The Next Service Wave: Prototyping Cloud-Native and Stealthy Applications

Josef Spillner <josef.spillner@zhaw.ch>
Service Prototyping Lab (blog.zhaw.ch/icclab)

Sep 9, 2015 | IBM Research Zürich

Who we are

Service Prototyping Lab @ ZHAW School of Engineering / InIT, Winterthur
- SPLab in conjunction with ICCLab

Özgür Özsu

Pietro Brossi

Dr. Josef Spillner

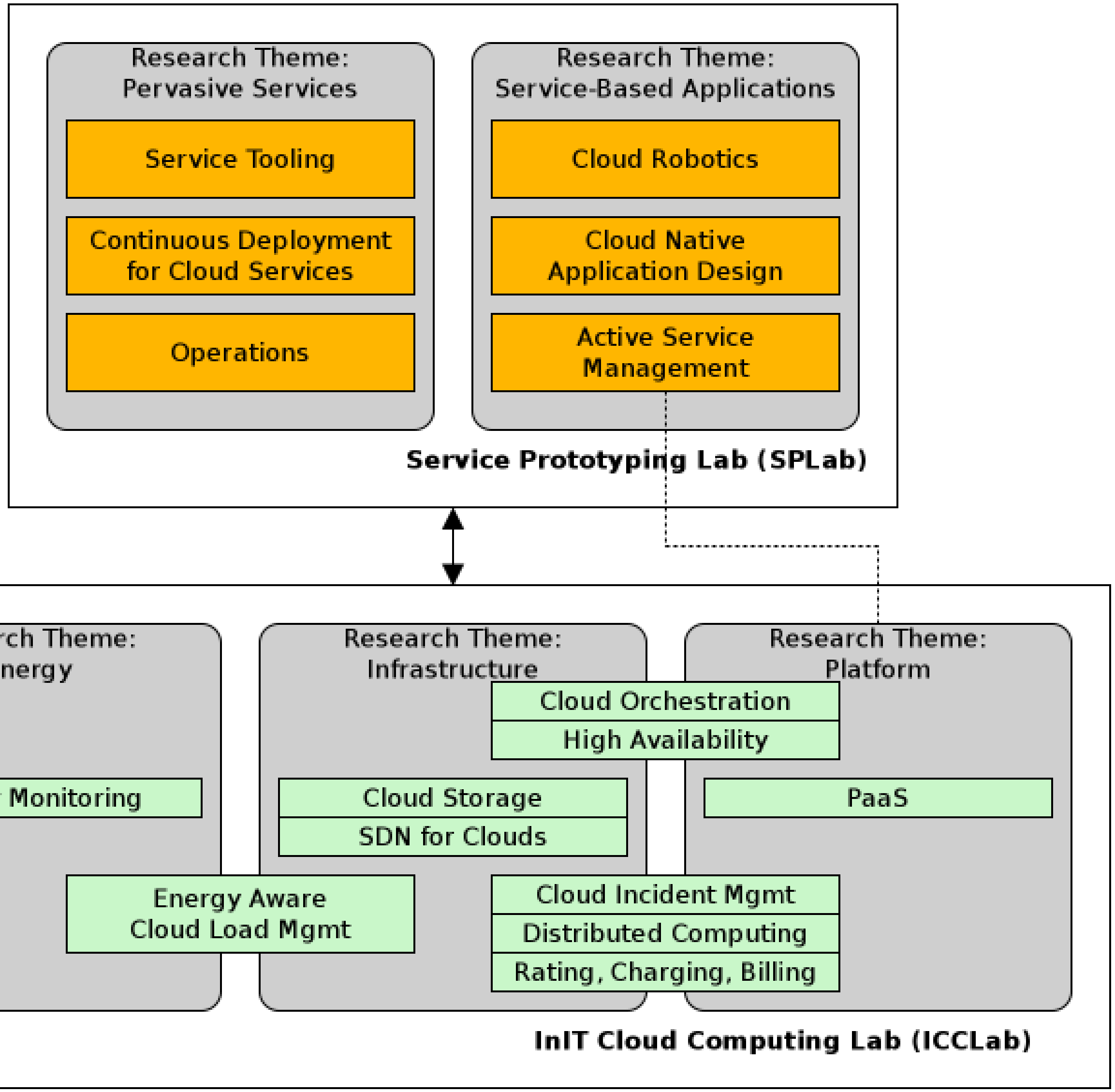
Christof Marti

Dr. Giovanni Toffetti Carughi

We are hiring :)

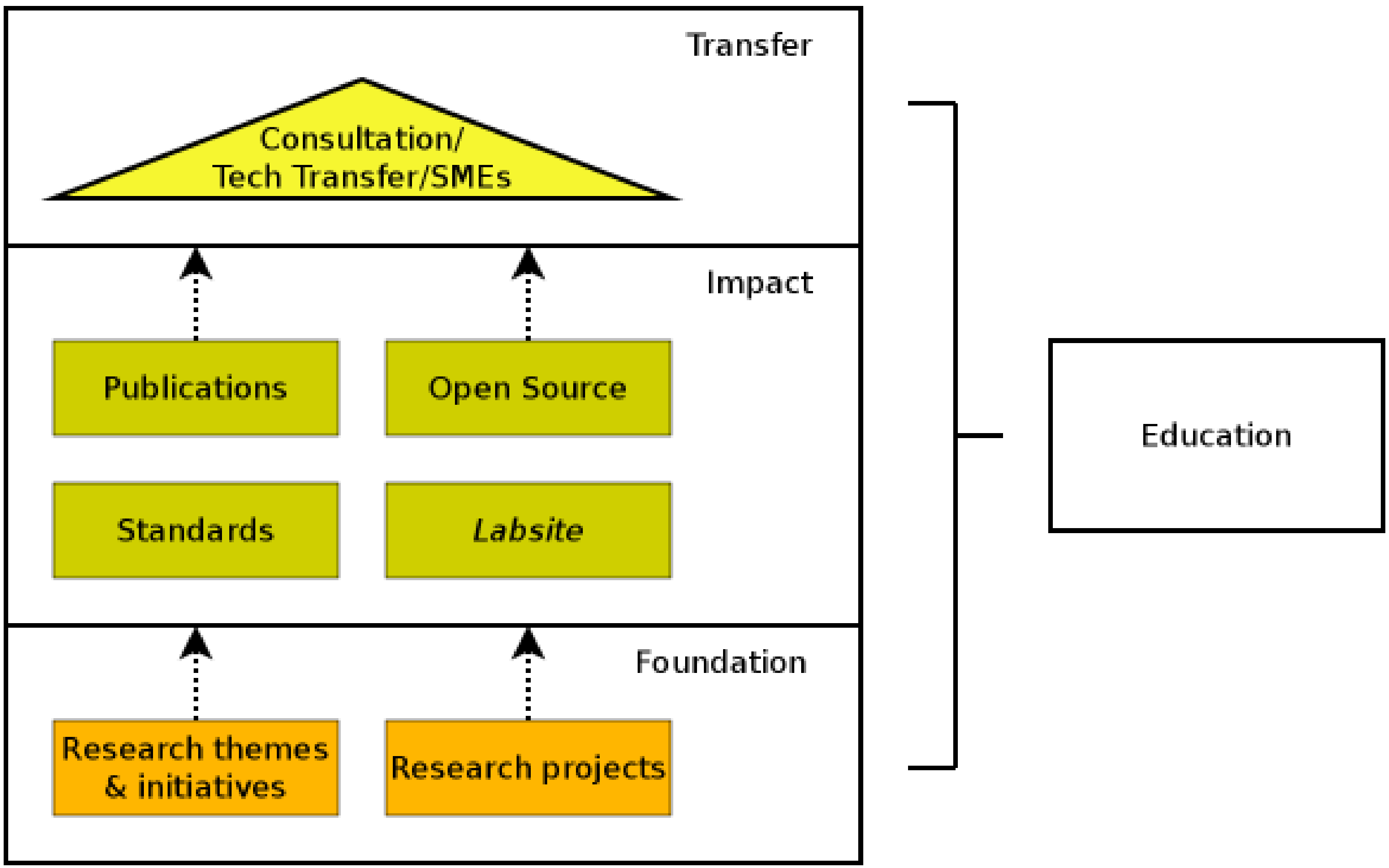
What we do

Themes, initiatives and projects



How we work

Fairly lab-style...



