A Practical Framework for Privacy-Preserving NoSQL Databases

36th IEEE International Symposium on Reliable Distributed Systems
Hong Kong, 27th September 2017

Ricardo Macedo[1], João Paulo[1], Rogério Pontes[1], Bernardo Portela[2],
Tiago Oliveira[2], Miguel Matos[3], Rui Oliveira[1]

[1] High Assurance Software Lab, INESC TEC and U. Minho, Portugal
[2] High Assurance Software Lab, INESC TEC and FCUP, Portugal
[3] INESC ID/IST, U. Lisboa, Portugal

Cloud Databases

- Multiple applications and online services are used in a daily basis
- Massive amounts of data are stored in databases
- Cloud storage services became the choice for millions of users and enterprises
  - Flexible storage capacity and elevated computing power
  - Costless acquisition and administration of a private infrastructure
  - High efficiency, scalability and ease of use

Challenges and Problems

- Migration of data control to a third party
- Sensitive data exposed through privacy and security failures
- Reluctance on cloud services adoption

Challenges and Problems

- Migration of data control to a third party
- Sensitive data exposed through privacy and security failures
- Reluctance on cloud services adoption

What can be done?

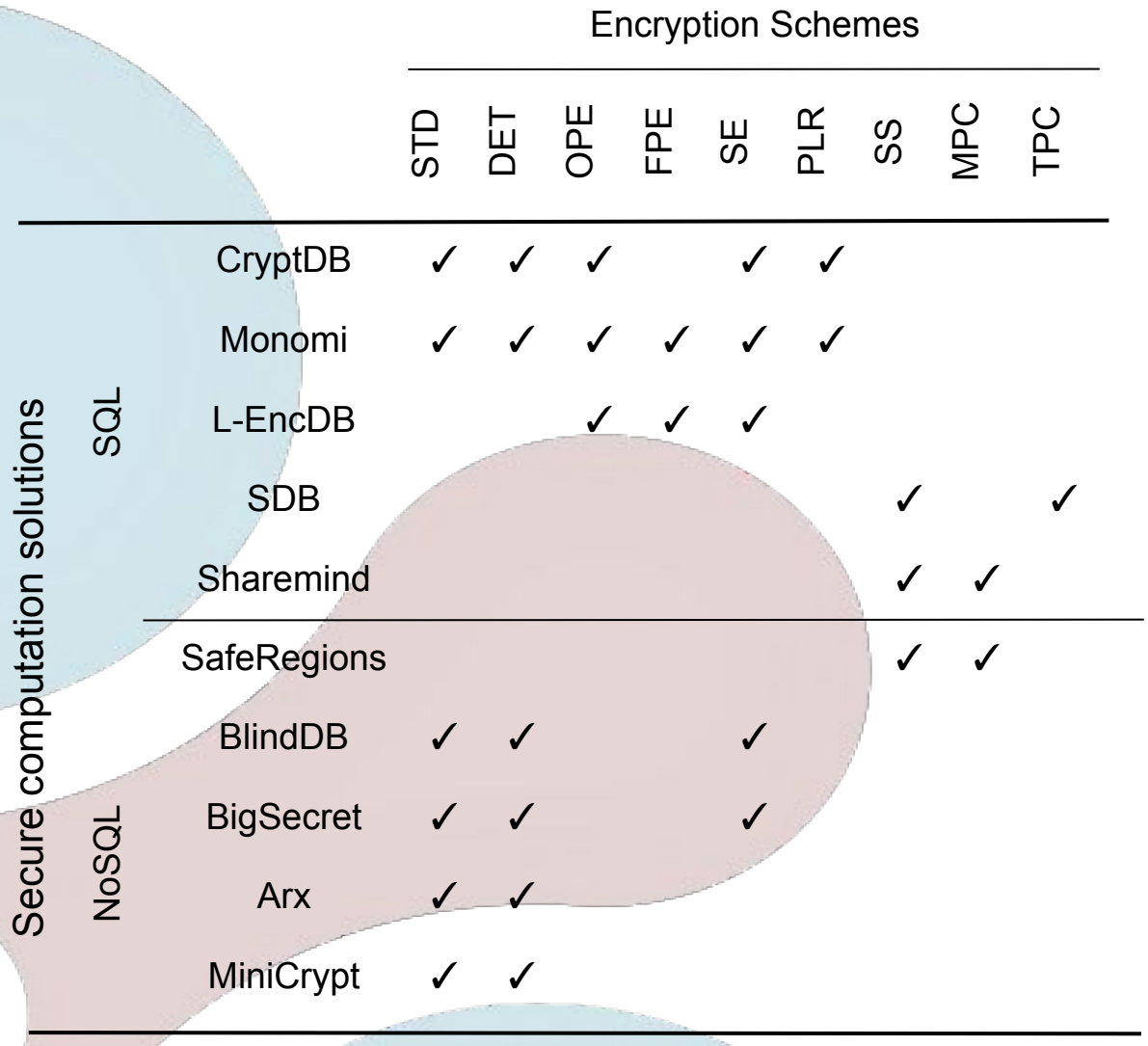
Encryption Schemes

Scheme	Construction	Properties	Operations
Standard Encryption	AES-128 CBC w/o IV	None	Insertions
Deterministic Encryption	AES-128 CBC w/ IV	Equality	Reads, equality filters
Order-Preserving Encryption	Boldyreva et al. ‘09	Equality, Order	Searches, equality and order filters

Security Guarantees ↑

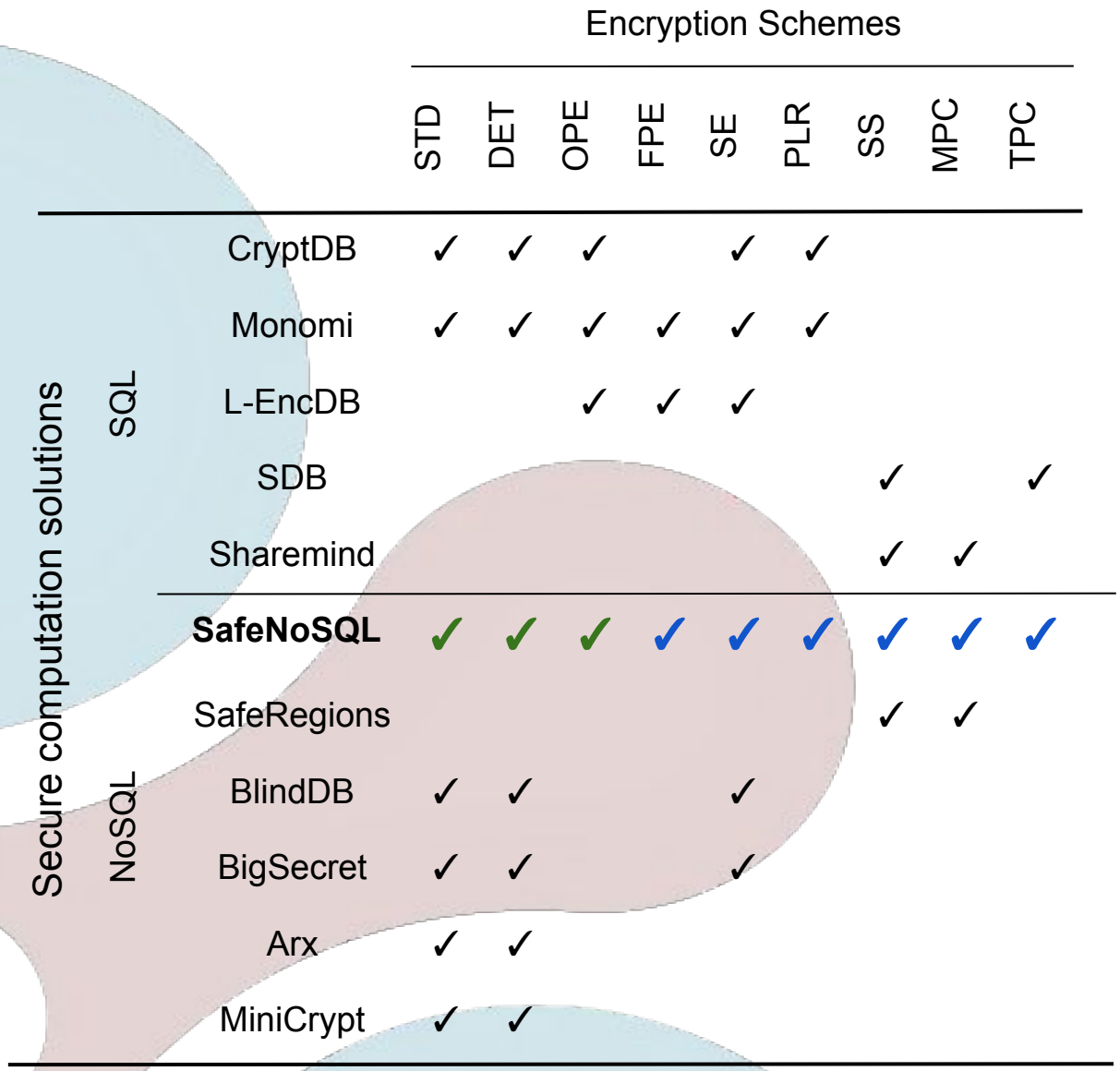
Allowed Operations ↓

Current Solutions



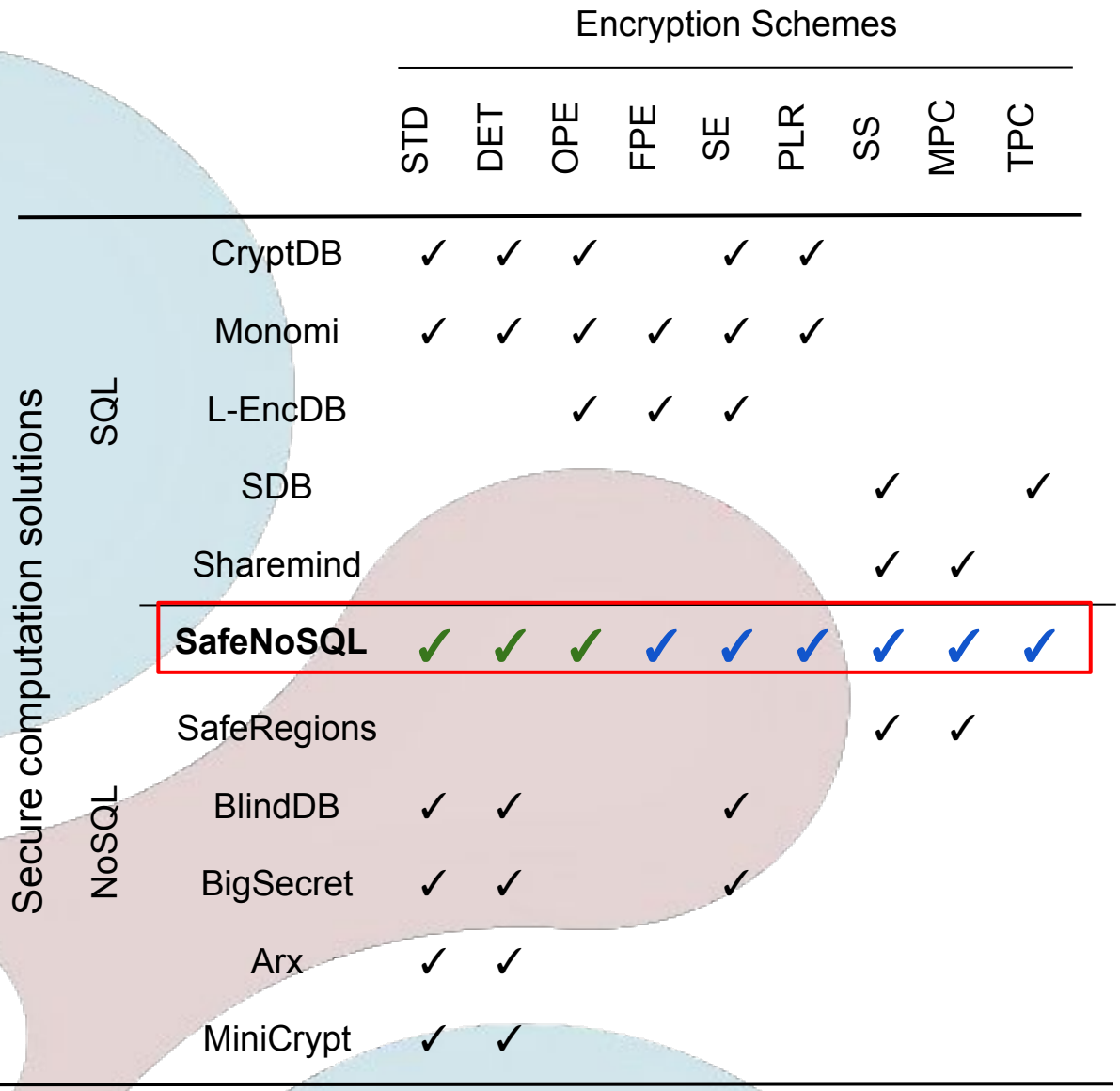
Encryption Schemes:
STD - Standard
DET - Deterministic
OPE - Order-Preserving
FPE - Format-Preserving
PLR - Paillier
SS - Secret Sharing
MPC - Multi-Party Computation
TPC - Two-Party Computation

Current Solutions



Encryption Schemes:
STD - Standard
DET - Deterministic
OPE - Order-Preserving
FPE - Format-Preserving
PLR - Paillier
SS - Secret Sharing
MPC - Multi-Party Computation
TPC - Two-Party Computation

