECT * FROM USERS
    WHERE login = ‚cb‘
    AND pass = ‚secret‘
```



Good ways to learn on SQL?

- Parse Tree validation approach did not use machine learning techniques
- Previous Data Mining approaches do discard query structure in pre-processing phases
- **What are good ways to model SQL for machine learning?**
  - Term-Vectors? n-grams?
  - Using parse tree to create SQL-Vectors
  - Using Tree-Kernel Methods on parse trees

```
SELECT * FROM USERS
    WHERE login = ‚cb‘
    AND pass = ‚secret‘
```

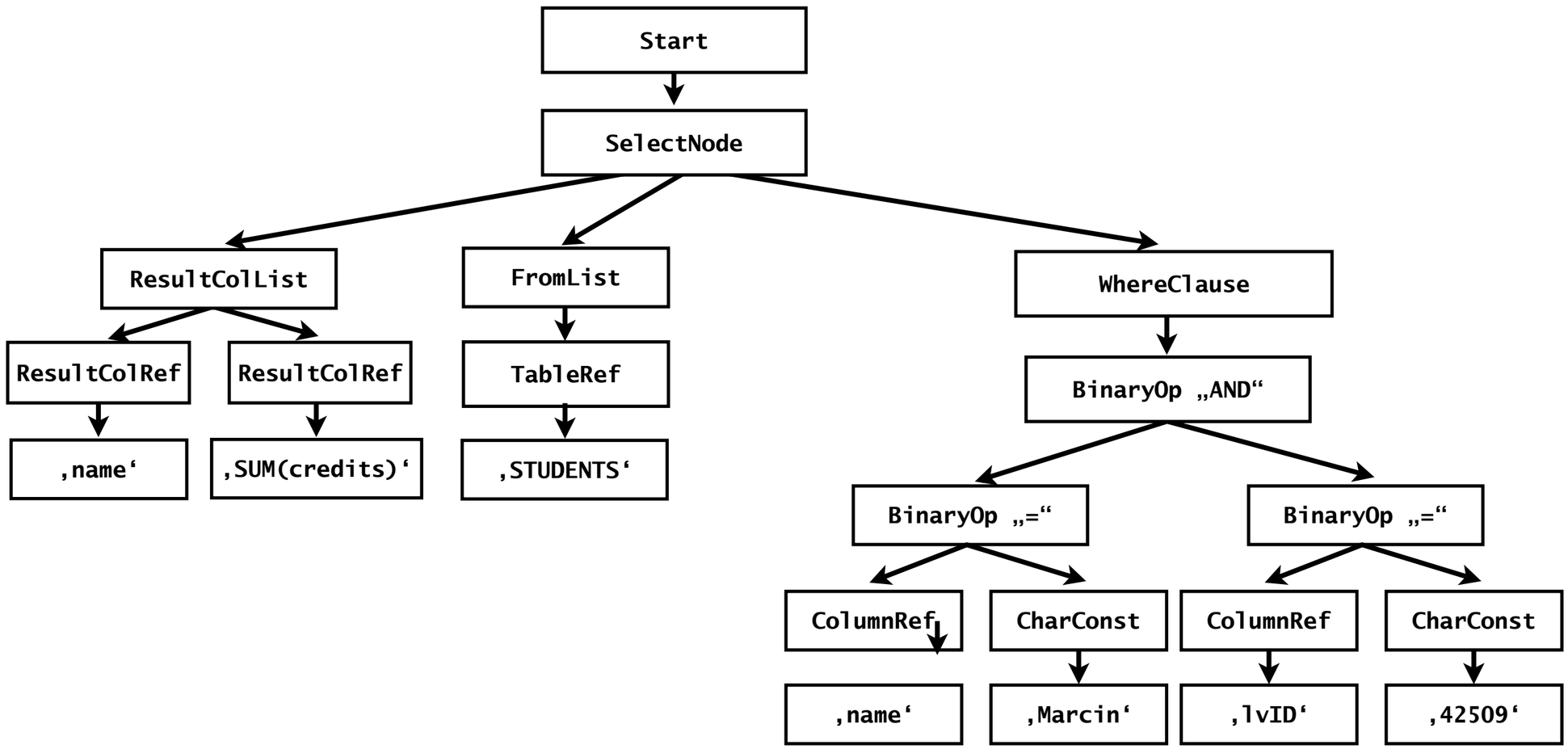


SQL Parse Trees

- SQL is a highly structured language
  - Allows easy and fast parsing