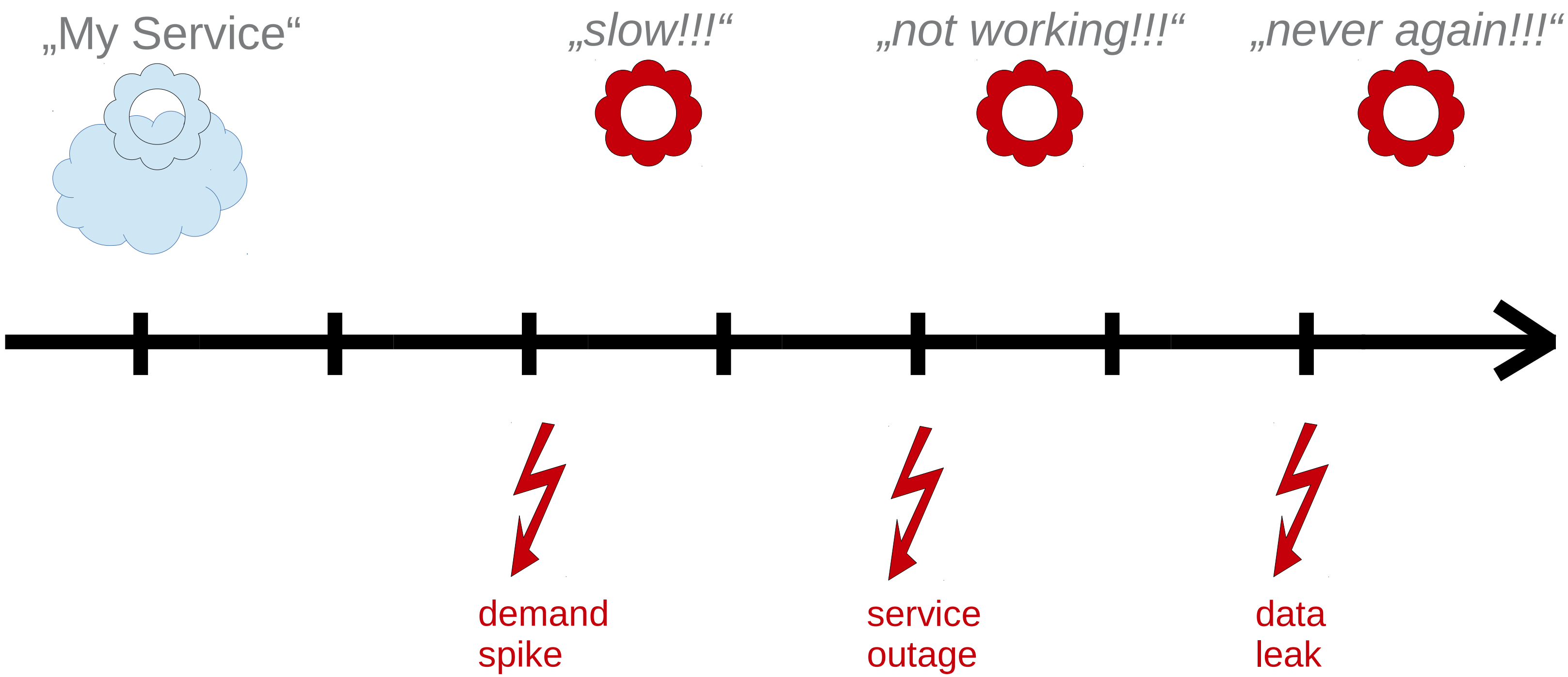
More information on SPLab...

Website: http://blog.zhaw.ch/icclab/

Microblog: Twitter @S_P_Lab

Cloud-native & stealthy applications

Motivation: Today's cloud applications and services



Analysis of protection goals and risks

Goals:

- availability
- confidentiality
- scalability
- reliability
- resilience

Risks:

- unavailability of dependency service
- permanent exit (e.g. business termination, bankruptcy)
- condition changes (e.g. price jumps)
- loss of data (& backup, too)
- leak of data

Constraints:

- re-usable solution

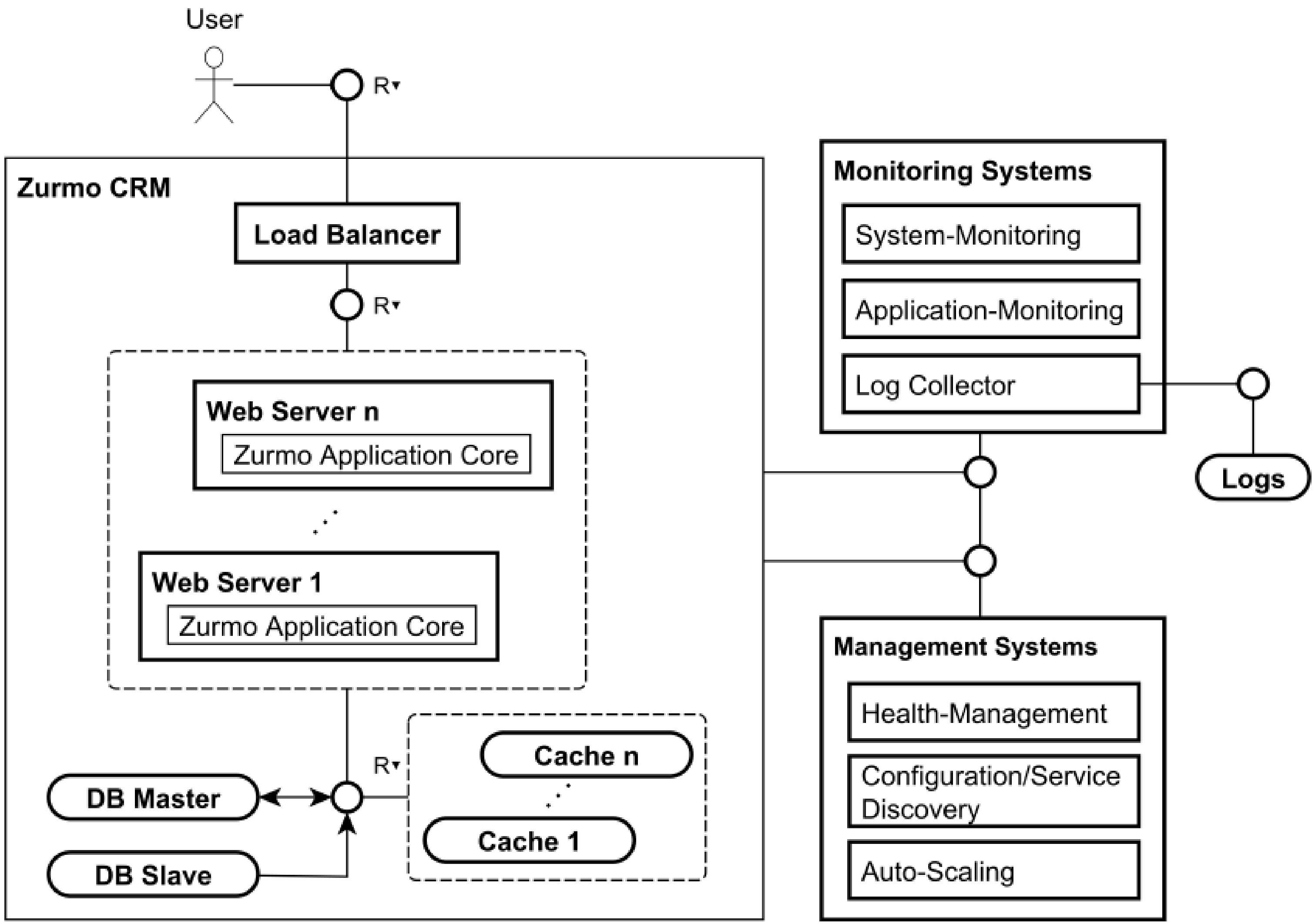
Cloud-native applications

Protection goals: Resilience and scalability

Focus: applications subdivided into microservices

Approach: Distributed consensus, container deployer and autoscaler

Cloud-native applications



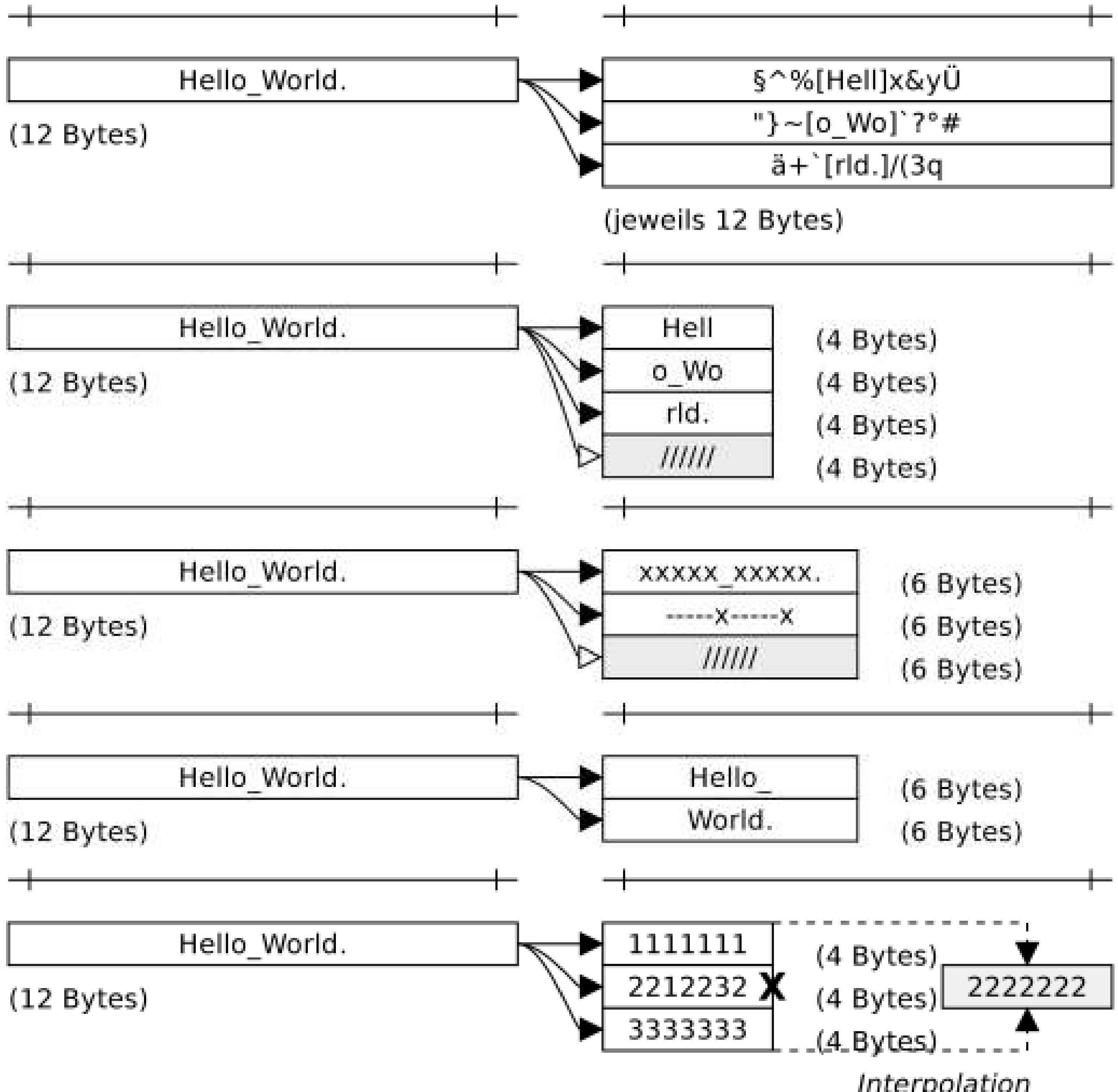
Stealth applications

Protection goals: Availability and confidentiality

Focus: data-processing applications

Approach: Operation-aware data coding, distribution and processing
(in one multi-service pipeline)