Current Solutions

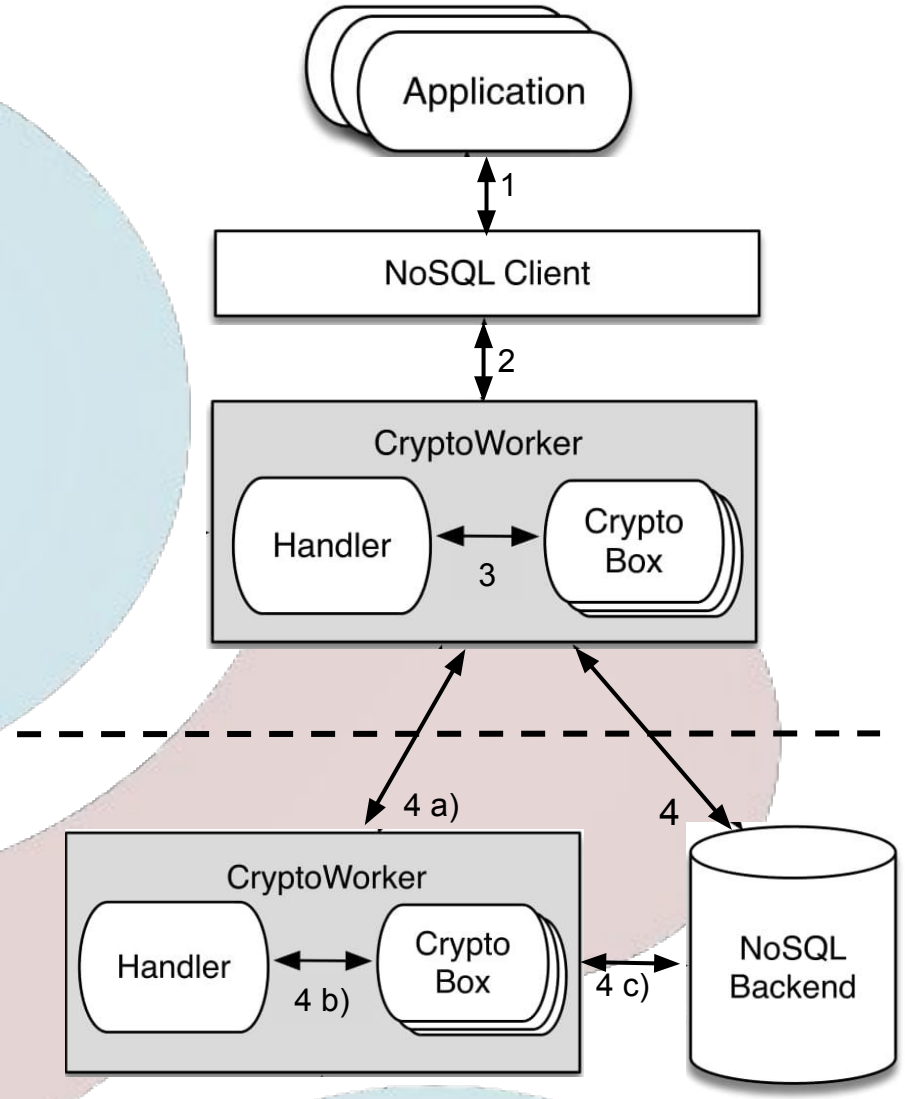


Encryption Schemes:
STD - Standard
DET - Deterministic
OPE - Order-Preserving
FPE - Format-Preserving
PLR - Paillier
SS - Secret Sharing
MPC - Multi-Party Computation
TPC - Two-Party Computation

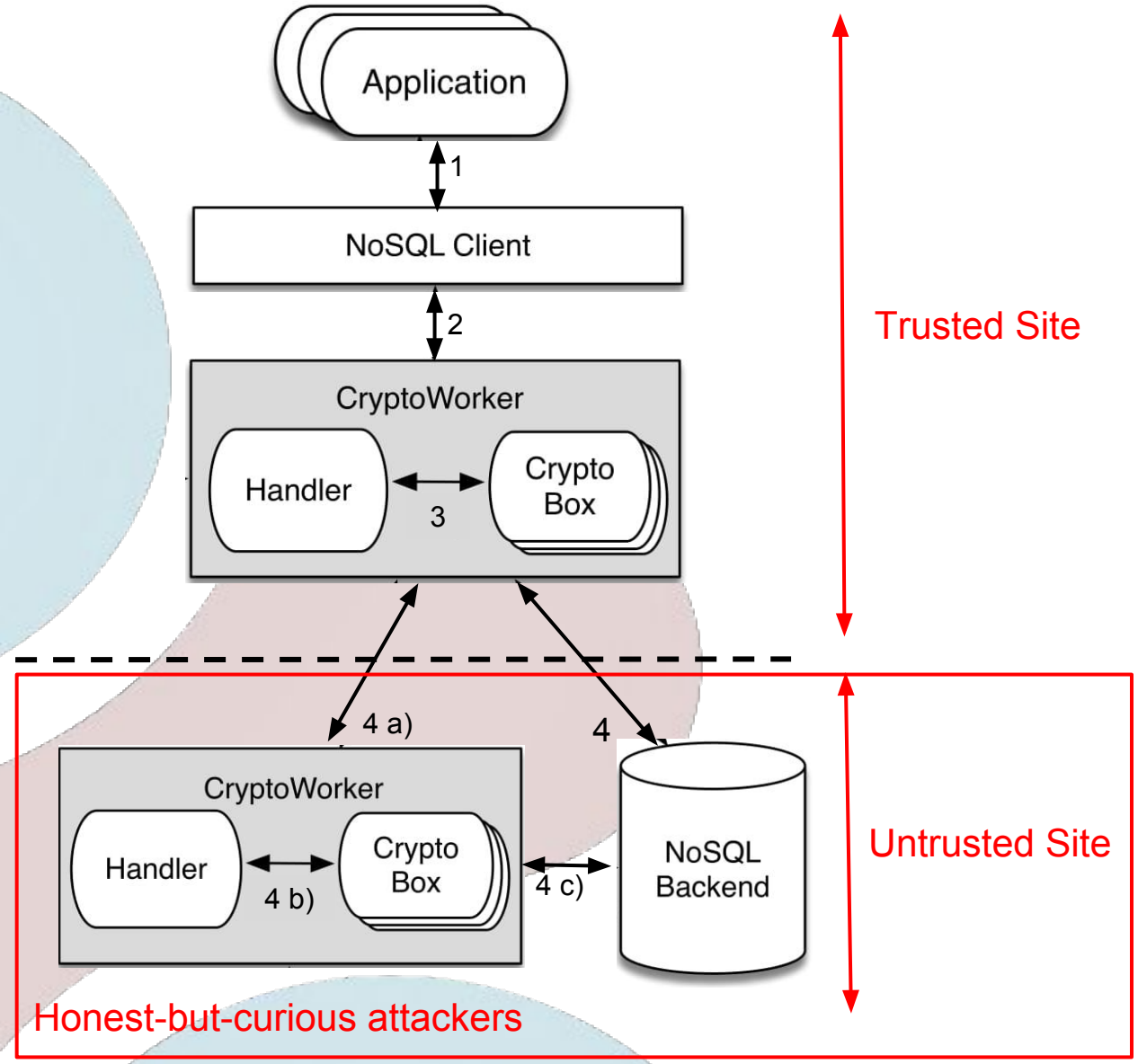
Contributions

- **SafeNoSQL**, a modular and extensible framework for NoSQL databases
  - Secure computation over sensitive data
  - Configurable security to tailor different system requirements
  - Generic to most of NoSQL Key-Value Stores
  - Extensible to several encryption schemes
  - Encryption schemes are exchangeable