  - DBMS will perform optimization on parse tree
- Every DBMS contains specific SQL parser for its dialect
  - we chose Apache Derby‘s parser (Java) (easy to use in embedded mode)
  - modified Derby parser to obtain parse tree
- Other ways to obtain parsers by generating them based on grammars (antlr.org, javacc)

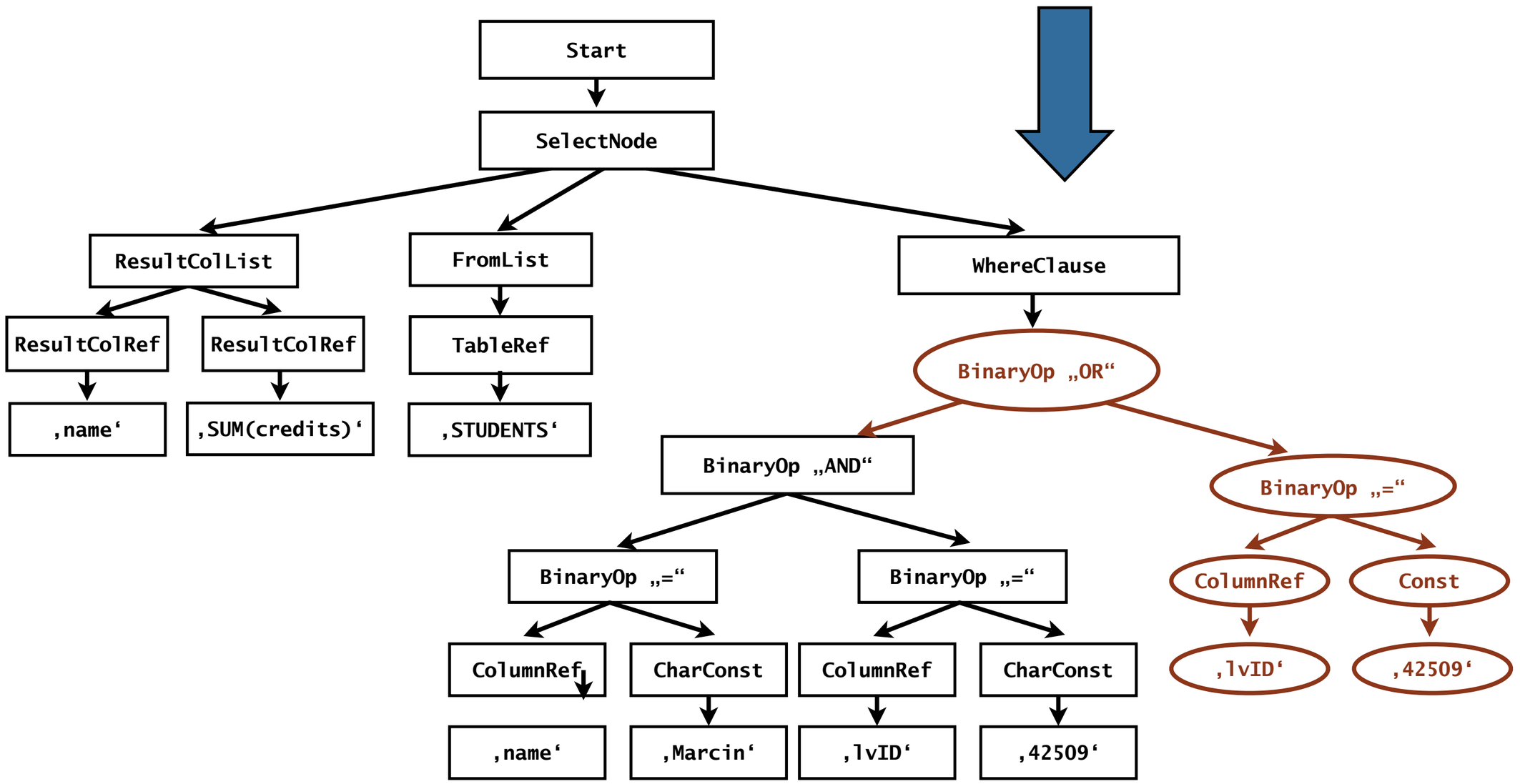
SQL Parse Tree

```
SELECT name,SUM(credits) FROM STUDENTS
    WHERE name = 'Marcin' AND lvID = '42509'
```



SQL Parse Tree of an SQL-Injection

```
SELECT name,SUM(credits) FROM STUDENTS
    WHERE name = 'Marcin' AND lvID = '42509‘ OR 1 > 0 --'
```

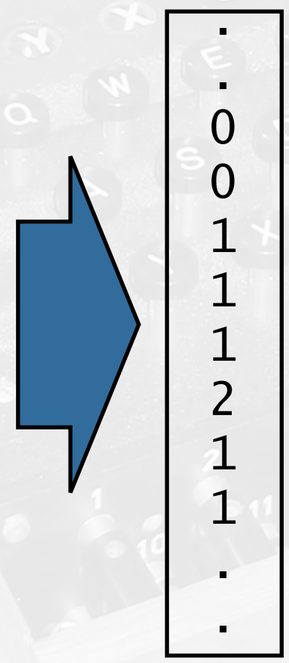


Vectorizing SQL Parse Trees