Data coding



secret sharing

erasure coding

replication

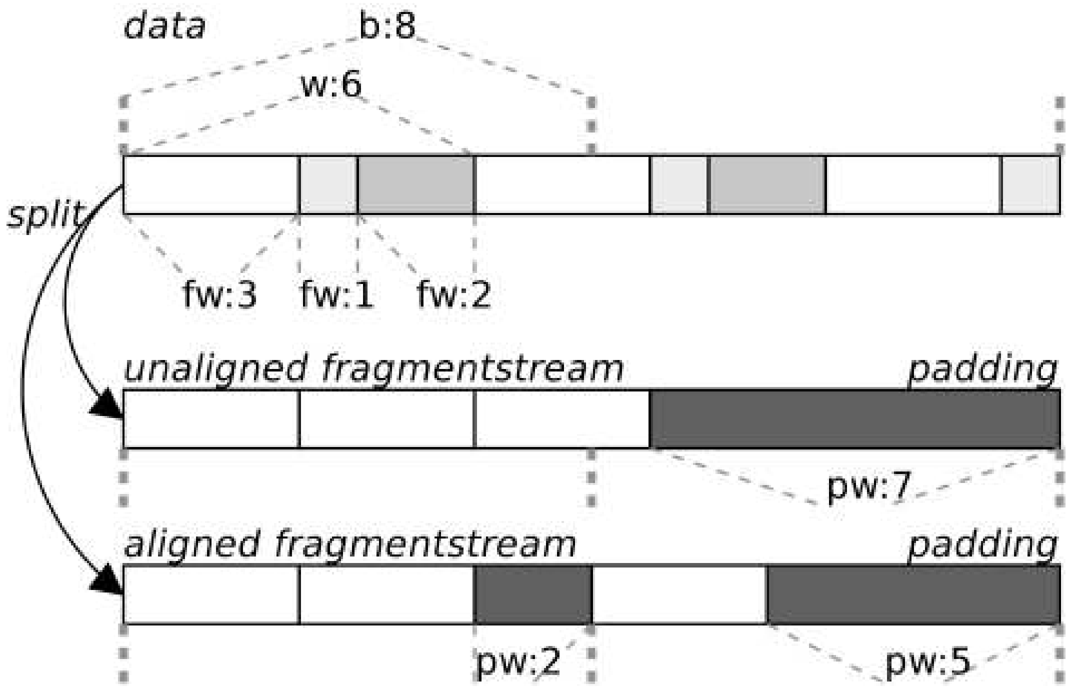
bit splitting

chunking

interpolation

SPLAB

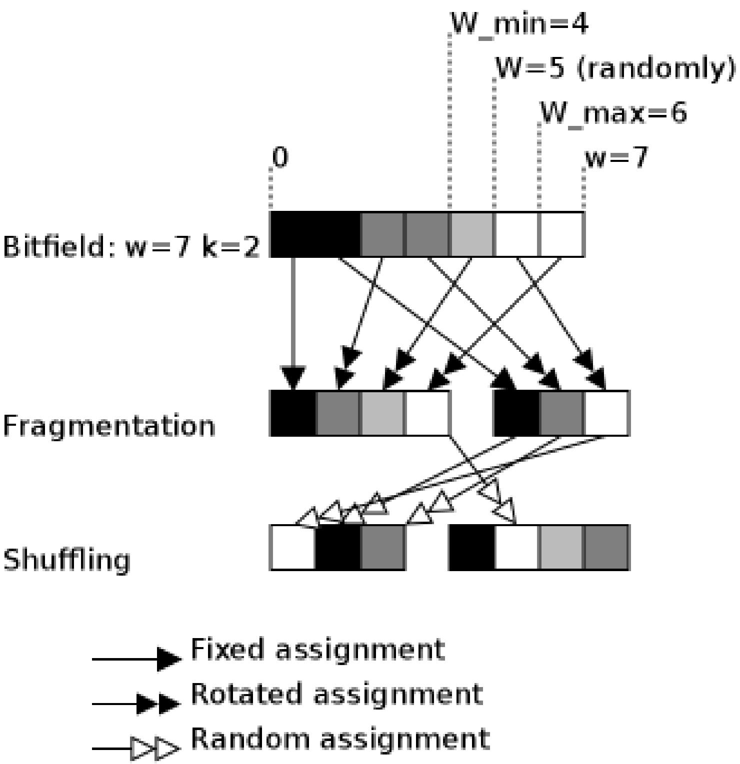
Data coding: Bitsplitting

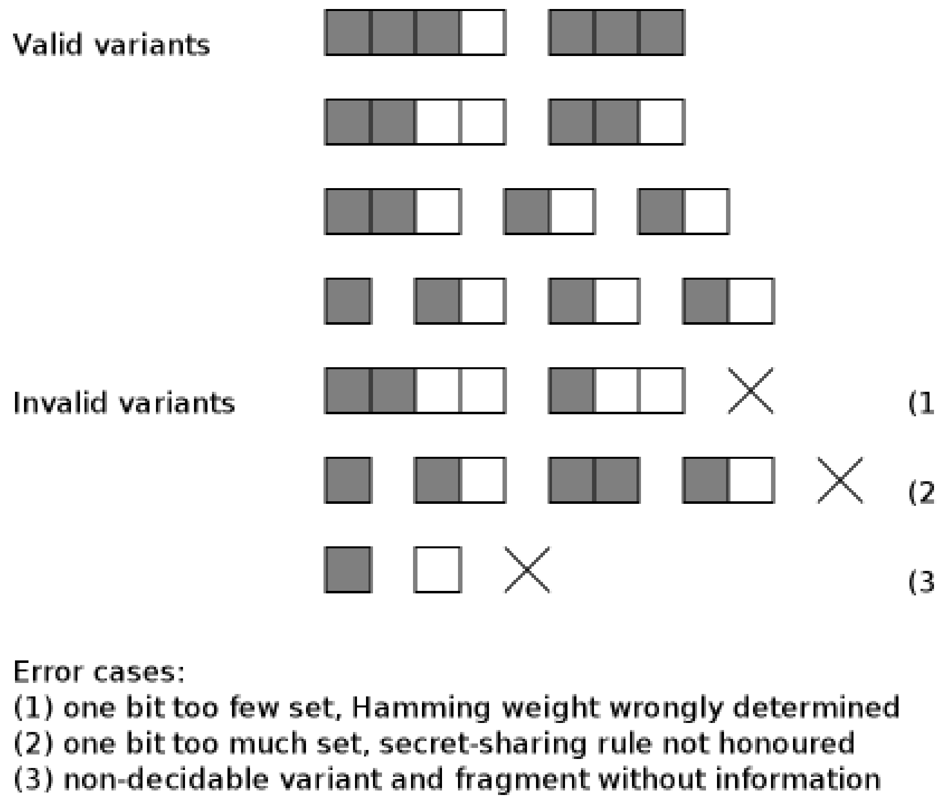


Implications for the application author:

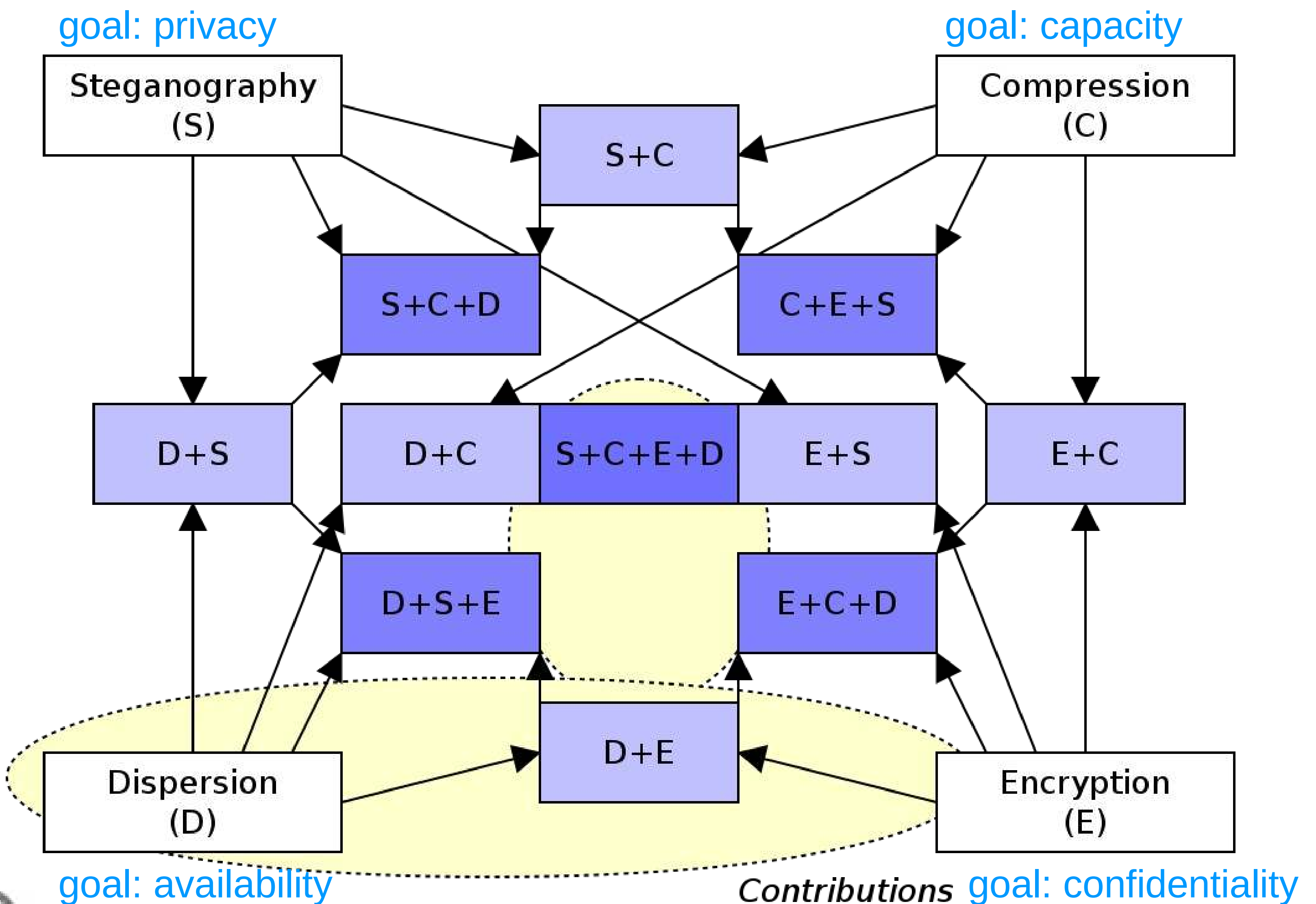
- should not need to worry
- advanced tooling needed