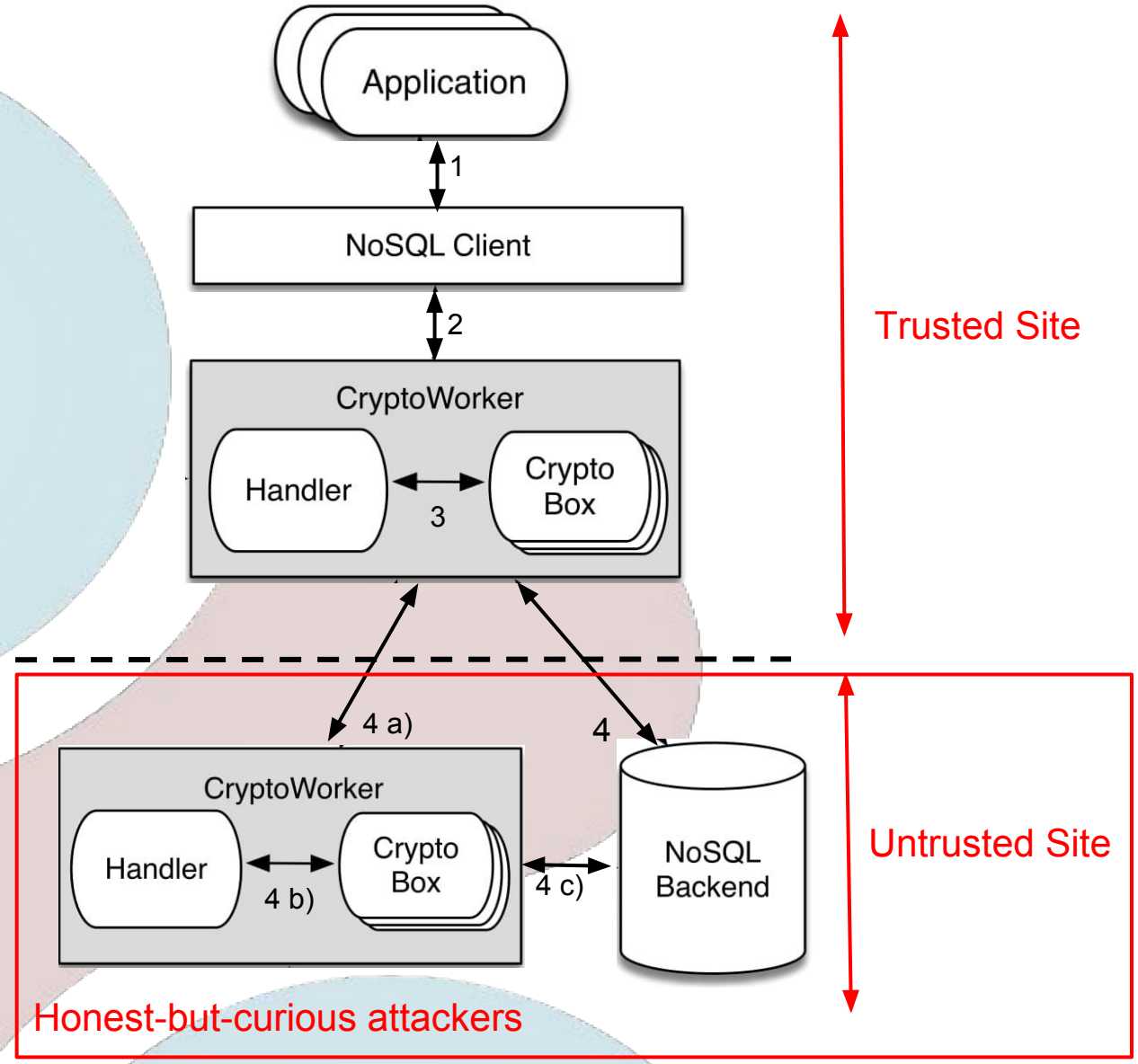
SafeNoSQL: Architecture



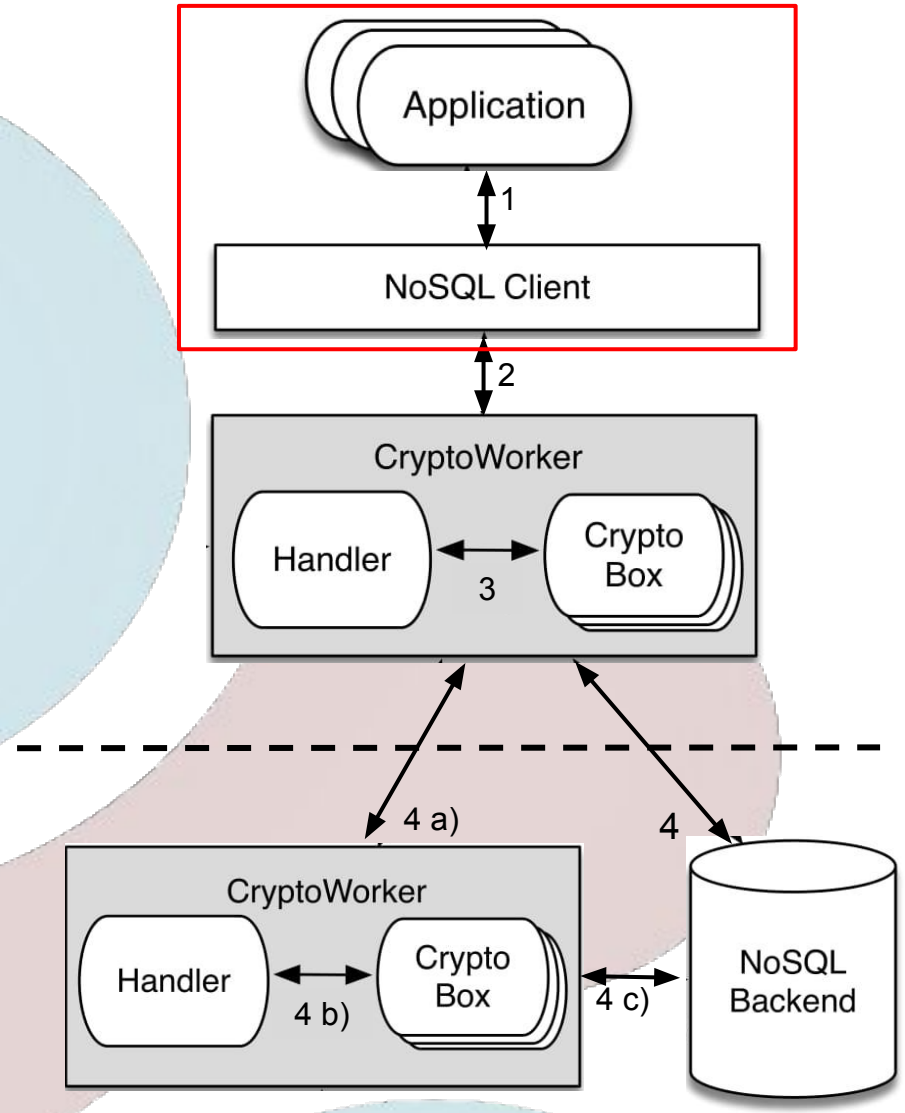
SafeNoSQL: Architecture



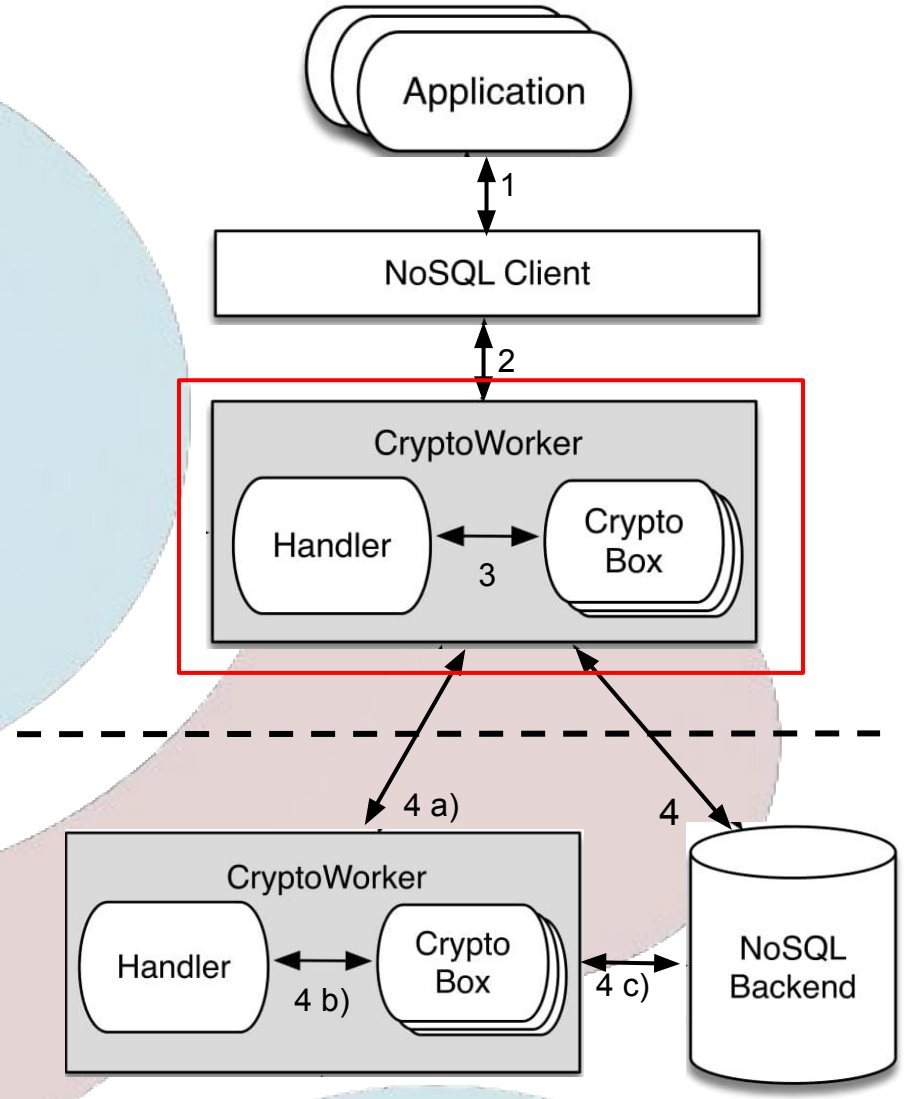
SafeNoSQL: Architecture



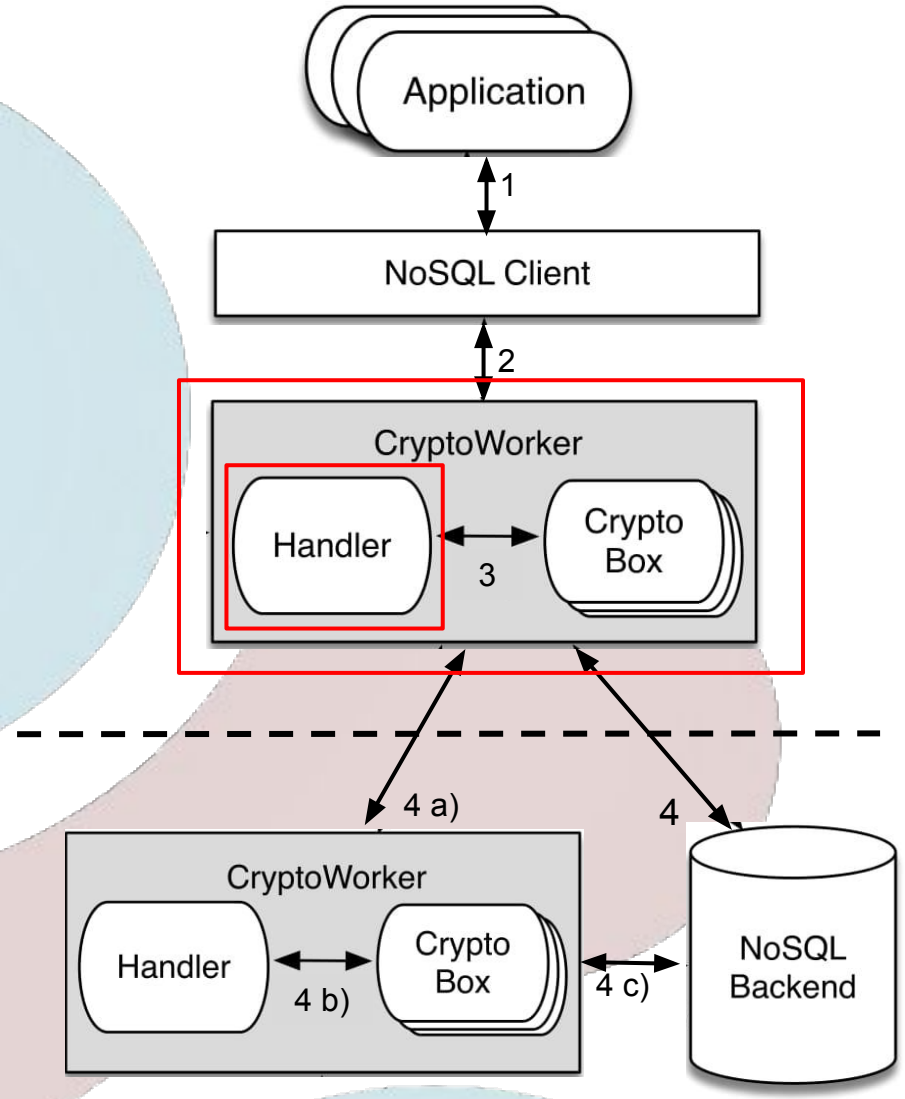
SafeNoSQL: Architecture



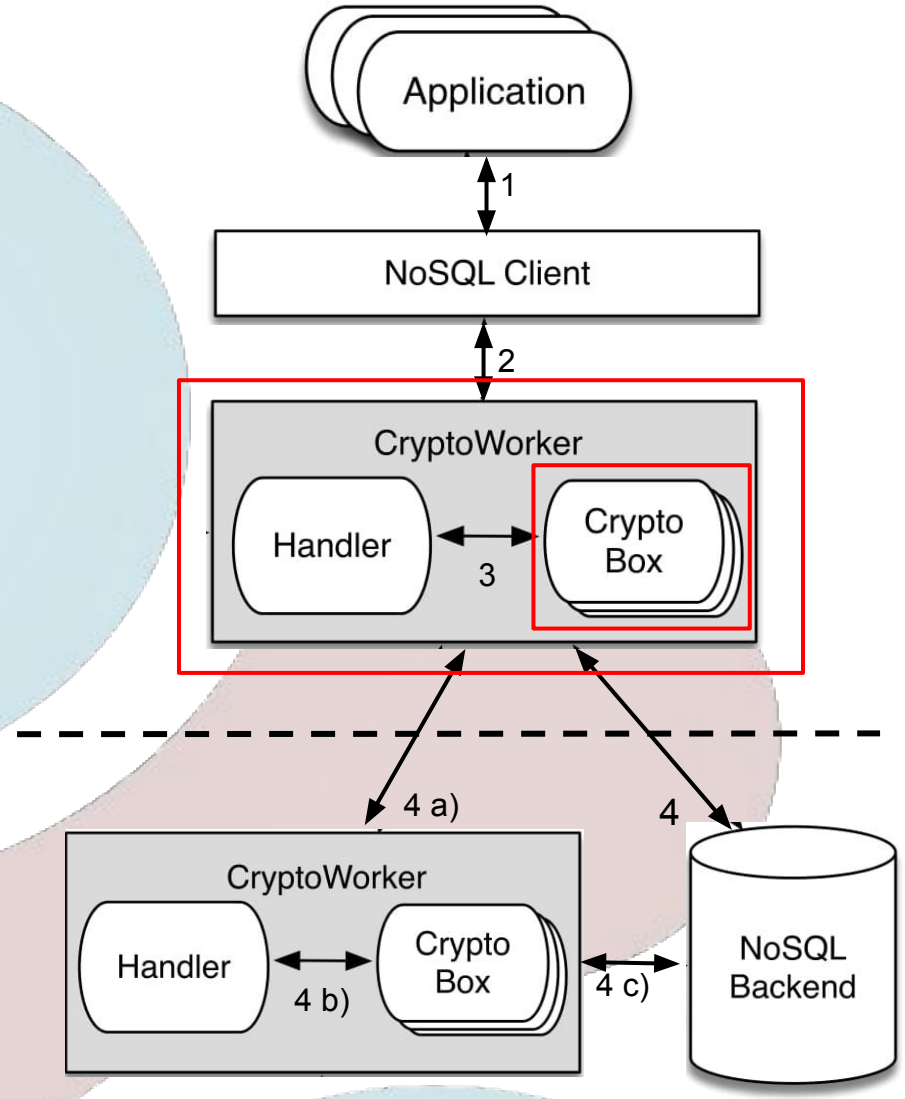
SafeNoSQL: Architecture



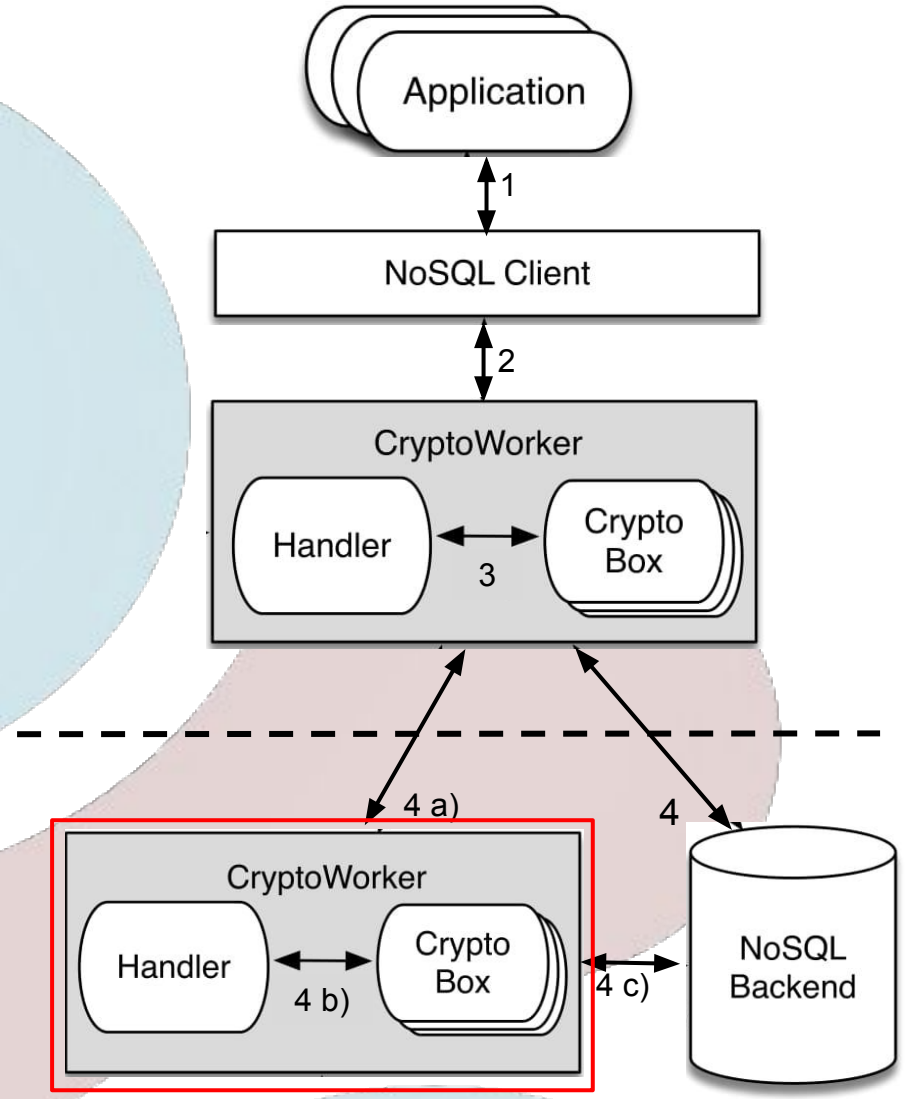
SafeNoSQL: Architecture



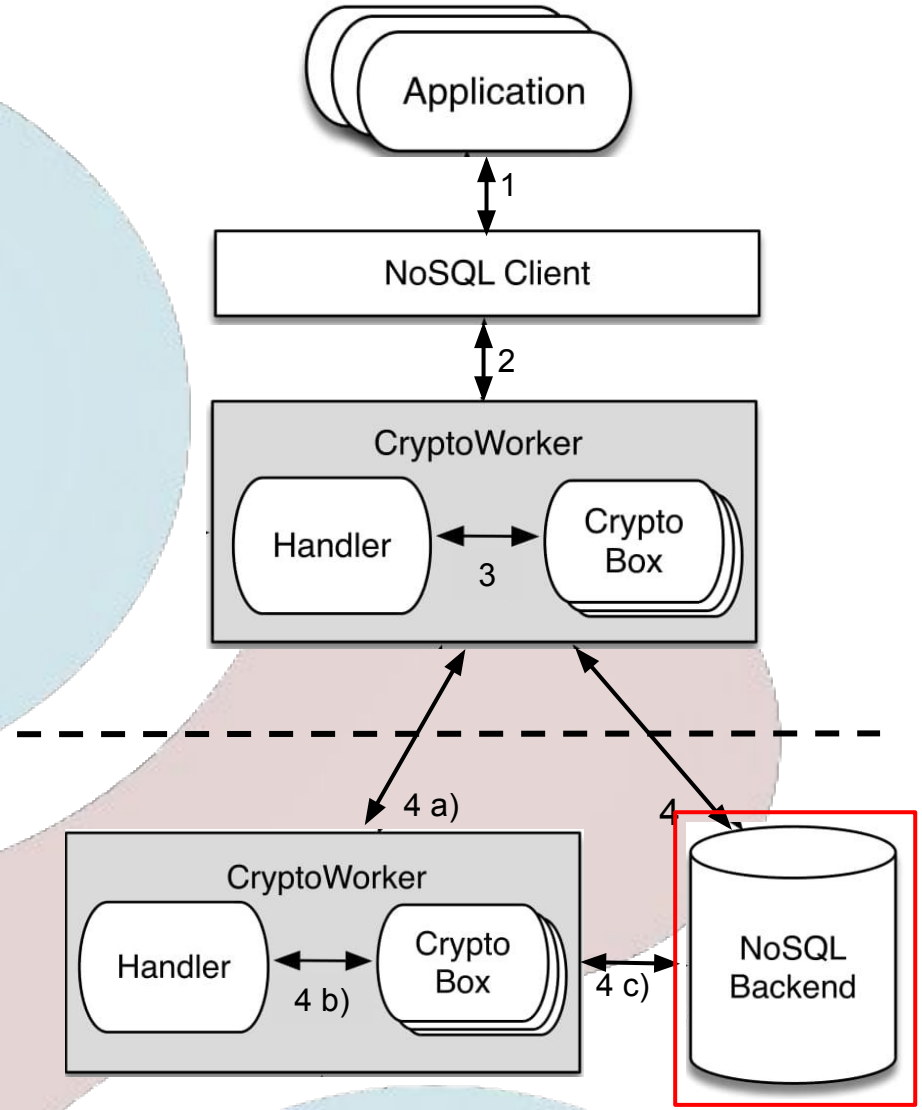
SafeNoSQL: Architecture