- We derived the production rules from SQL parse trees to create a histogram-vector of a query
- The i-th component of a vector denotes the number of applications of the i-th grammar rule

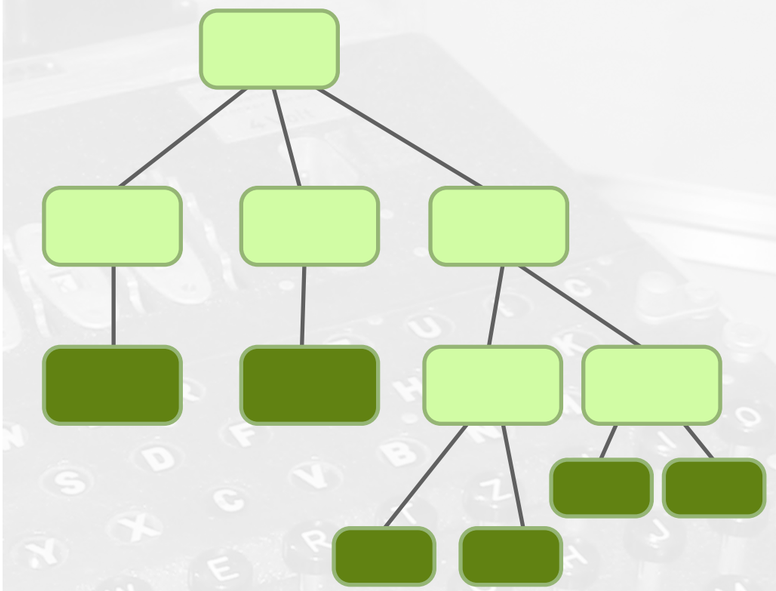
```
SELECT name,SUM(PUNKTE) FROM STUDENTS
       WHERE name = 'Marcin' AND lvID = '42509'

Start --> SelectNode
  SelectNode --> ResultCols FromList WhereClause
      ResultCols --> ResultColumn ResultColumn
    ResultColumn --> ColumnReference
      ColumnReference --> 'NAME'
    ResultColumn --> ColumnReference
      AggregateNode --> SUM
        ColumnReference --> 'PUNKTE'
```



Vectorizing SQL Parse Trees