Data coding: Secret Sharing



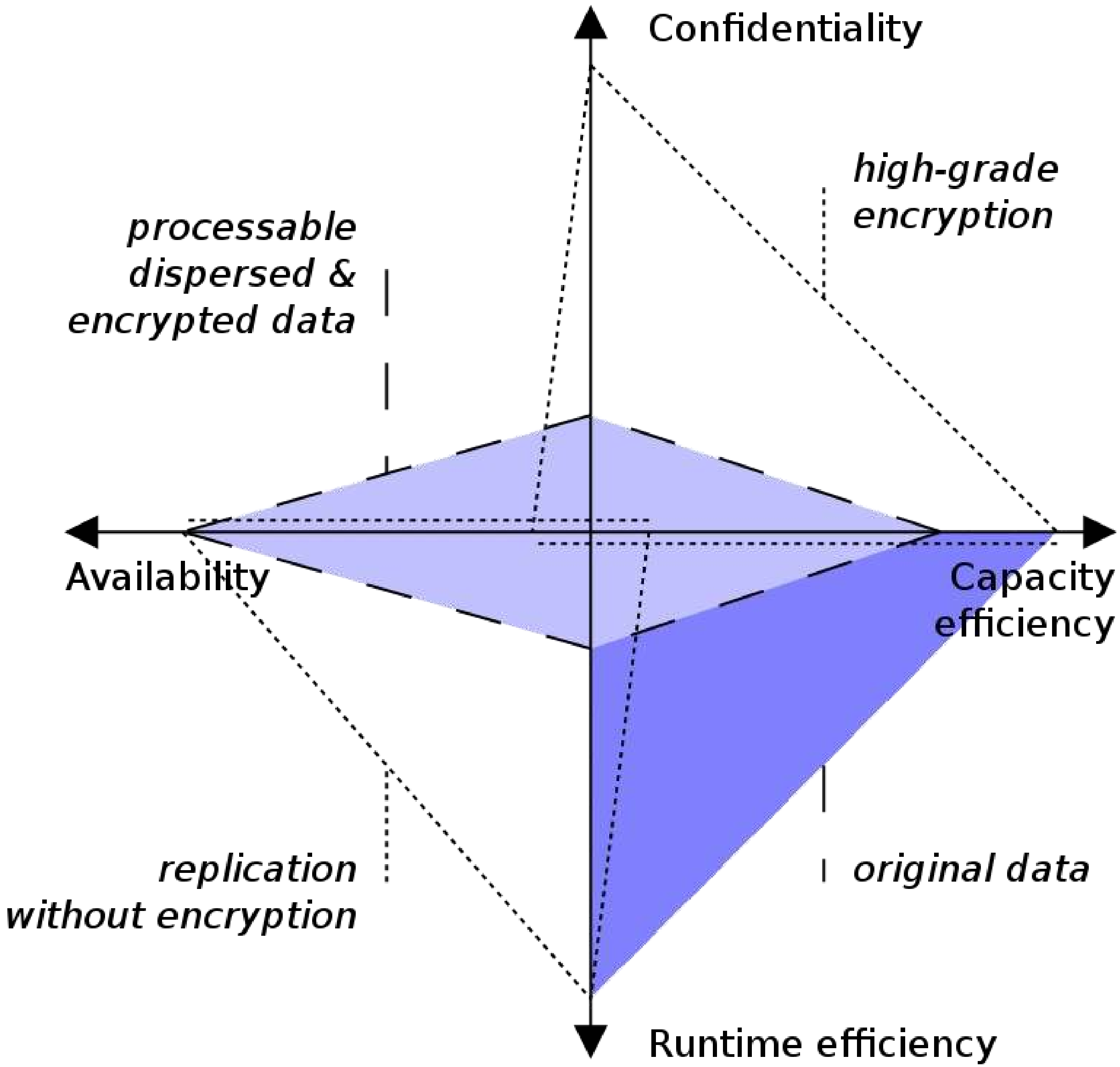


Combined („stealthy“) data coding

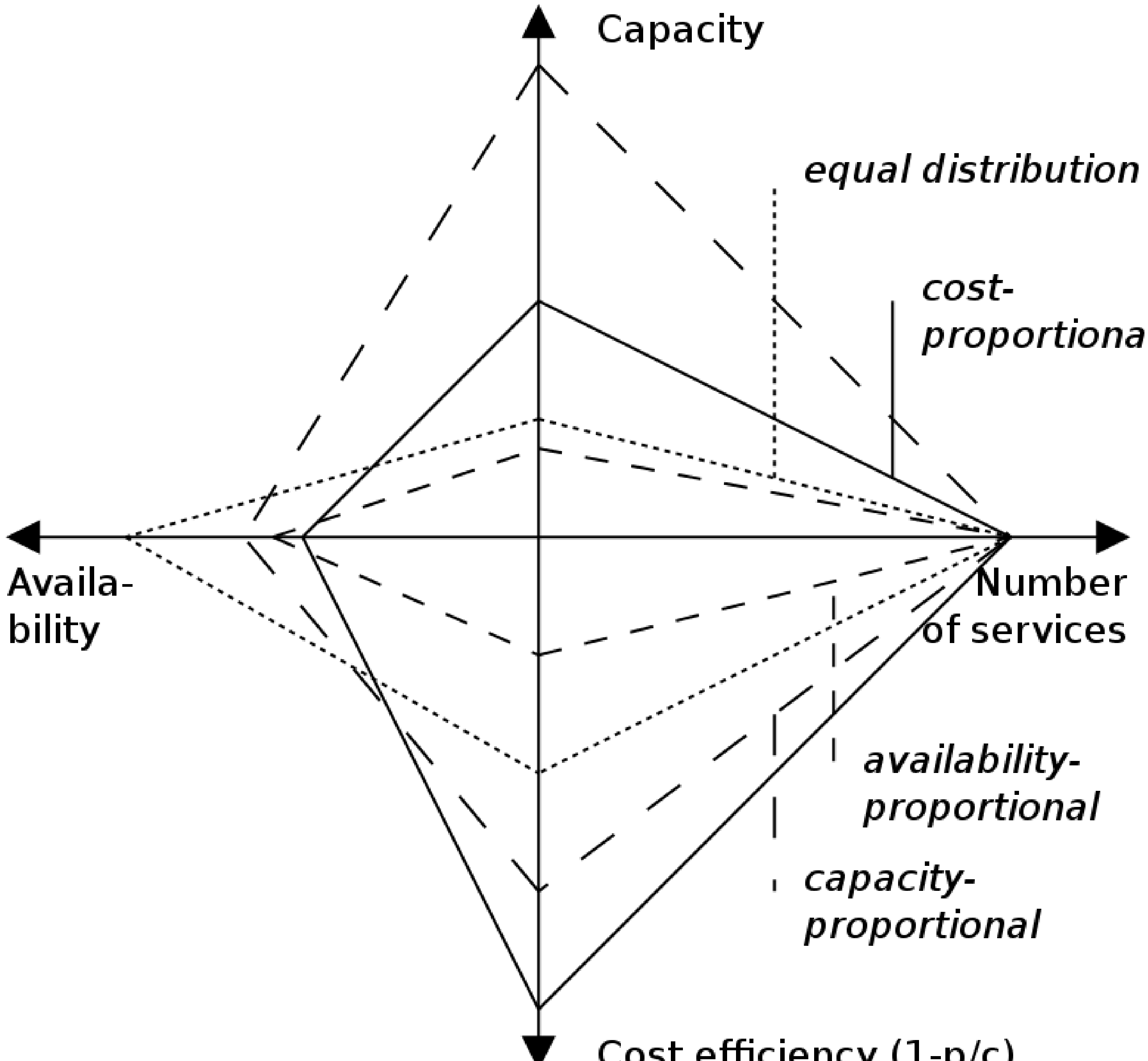


goal: availability
implied: privacy, confidentiality

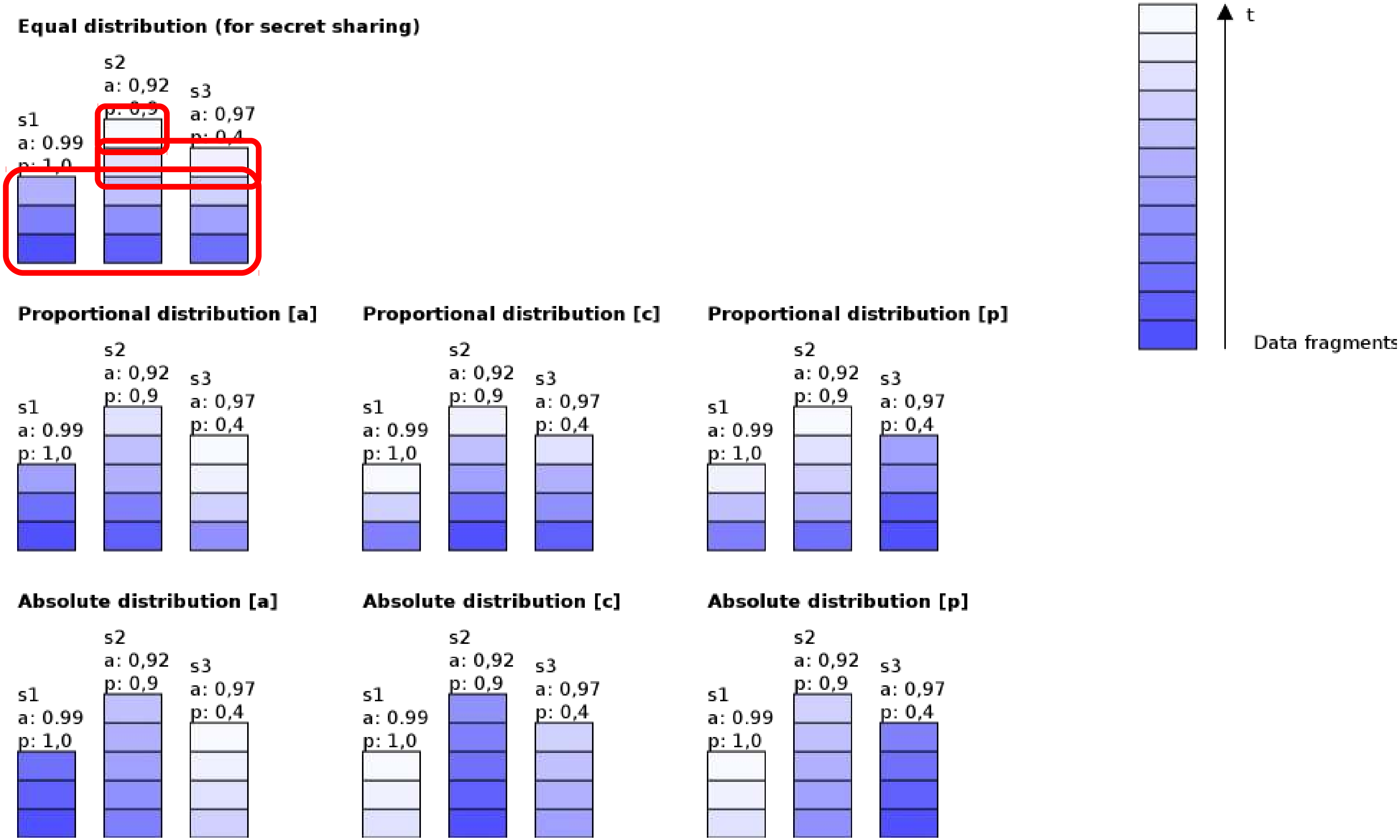
Data coding trade-offs



Data distribution trade-offs



Data distribution strategies/algorithms