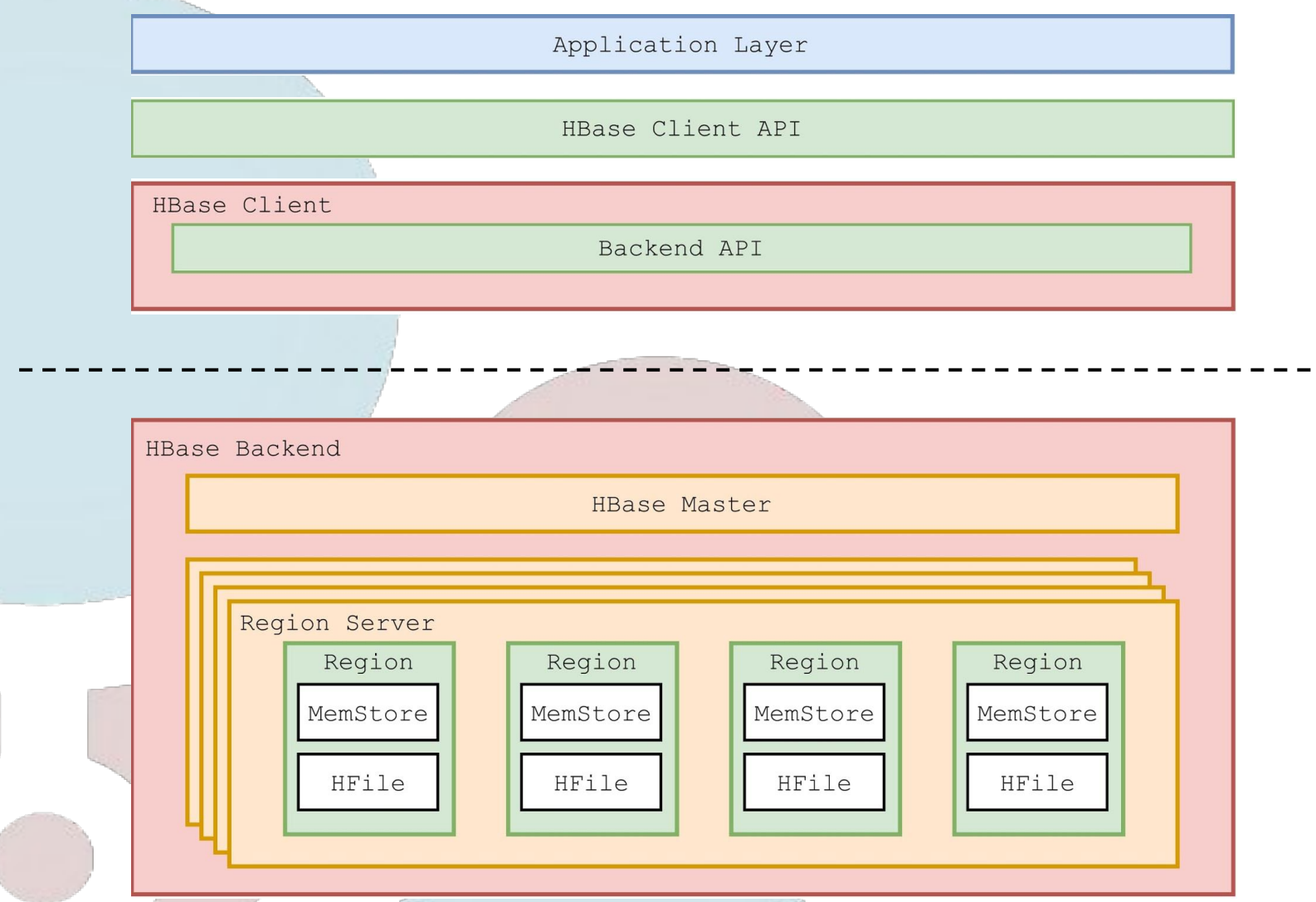
SafeNoSQL: Architecture



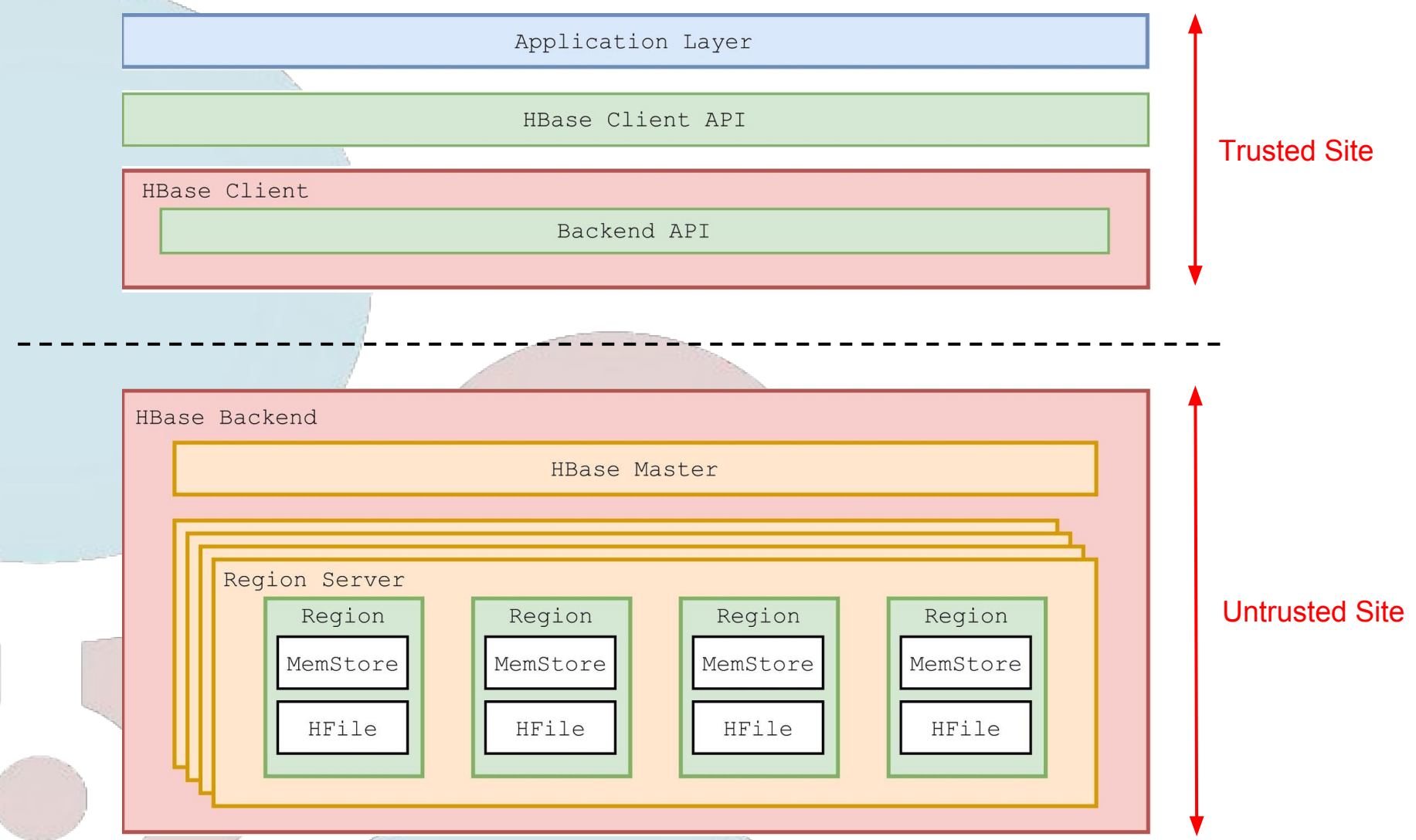
SafeNoSQL: Architecture



Apache HBase



Apache HBase

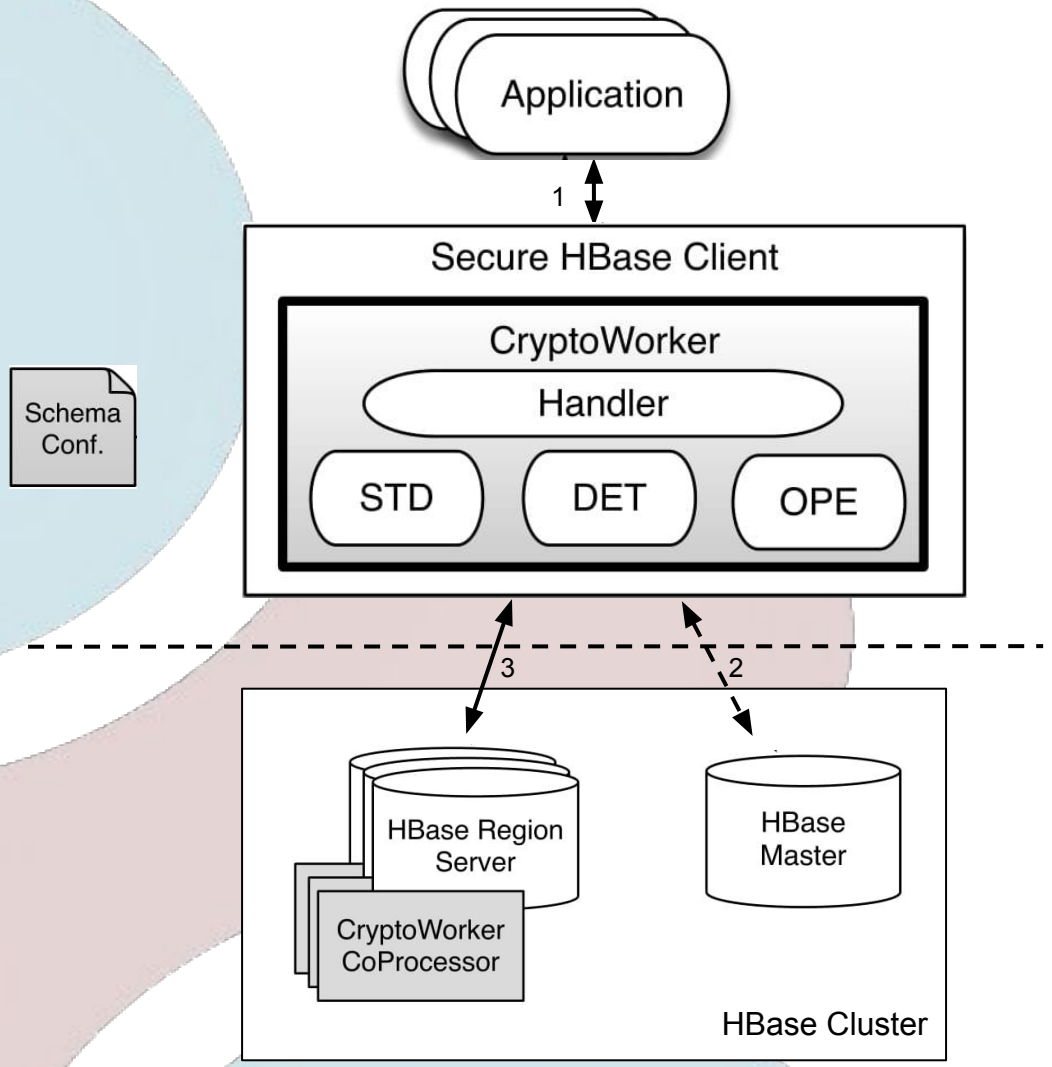


HBase Table

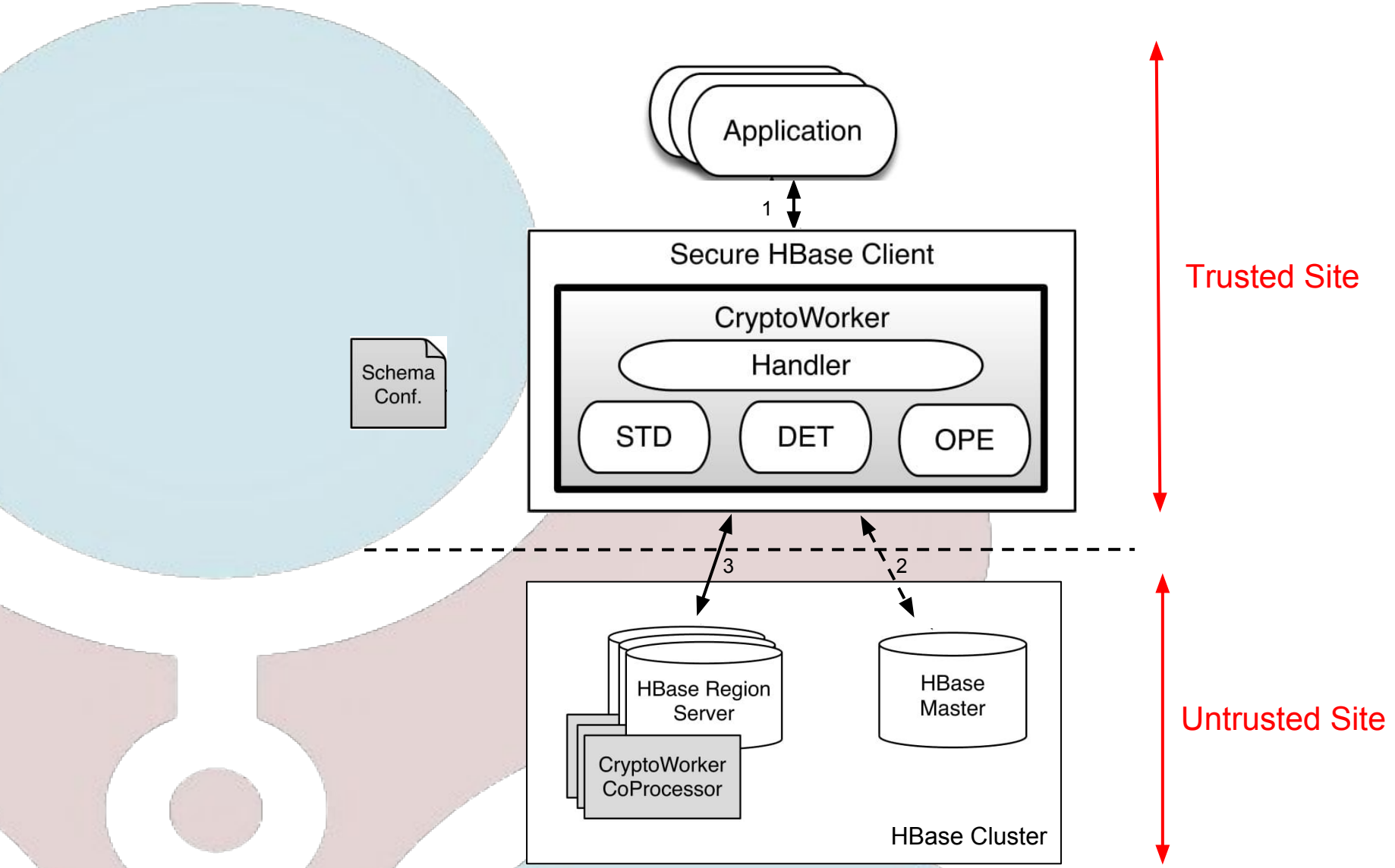
Key	Name (CF)		Contacts (CF)		
	First (CQ)	Last (CQ)	Phone Number(CQ)	Mobile (CQ)	Email (CQ)
1627	John	Doe	(800) 609-2233	(800) 420-1372	jdoe@gmail.com
1821	Anna	Far	(202) 513-4280	(202) 698-3281	far_a@gmail.com