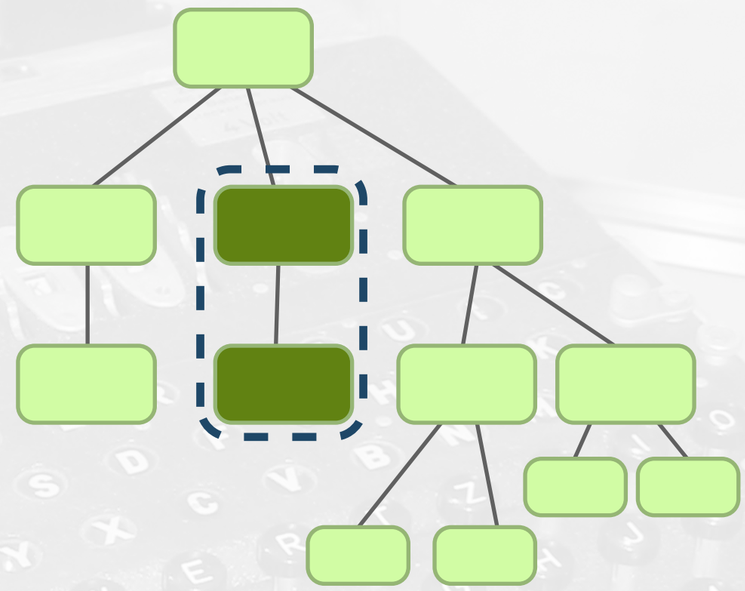
- Term vectors only consider leaves of parse trees



Term Vectorization

Vectorizing SQL Parse Trees

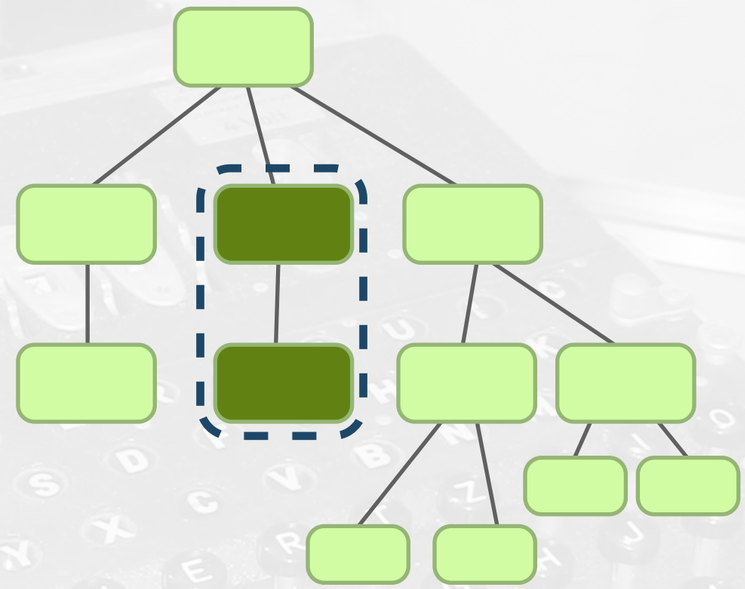
- Term vectors only consider leaves of parse trees
- SQL rule vectorizations includes small context of each node (i.e. predecessor)



SQL (Rule) Vectors

Vectorizing SQL Parse Trees

- Term vectors only consider leaves of parse trees
- SQL rule vectorizations includes small context of each node (i.e. predecessor)
- Integer values vectors can be used in a variety of Data Mining methods
  - Clustering, Outlier detection, Classification



SQL (Rule) Vectors

How does context help?