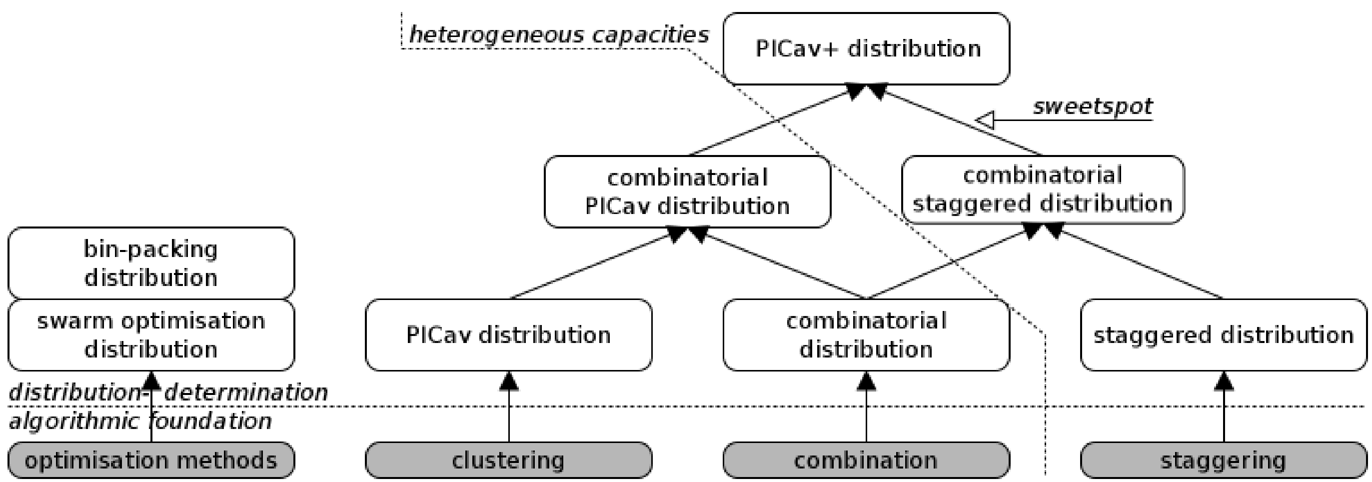
PICav+: The best-of-breed strategy

- powerful: optimises for capacity, price, availability constraints & runtime
- staggered: considers all elements in powerset of candidate service set
- sliced: capacity-maximising calculation rings
- iterative: finds some result first, finds best result eventually
- fast (clustering) & precise