Key - Row Key; CF - Column Family; CQ - Column Qualifier

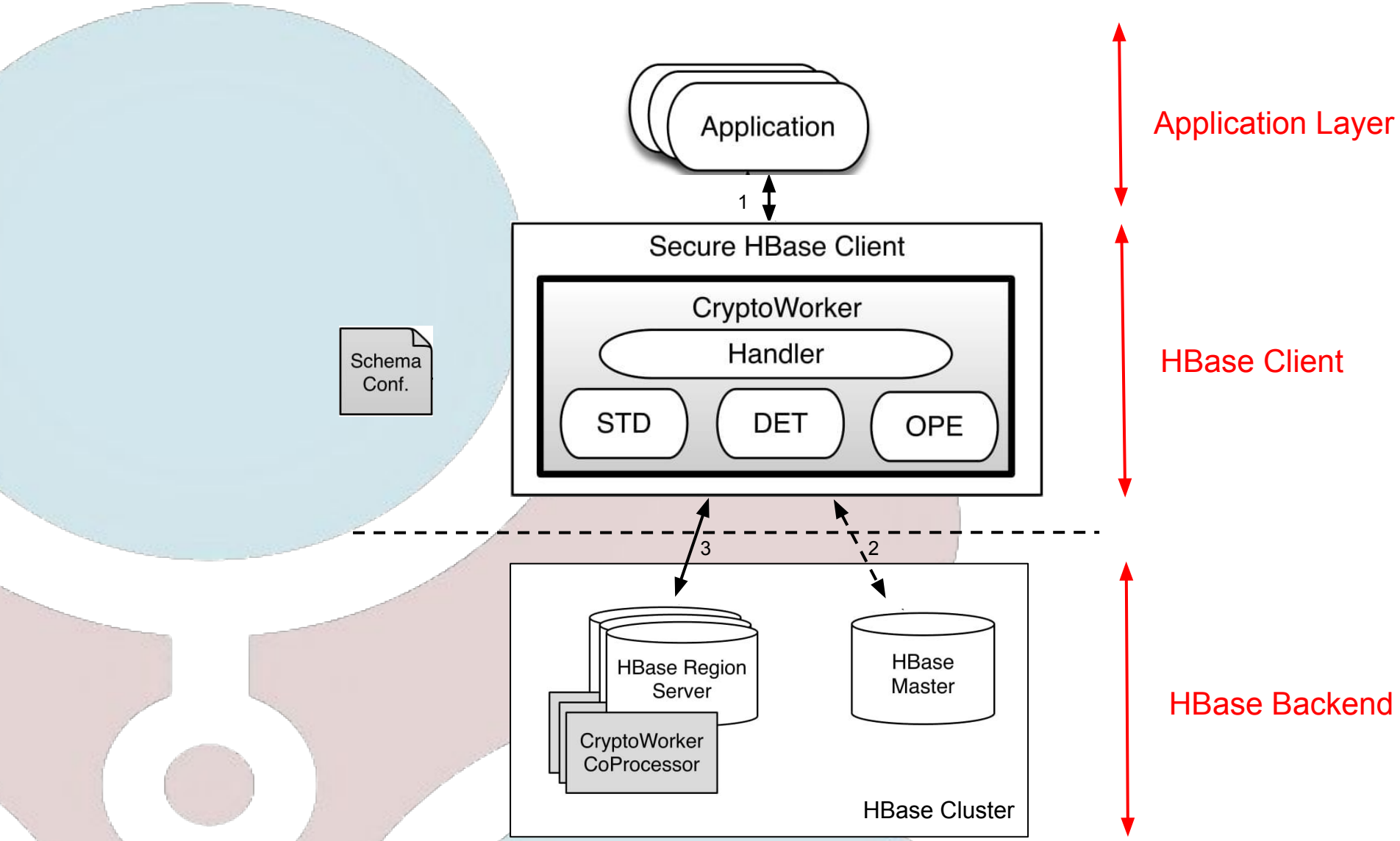
SafeNoSQL: Implementation



SafeNoSQL: Implementation

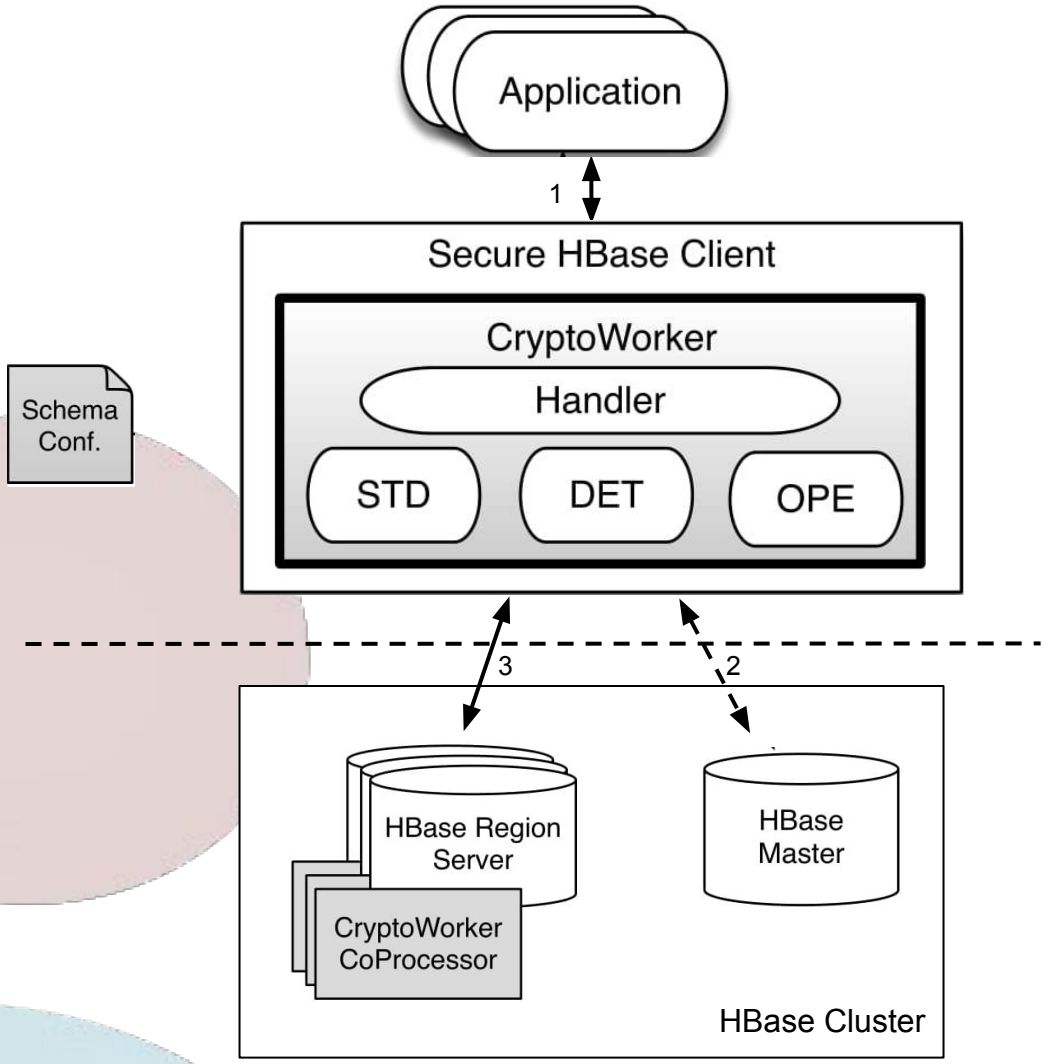


SafeNoSQL: Implementation



SafeNoSQL: Operation flow

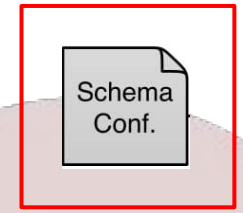
RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	42331	Osteopathy	2017-09-27 2:30PM
57810	65240	Dental	2017-09-30 1:00PM
83921	15421	Cardiology	2017-09-18 10:00AM