- We investigated the separability of n-grams, term-vectors and SQL-rule vectors
- Each query was represented by a vector and an SVM was trained to distinguish between queries labeled as attacks/normal

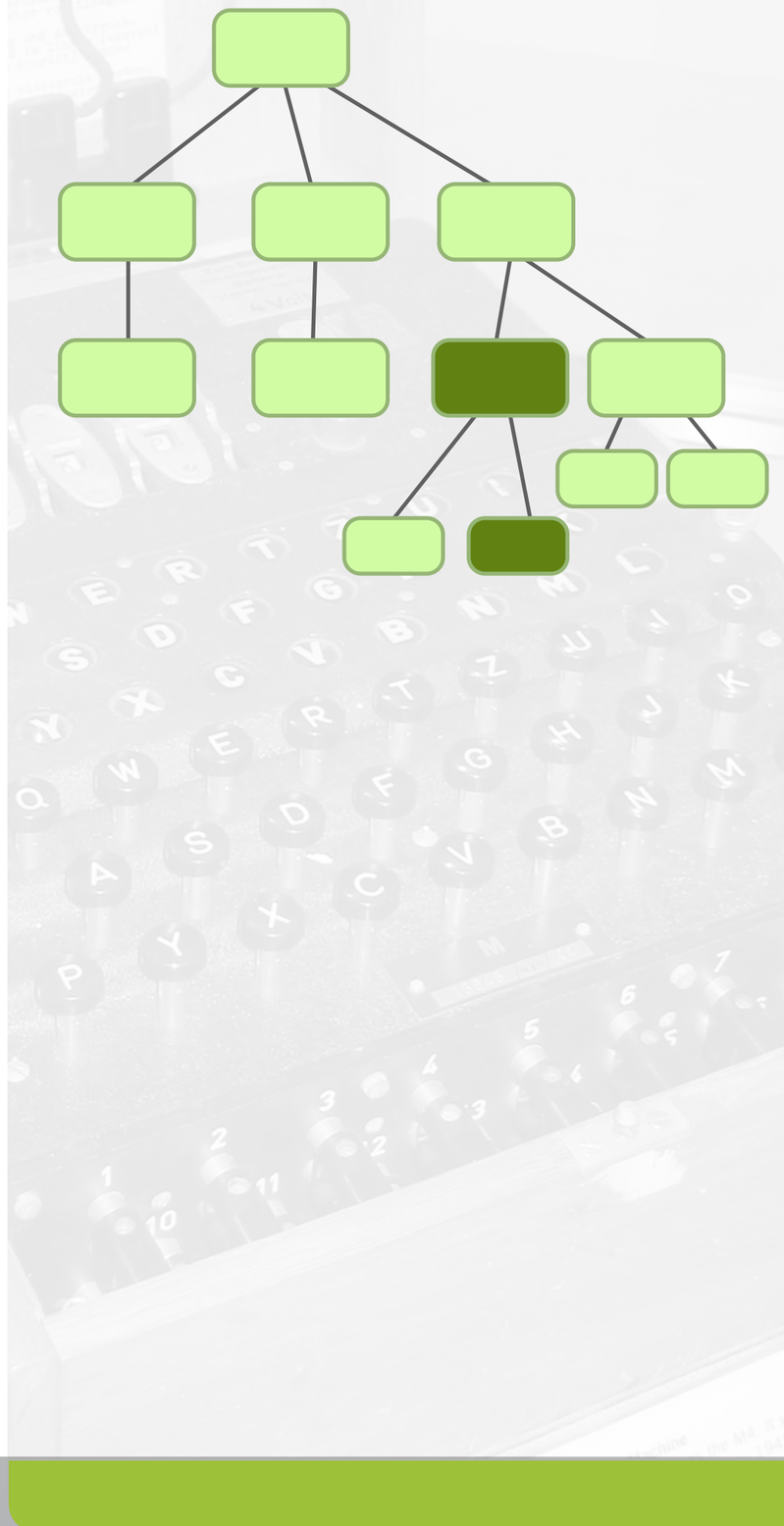
	true pos	false pos	time (s)	true pos	false pos	time (s)
3-gram	0,667	0,000	71	0,667	0,002	643
4-gram	0,333	0,000	149	0,733	0,002	1.055
Term-vectors	0,667	0,001	2	0,733	0,002	283
SQL-vectors	0,867	0,000	3	0,867	0,001	67
	200 legal queries, 15 attacks			1000 legal queries, 15 attacks		

SVM results on different models

(10-fold Xval, optimized C,kernel-type)

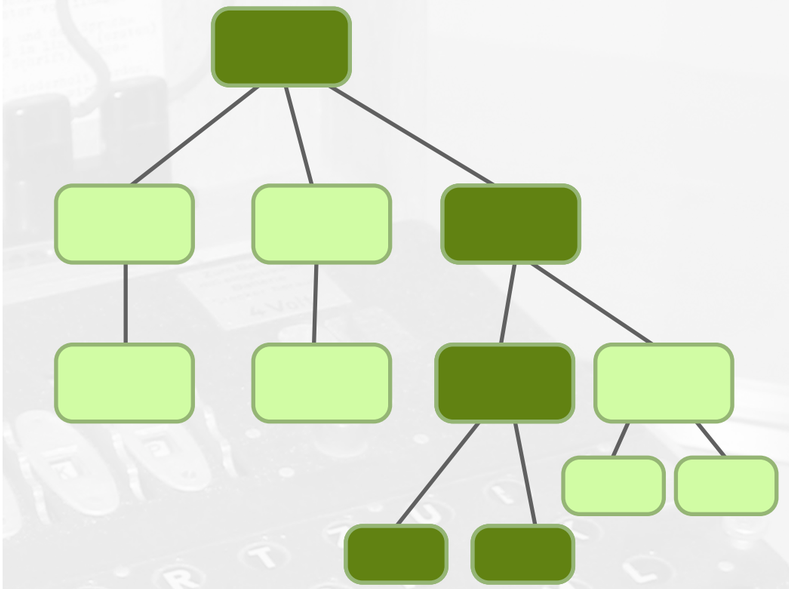
What about more context?