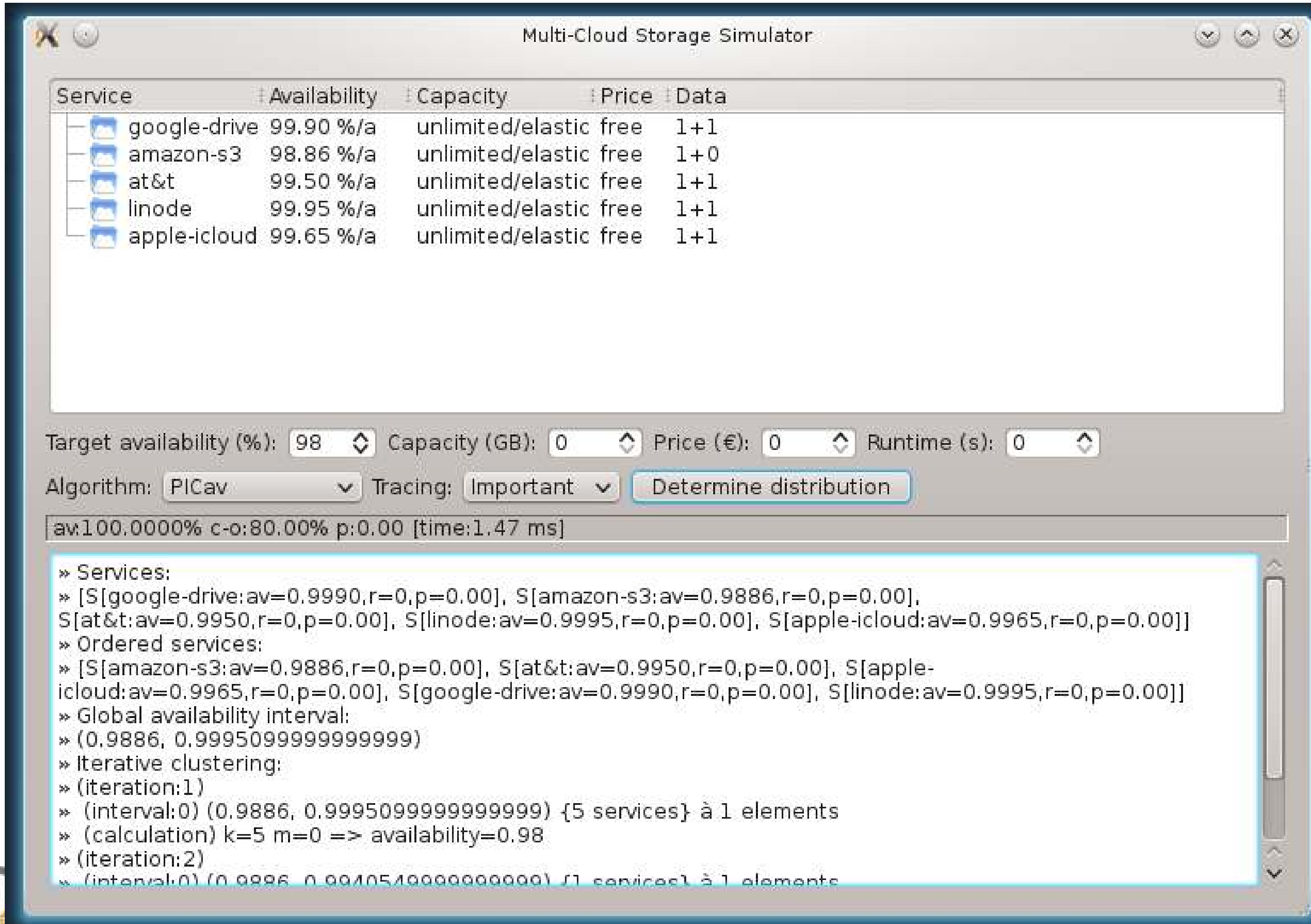
For each slice:

Homogeneous complexity: $$availability = \sum_{i=k}^{n} \binom{n}{i} a_1^i (n - a_1)^{n-i}$$

Heterogeneous complexity: $$availability = \sum_{S \in P_{>k}(C)} \left(\prod_{i \in S} a_i \cdot \prod_{i \in C \setminus S} 1 - a_i \right)$$



Data distribution tool: MCS-SIM/EMU



MCS-SIM results

```
Services: T1,T10,T2,T3,T4,T5,T6,T7,T8,T9
Distribution [algorithm:        fixed[sp] time:  41.49]: {1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0} error, no solution found; discarding availability=0.5653
Distribution [algorithm:        fixed[re] time:  39.86]: {1+0,0+1,0+1,0+1,0+1,0+1,0+1,0+1,0+1,0+1} availability=1.0000 price=10.00 capacity-overhead=0.90
Distribution [algorithm: proportional[av] time:  39.66]: {6+0,0+0,1+0,1+0,1+0,1+0,0+0,0+0,0+0,0+0} error, no solution found; discarding availability=0.8583
Distribution [algorithm: proportional[ca] time:  39.80]: {1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0,1+0} error, no solution found; discarding availability=0.5653
Distribution [algorithm: proportional[pr] time:  39.75]: {0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,10+0} error, no solution found; discarding availability=0.9100
Distribution [algorithm:     absolute[av] time:  39.72]: {10+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0} availability=0.9900 price=1.00 capacity-overhead=0.00
Distribution [algorithm:     absolute[ca] time:  39.94]: {10+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0} availability=0.9900 price=1.00 capacity-overhead=0.00
Distribution [algorithm:     absolute[pr] time:  39.79]: {10+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0} availability=0.9900 price=1.00 capacity-overhead=0.00
Distribution [algorithm:       random     time:  40.05]: {1+0,1+1,0+0,1+1,1+0,1+0,0+0,1+1,1+0,0+0} availability=0.9582 price=7.00 capacity-overhead=0.30
Distribution [algorithm:  combinatory     time: 300.17]: {1+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0,0+0} availability=0.9900 price=1.00 capacity-overhead=0.00
Distribution [algorithm:    staggered[pl] time: 386.44]: {1+x,1+x,1+x,1+x,1+x,1+x,1+x,1+x,1+x,1+x} error, solution found but runtime exceeded by 0.09s
Distribution [algorithm:    staggered[co] time: 301.59]: {1+x,1+x,1+x,1+x,0+x,0+x,0+x,0+x,0+x,0+x} availability=0.9931 price=4.00 capacity-overhead=0.75
Distribution [algorithm:        picav     time:  78.41]: {1+1,1+0,1+1,1+1,1+1,1+1,1+0,1+0,1+0,1+0} availability=0.9993 price=10.00 capacity-overhead=0.50
Distribution [algorithm:       picav+[av] time: 300.35]: {1+1,1+0,1+1,1+1,1+1,0+1,0+0,0+0,0+0,0+0} availability=0.9996 price=5.00 capacity-overhead=0.50
Distribution [algorithm:       picav+[ca] time: 301.73]: {1+1,1+0,1+1,1+1,1+1,0+1,0+0,0+0,0+0,0+0} availability=0.9996 price=5.00 capacity-overhead=0.50
Distribution [algorithm:       picav+[pr] time: 300.30]: {1+1,1+0,1+1,1+1,1+1,0+1,0+0,0+0,0+0,0+0} availability=0.9996 price=5.00 capacity-overhead=0.50
```