SafeNoSQL: Operation flow

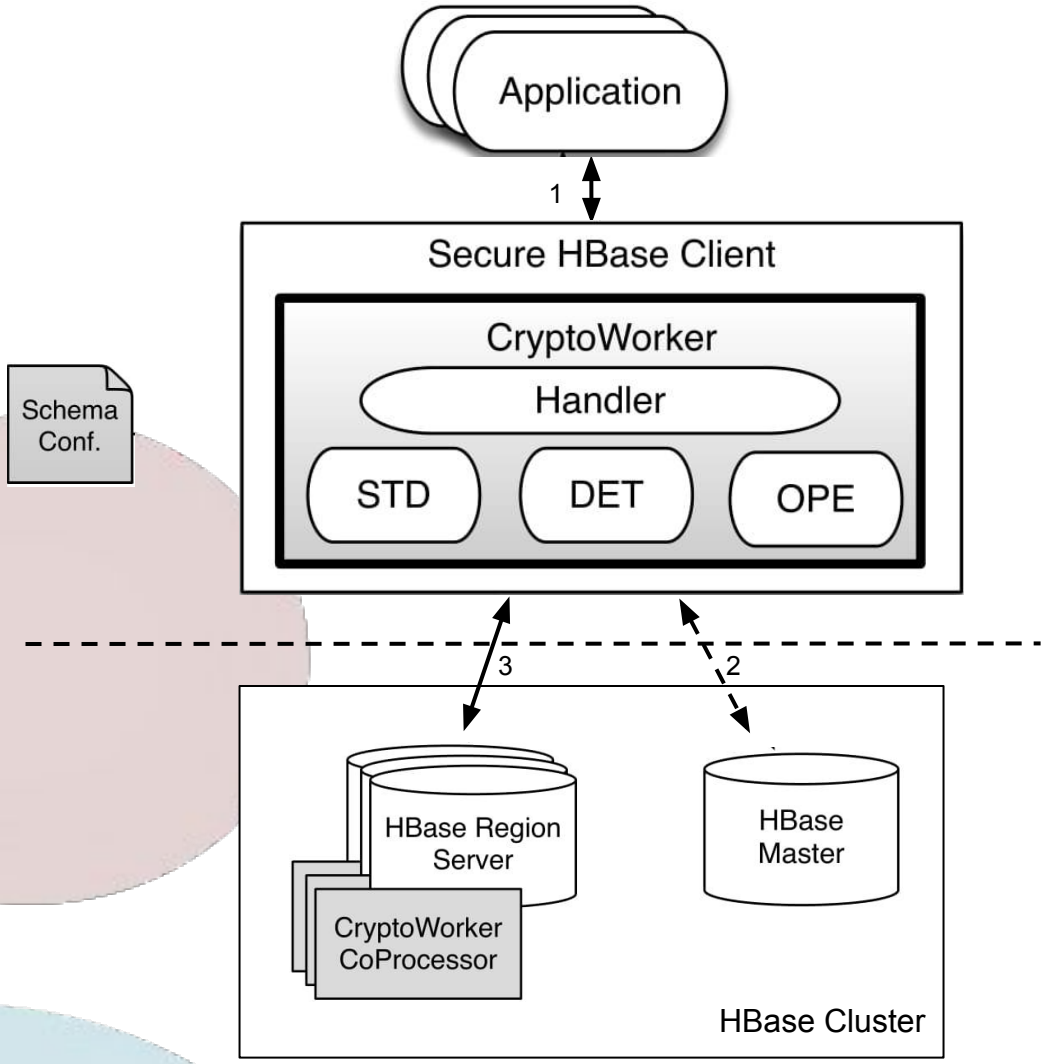
DET STD OPE

RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	42331	Osteopathy	2017-09-27 2:30PM
57810	65240	Dental	2017-09-30 1:00PM
83921	15421	Cardiology	2017-09-18 10:00AM



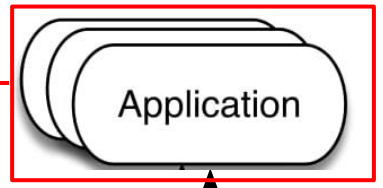
SafeNoSQL: Operation flow

RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d

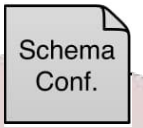


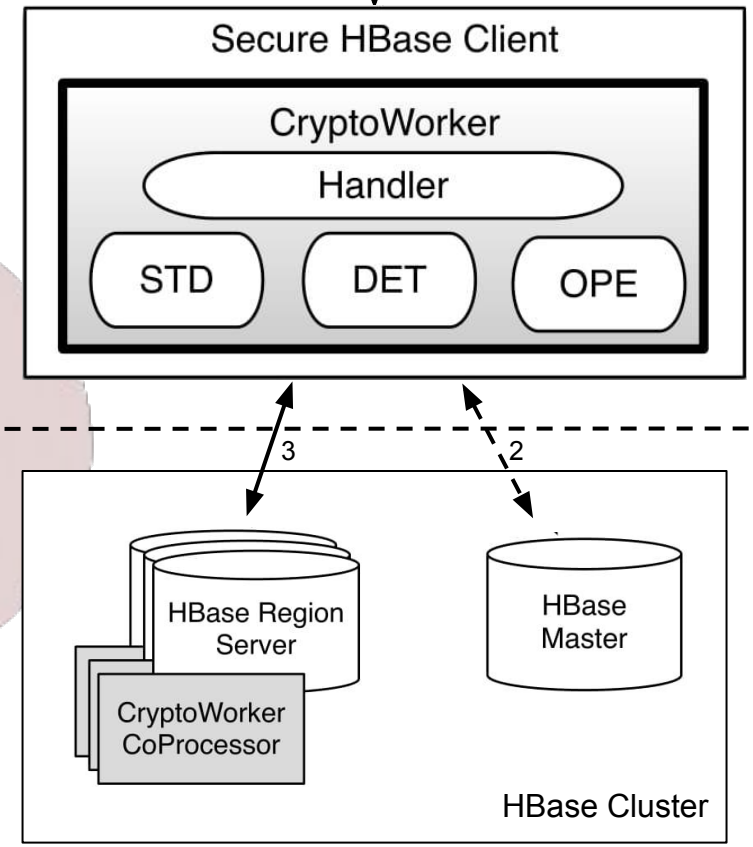
SafeNoSQL: Operation flow

Scan(SingleColumnValueFilter(*'Appointments'*, *'Date'*, ≥, *'2017-09-27'*))



RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d



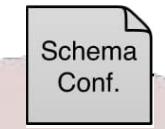


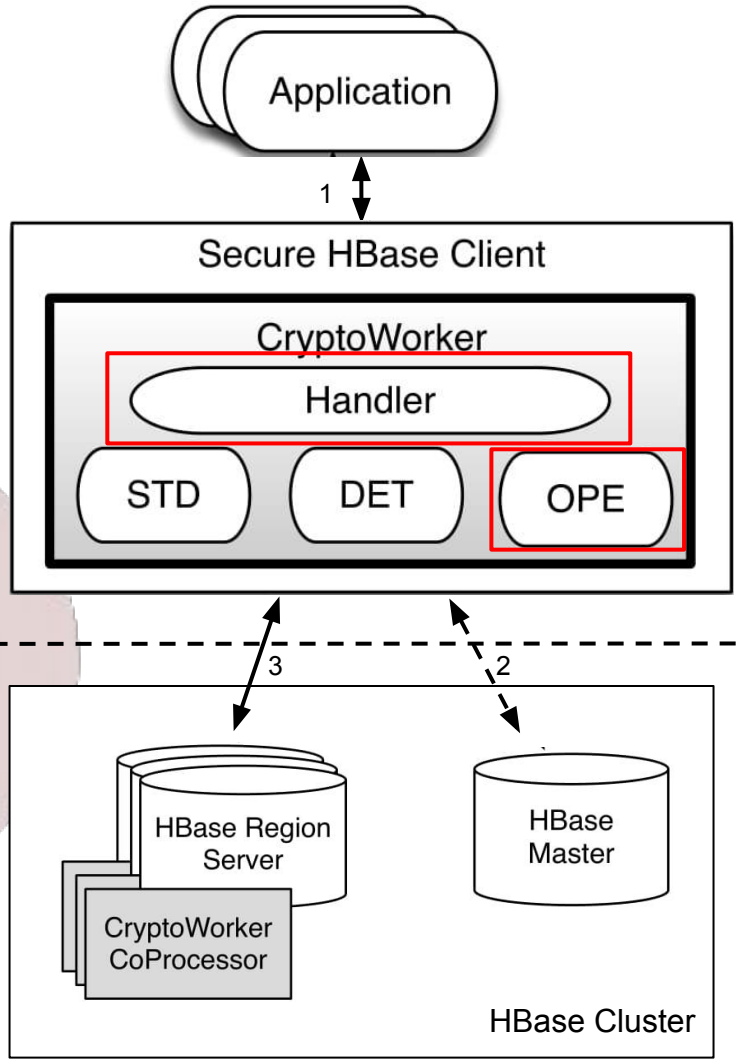
SafeNoSQL: Operation flow

x35903643..0f

Scan(SingleColumnValueFilter(*'Appointments'*, *'Date'*, ≥, OPE_{Enc}(key,*'2017-09-27'*)))

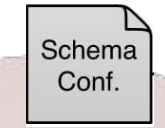
RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d