- Rule based vectors consider ancestor context
- Tree/Graph Kernel functions approach in Machine Learning
  - Used in Natural Language Processing
  - Applied for IDS on Protocol Layer
- *kernel function* on trees: basically measuring similarity of two trees



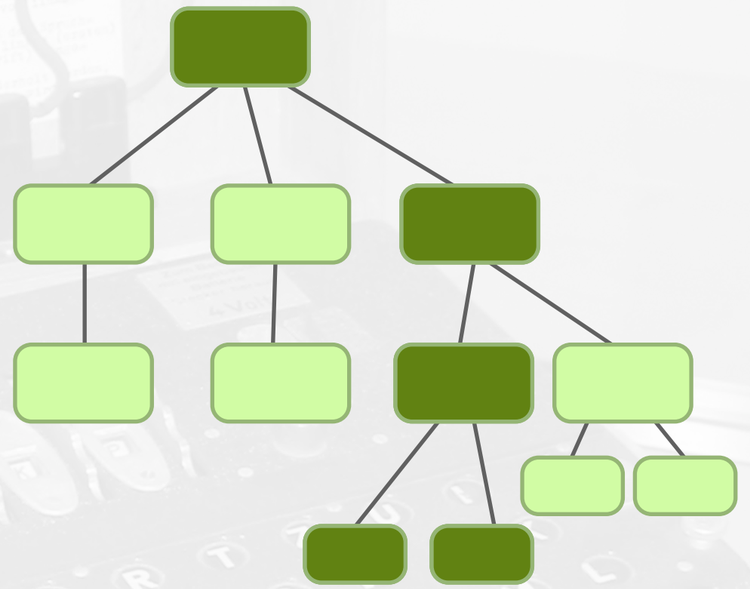
What about more context?

- Rule based vectors consider ancestor context
- Tree/Graph Kernel functions approach in Machine Learning
  - Used in Natural Language Processing
  - Applied for IDS on Protocol Layer
- *kernel function* on trees: basically measuring similarity of two trees



What about more context?

- Rule based vectors consider ancestor context
- Tree/Graph Kernel functions approach in Machine Learning
  - Used in Natural Language Processing
  - Applied for IDS on Protocol Layer



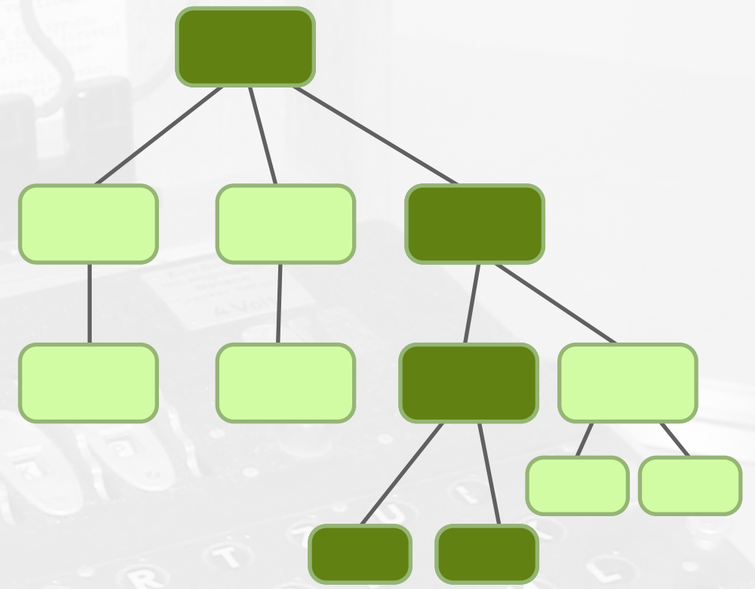
Convolution Kernels on Discrete Structures
D. Haussler, *Technical Report, University of Santa Cruz, 1999*

Kernels and Distances for Structured Data
Thomas Gärtner, John W. Lloyd, Peter A. Flach, *Machine Learning, 2004*

Incorporation of Application Layer Protocol Syntax into Anomal. Detc.
Düssel, Gehl, Laskov, Rieck, in *Proc. of Int. Conf on Information Systems Security*, 2008

What about more context?

- Rule based vectors consider ancestor context
- Tree/Graph Kernel functions approach in Machine Learning
  - Used in Natural Language Processing
  - Applied for IDS on Protocol Layer



- *kernel function* on trees: basically measuring similarity of two trees

$$k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$$

Convolution Kernels on Discrete Structures
D. Haussler, *Technical Report, University of Santa Cruz, 1999*

Kernels and Distances for Structured Data
Thomas Gärtner, John W. Lloyd, Peter A. Flach, *Machine Learning, 2004*

Incorporation of Application Layer Protocol Syntax into Anomal. Detc.