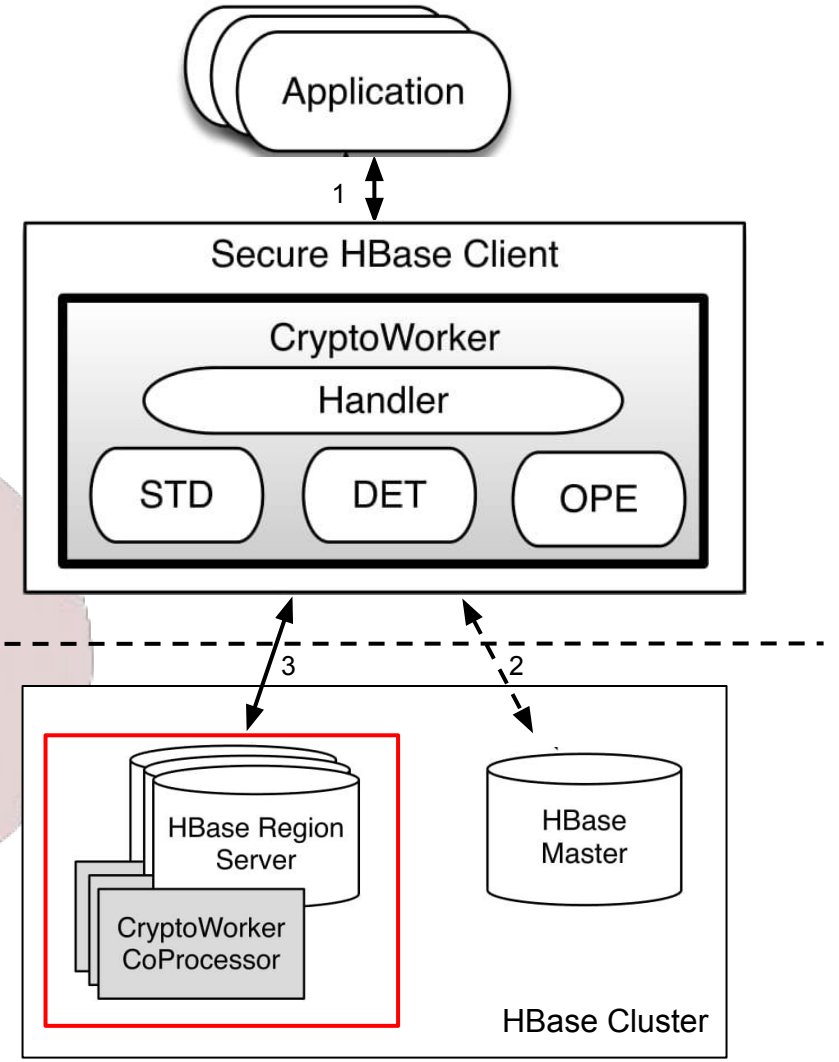




SafeNoSQL: Operation flow

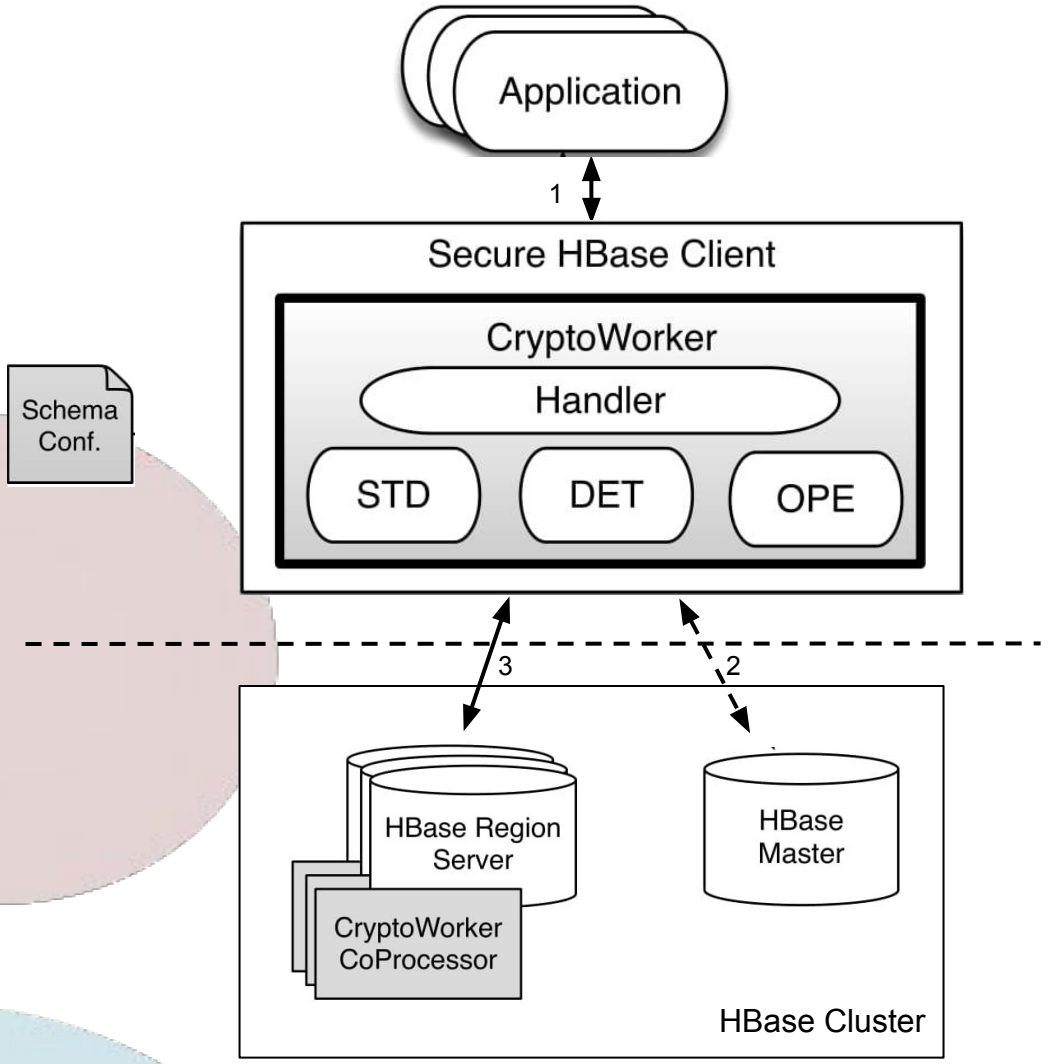
RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d





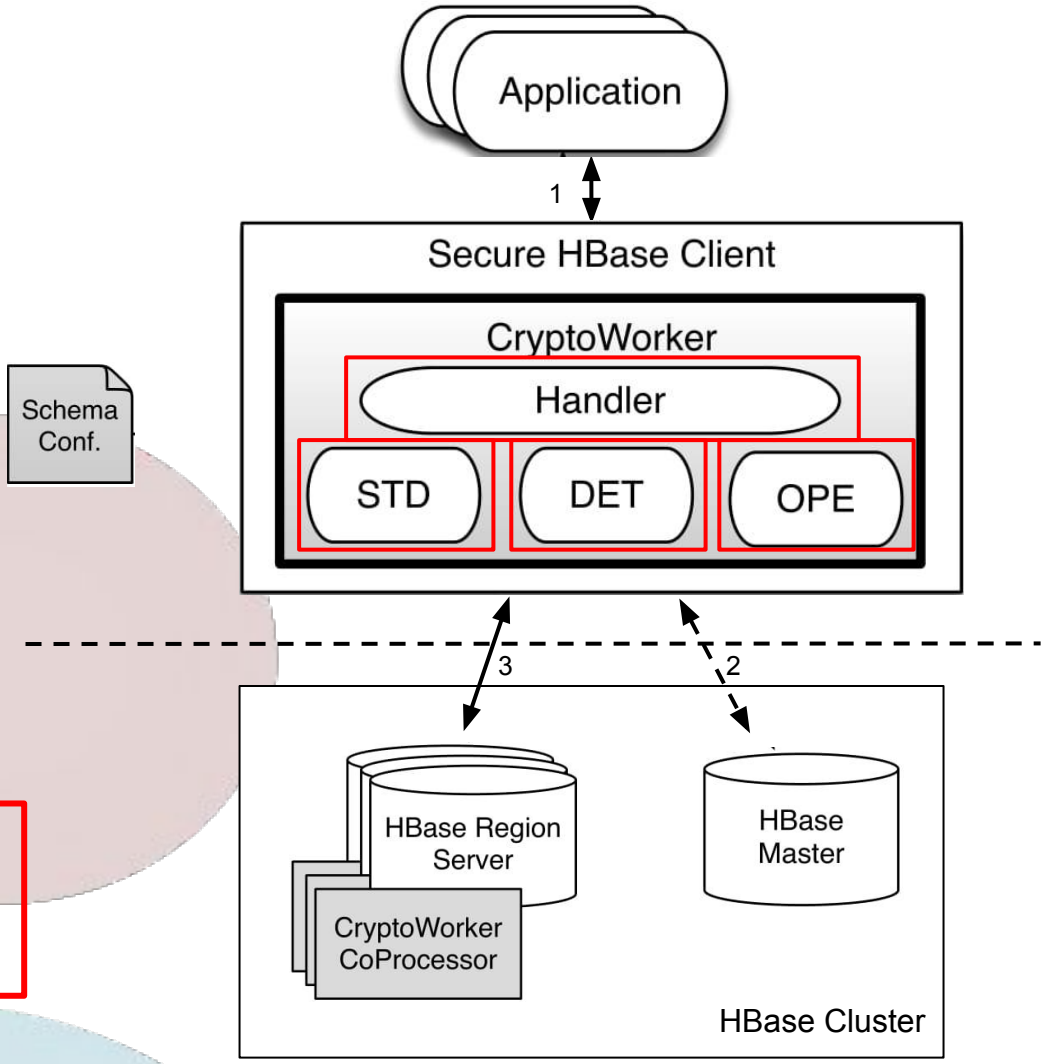
SafeNoSQL: Operation flow

RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d



SafeNoSQL: Operation flow

RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d

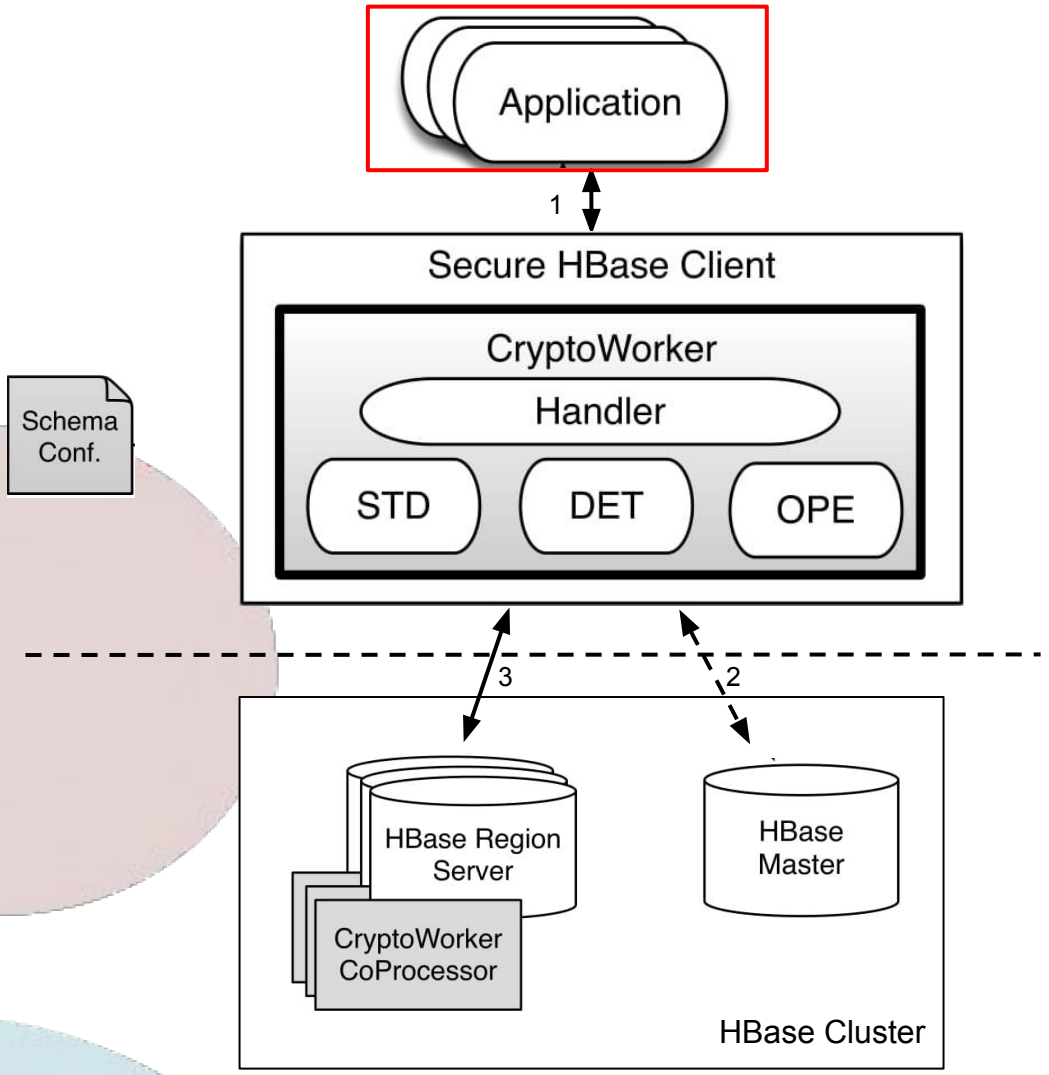


RowKey, Physician:PhysicianID, Appointments:Type, Appointments:Date	PLT(12345), DET_{Dec}(key, x7f199fb6..da), STD_{Dec}(key, xfe6477f3..85), OPE_{Dec}(key, x360e6ef4..7d)

SafeNoSQL: Operation flow

RowKey	Physician	Appointments	
	PhysicianID	Type	Date
12345	x7f199fb6..da	xfe6477f3..85	x360e6ef4..7d
57810	x6c6d5cd2..bf	x9aea3ee1..19	x369e3e92..65
83921	x67ddaf77..2b	xbe5275d5..ad	x3533b639..2d

RowKey, Physician:PhysicianID, Appointments:Type, Appointments:Date	12345, 42331, Osteopathy, 2017-09-27 2:30PM



Experimental Evaluation

- Micro- and macro-experiments
- 5 HBase nodes and a single client
- Database pre-populated with 10^6 rows
- YCSB as benchmarking platform
- 5 executions for each workload
- 20 minutes per execution

QEF - Qualifier Equality Filter
QRF - Qualifier Range Filter

Workload	Insert	Update	RMW	Read	Scan	QEF	QRF
A	-	50%	-	50%	-	-	-
B	-	5%	-	95%	-	-	-
E_1	5%	-	-	-	75%	10%	10%
E_2	5%	-	-	-	75%	20%	-
F	-	-	50%	50%	-	-	-
G	50%	-	15%	15%	-	10%	10%
H	10%	-	45%	30%	-	15%	-

Experimental Evaluation

- Micro- and macro-experiments
- 5 HBase nodes and a single client
- Database pre-populated with 10^6 rows
- YCSB as benchmarking platform
- 5 executions for each workload
- 20 minutes per execution

QEF - Qualifier Equality Filter
QRF - Qualifier Range Filter

Workload	Insert	Update	RMW	Read	Scan	QEF	QRF
A	-	50%	-	50%	-	-	-
B	-	5%	-	95%	-	-	-
E_1	5%	-	-	-	75%	10%	10%
E_2	5%	-	-	-	75%	20%	-
F	-	-	50%	50%	-	-	-
G	50%	-	15%	15%	-	10%	10%
H	10%	-	45%	30%	-	15%	-

Experimental Evaluation

- Micro- and macro-experiments
- 5 HBase nodes and a single client
- Database pre-populated with 10^6 rows
- YCSB as benchmarking platform
- 5 executions for each workload
- 20 minutes per execution

QEF - Qualifier Equality Filter
QRF - Qualifier Range Filter

Workload	Insert	Update	RMW	Read	Scan	QEF	QRF
A	-	50%	-	50%	-	-	-
B	-	5%	-	95%	-	-	-
E_1	5%	-	-	-	75%	10%	10%
E_2	5%	-	-	-	75%	20%	-
F	-	-	50%	50%	-	-	-
G	50%	-	15%	15%	-	10%	10%
H	10%	-	45%	30%	-	15%	-

Experimental Evaluation: Database Schemas

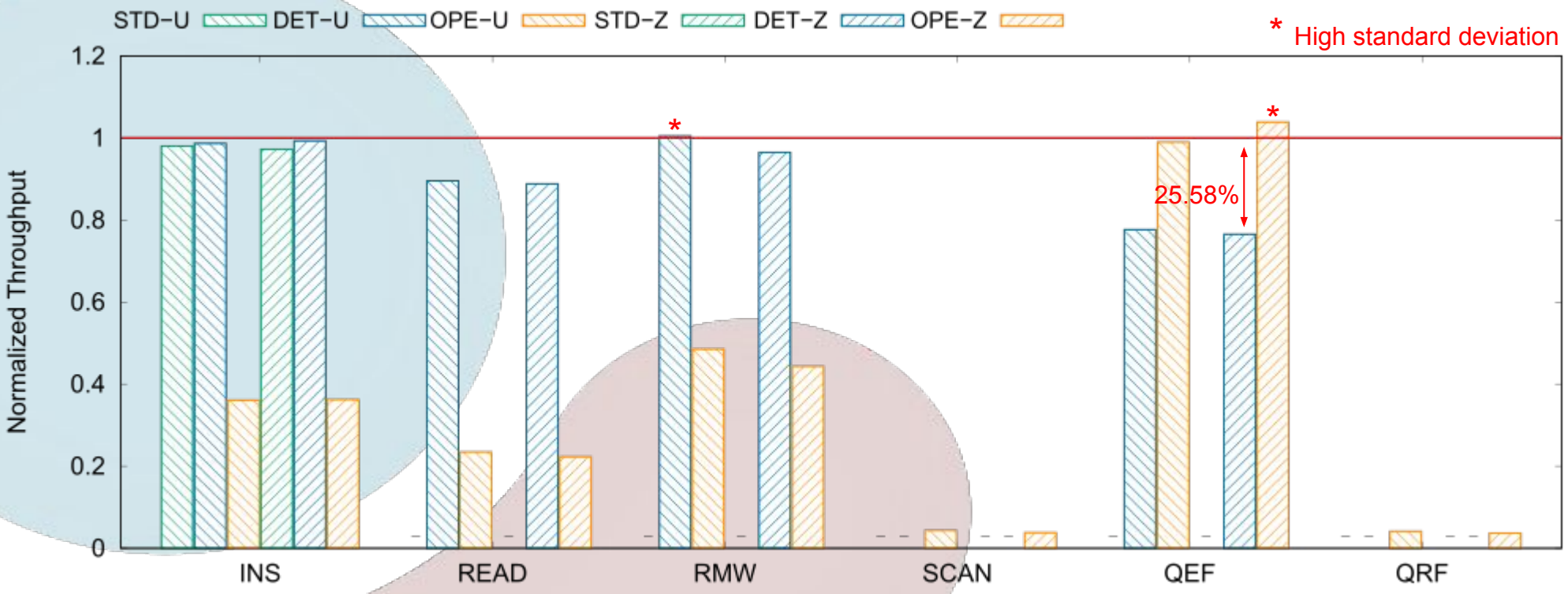
Personal Information about Hospital Patients

Key	Identification						Contacts		Obs.	App.
	MainID	Surname	Name	Birth	Nat.	C. ID	Address	Contact	Obs.	[1-*]
8	64	64	64	14	4	9	256	13	1024	8
DET	DET	STD	STD	STD	STD	STD	STD	STD	STD	STD

Hospital Appointments

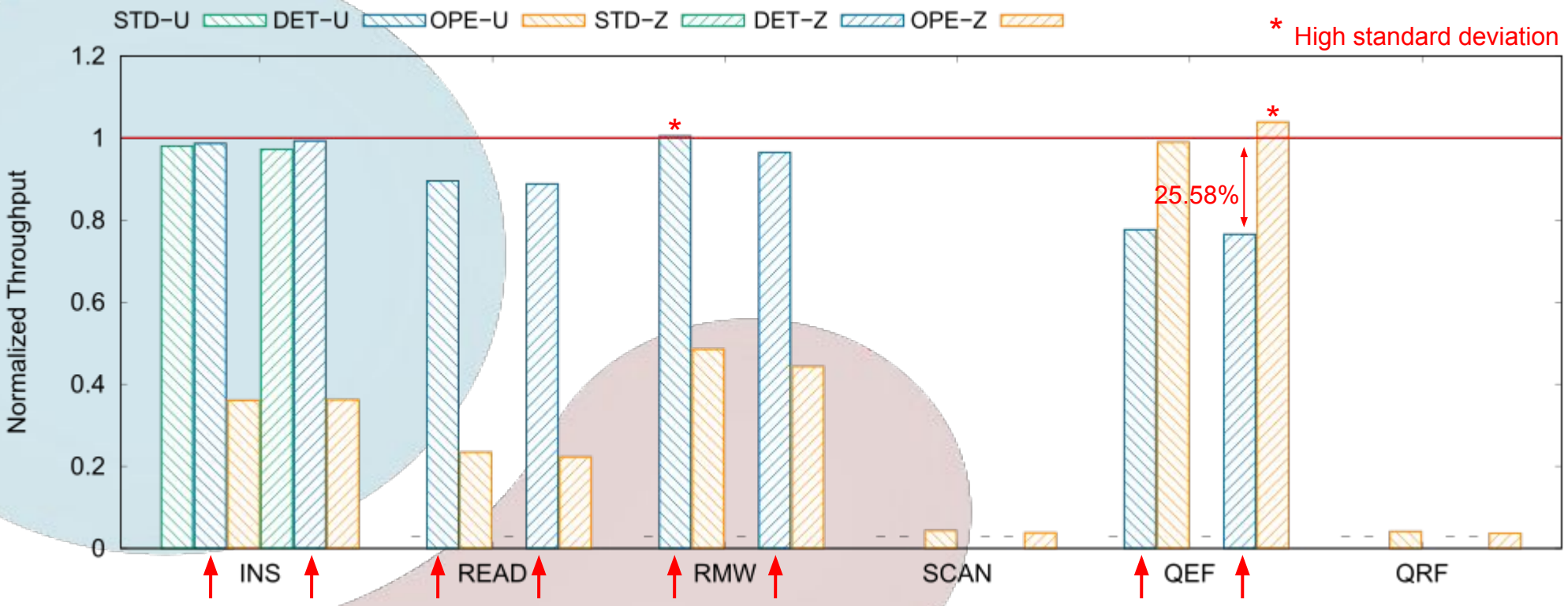
Key	Physician	Patient	Appointment				Institution	
	PhysicianID	PatientID	Date	Date-STD	Type	Obs.	Name	Address
8	16	16	14	14	64	1024	128	256
DET	DET	STD	OPE	STD	STD	STD	STD	STD

Experimental Evaluation: Micro Tests



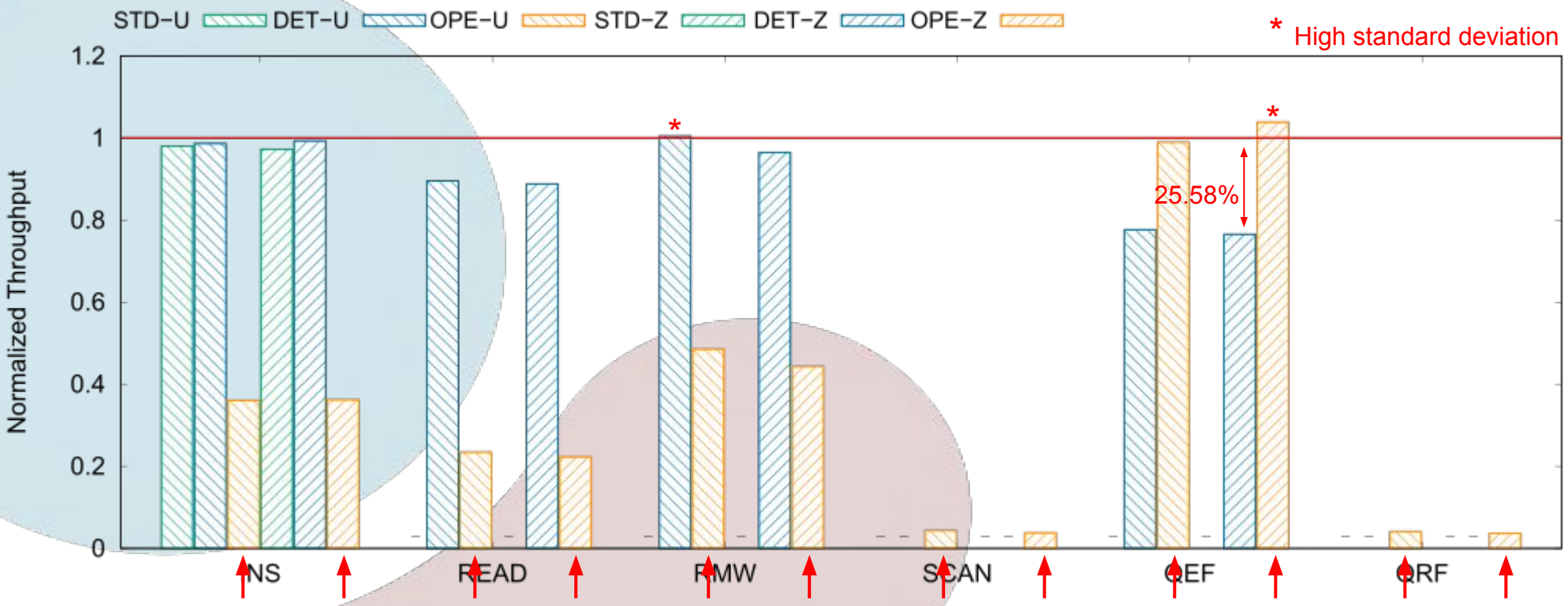
- Micro-experiments were performed for the *Appointments'* schema
- Only row-key encryption
- DET scheme reveals a performance loss less than 26%
- OPE operations' performance are restricted by the cipher's performance

Experimental Evaluation: Micro Tests



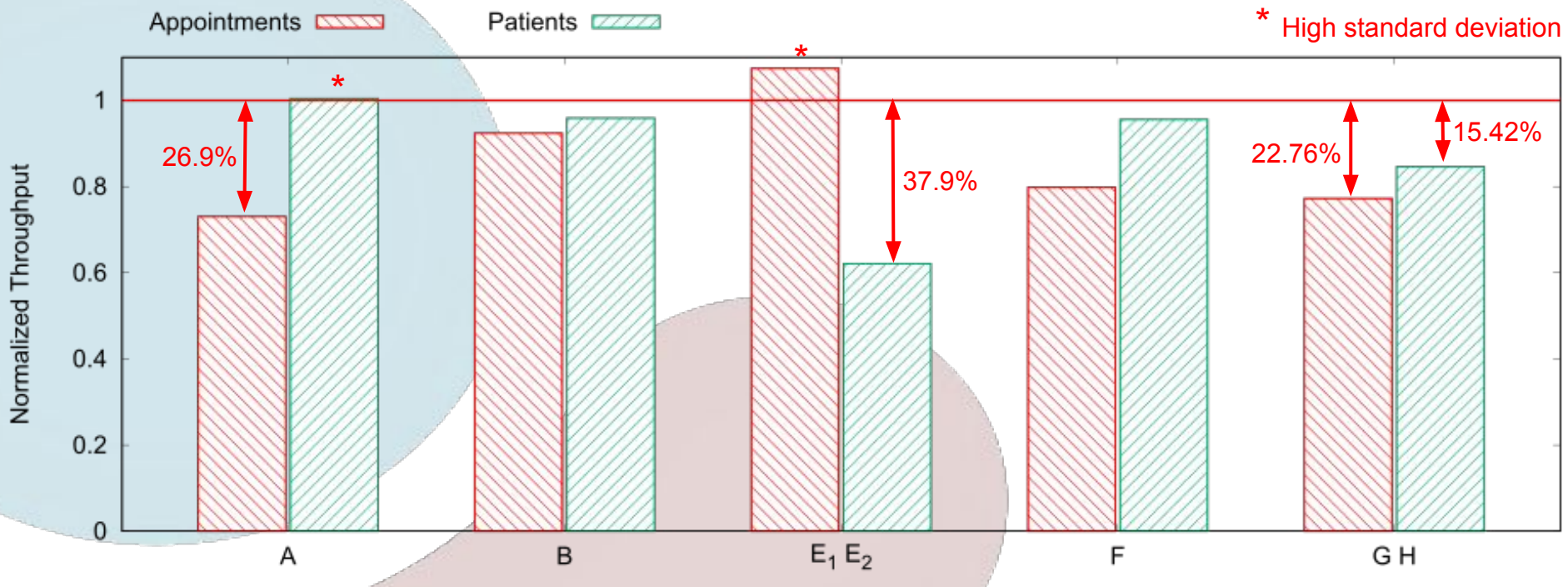
- Micro-experiments were performed for the *Appointments'* schema
- Only row-key encryption
- DET scheme reveals a performance loss less than 26%
- OPE operations' performance are restricted by the cipher's performance

Experimental Evaluation: Micro Tests



- Micro-experiments were performed for the *Appointments'* schema
- Only row-key encryption
- DET scheme reveals a performance loss less than 26%
- OPE operations' performance are restricted by the cipher's performance

Experimental Evaluation: Macro Tests



- Average performance loss
  - 14.03% for the *Appointments* schema
  - 12.29% for *Patients* schema
- Appropriate trade-offs balancing

Conclusion

- Strict combination of encryption schemes cannot fulfill the user's requirements

- **SafeNoSQL,** a privacy-preserving framework for NoSQL databases
  - High modularity and flexibility
  - Generic to most of NoSQL Key-Value Stores
  - Extensible to several encryption schemes
  - Performance overhead of 15% (in average)

Future Work

- Integration with more encryption schemes
- Support of more NoSQL databases
- Encryption keys management and access control
- Query planner and schema designer

A Practical Framework for Privacy-Preserving NoSQL Databases

36th IEEE International Symposium on Reliable Distributed Systems
Hong Kong, 27th September 2017

Ricardo Macedo[1], João Paulo[1], Rogério Pontes[1], Bernardo Portela[2],
Tiago Oliveira[2], Miguel Matos[3], Rui Oliveira[1]

[1] High Assurance Software Lab, INESC TEC and U. Minho, Portugal
[2] High Assurance Software Lab, INESC TEC and FCUP, Portugal
[3] INESC ID/IST, U. Lisboa, Portugal