Düssel, Gehl, Laskov, Rieck, in *Proc. of Int. Conf on Information Systems Security*, 2008

Clustering of SQL

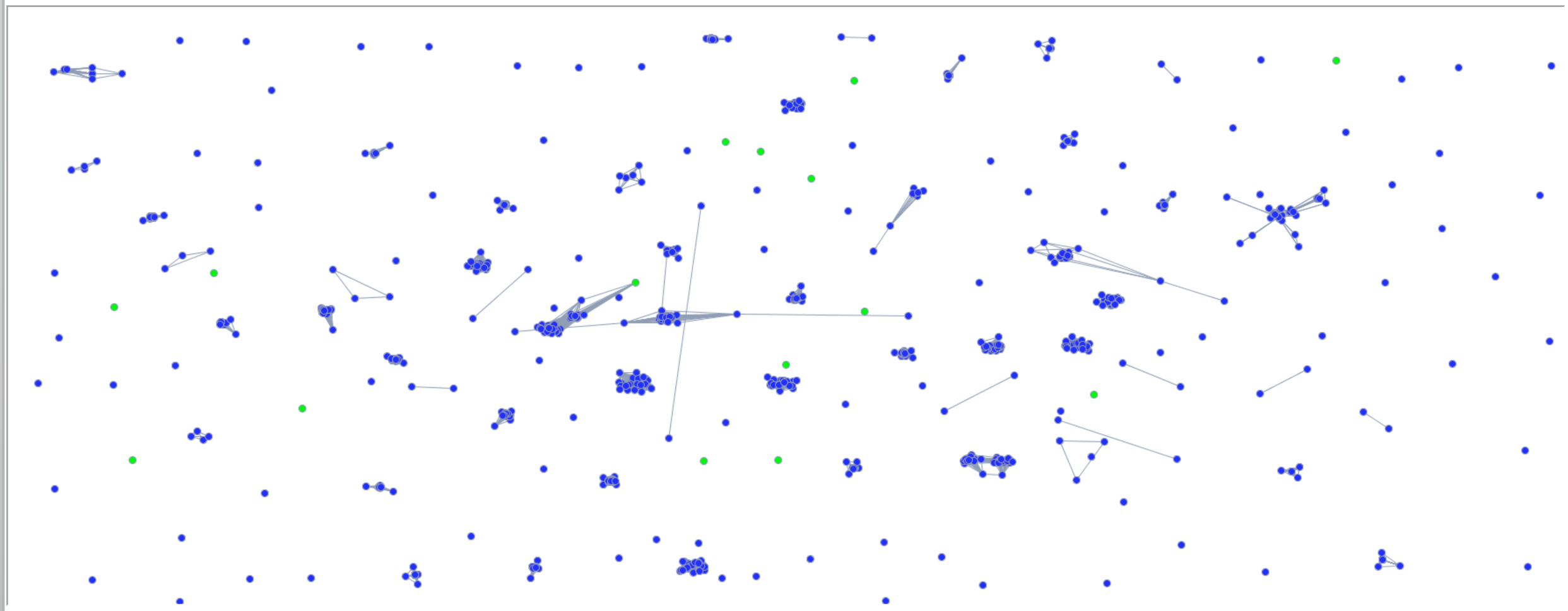
- No SVM with the tree-kernel, yet, therefore we used the tree kernel as similarity measure for:
  - Clustering of SQL queries
  - Outlier detection (work in progress)

- Distance based on kernel *k* can be defined as

$$d_k(x, x') = \sqrt{k(x, x) - 2k(x, x') + k(x', x')}$$

Clustering of SQL

- We created a similarity matrix (pairwise similarity of queries) and visualized this in an ISOM



the typo3 *Big Picture* :-)

Clustering of SQL

- We inspected the clusters manually, showing reasonable results
  - similarity measure follows intuition
  - clusters revealed groups of page-fetches, user-logins, session invalidation, etc.
- Inserted (synthetical) attacks remained isolated, even though only marginally changed

Summary / Future Work

- The work addresses the use of syntactical context for anomaly detection in SQL queries
- First results show precision gains by incorporating syntax into SQL log-file analysis
- **Future work** will focus on
  - additional evaluation (more applications)
  - parser/grammar extensions (other dialects)
  - unsupervised detection methods (outlier search)
  - Derivation of application specific SQL policies (*SQL-firewall rules)*