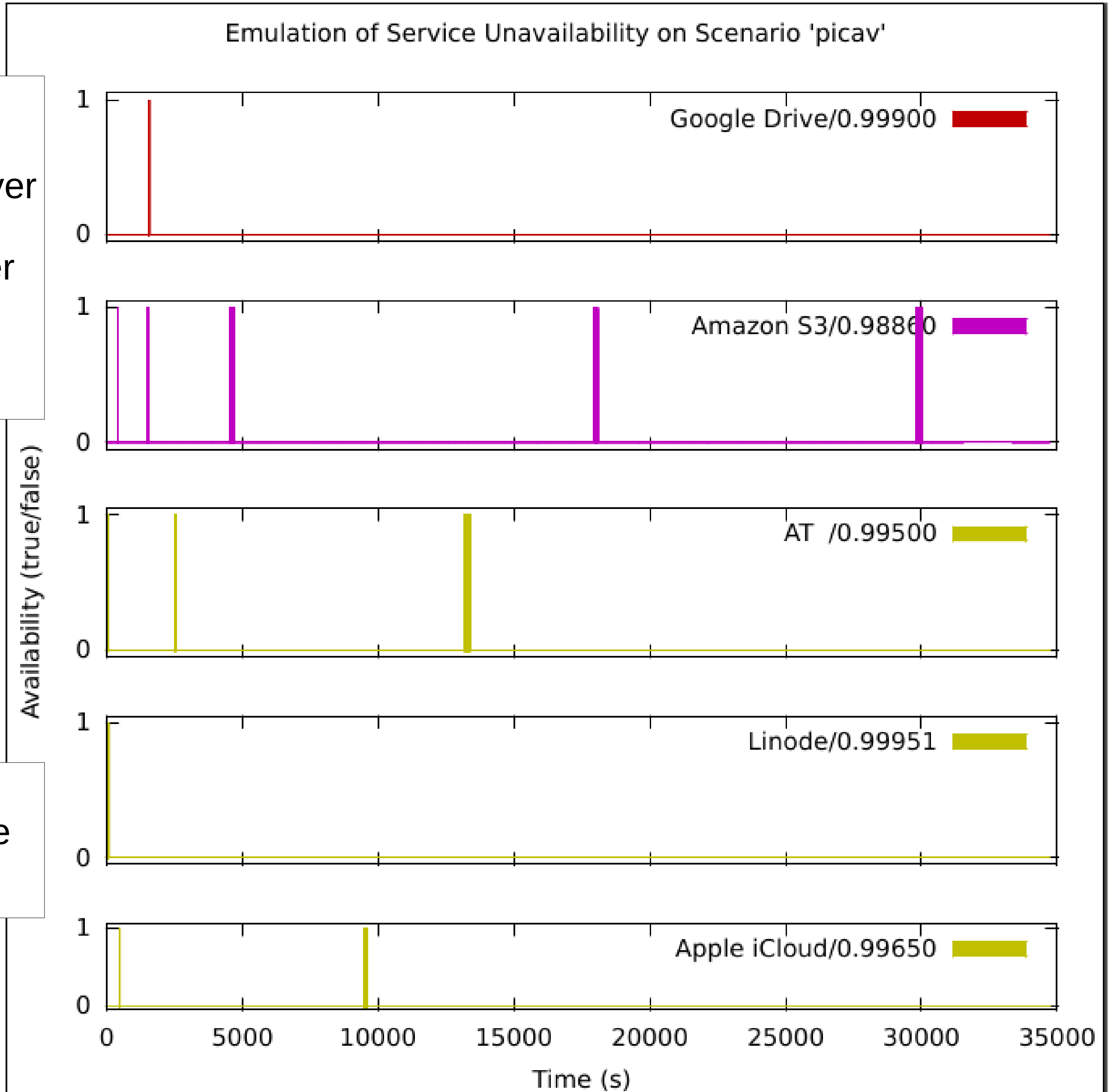
```
fixed[sp]     time: 41.49: {1+0,1+0,...,1+0} error, no solution available
fixed[re]     time: 39.86: {1+0,0+1,...,0+1} availability=1.0000 price=10.00 capov=..
staggered[pl] time:386.44: {1+x,1+x,...,1+x} error, solution found but rt exceeded

proportional[av/ca/pr]
absolute[av/ca/pr]
random
combinatory
staggered[co] *
picav
picav+[av/ca/pr] *
= 16 assignment algorithm variants, 4 combinatory x 3 staggered = 2 optim.
```

$$\mathcal{C}_n \subset \mathcal{C}_h \in \mathfrak{P}(\mathcal{C}) = \{\{T_1\}, \{T_1,T_2\}, \ldots, \underline{\{T_1, \ldots, T_n\}}\}$$

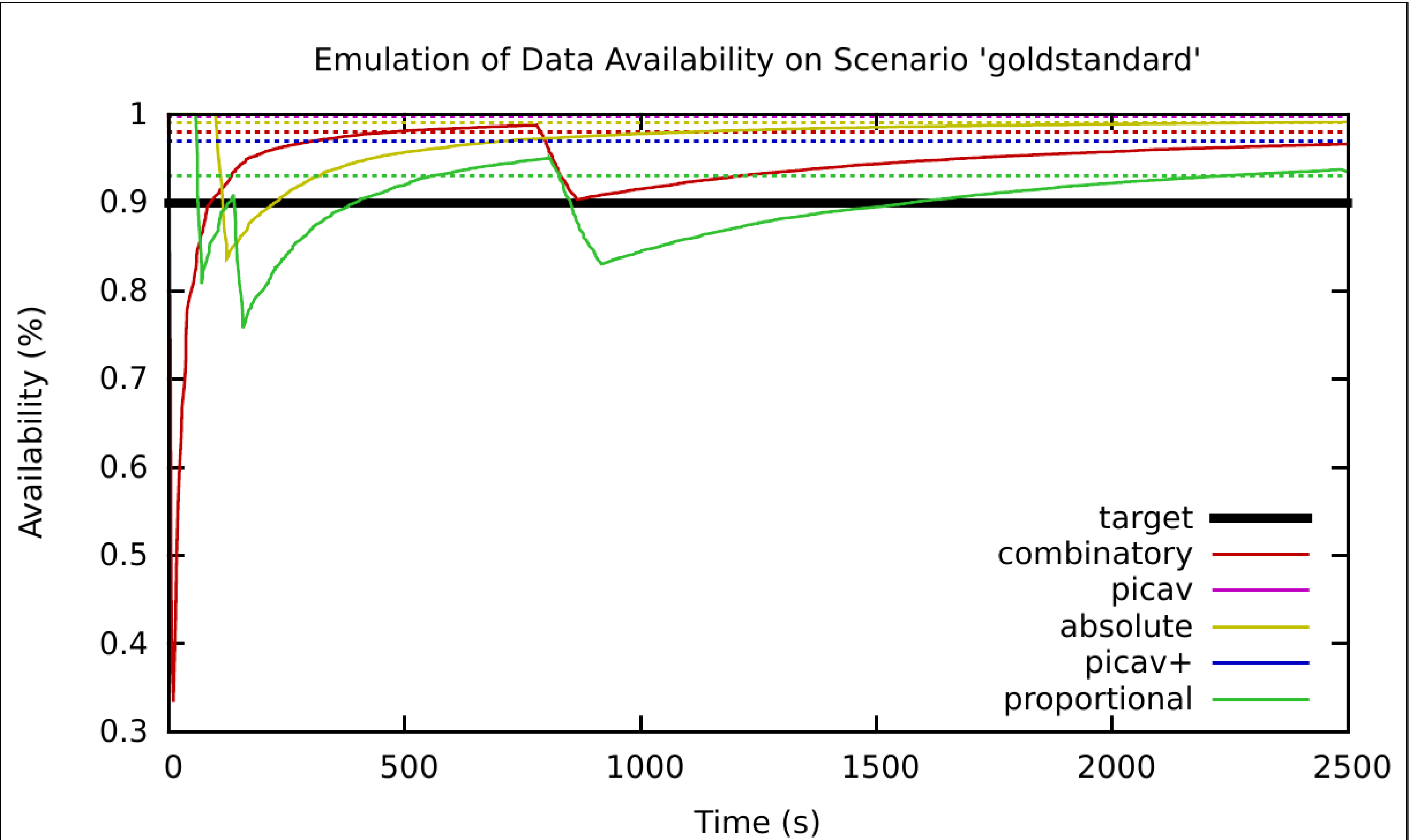
MCS-EMU results

4 targets:
- no-op (sim.)
- web/file server (storage)
- OS container (compute)
- L4 proxy (network)

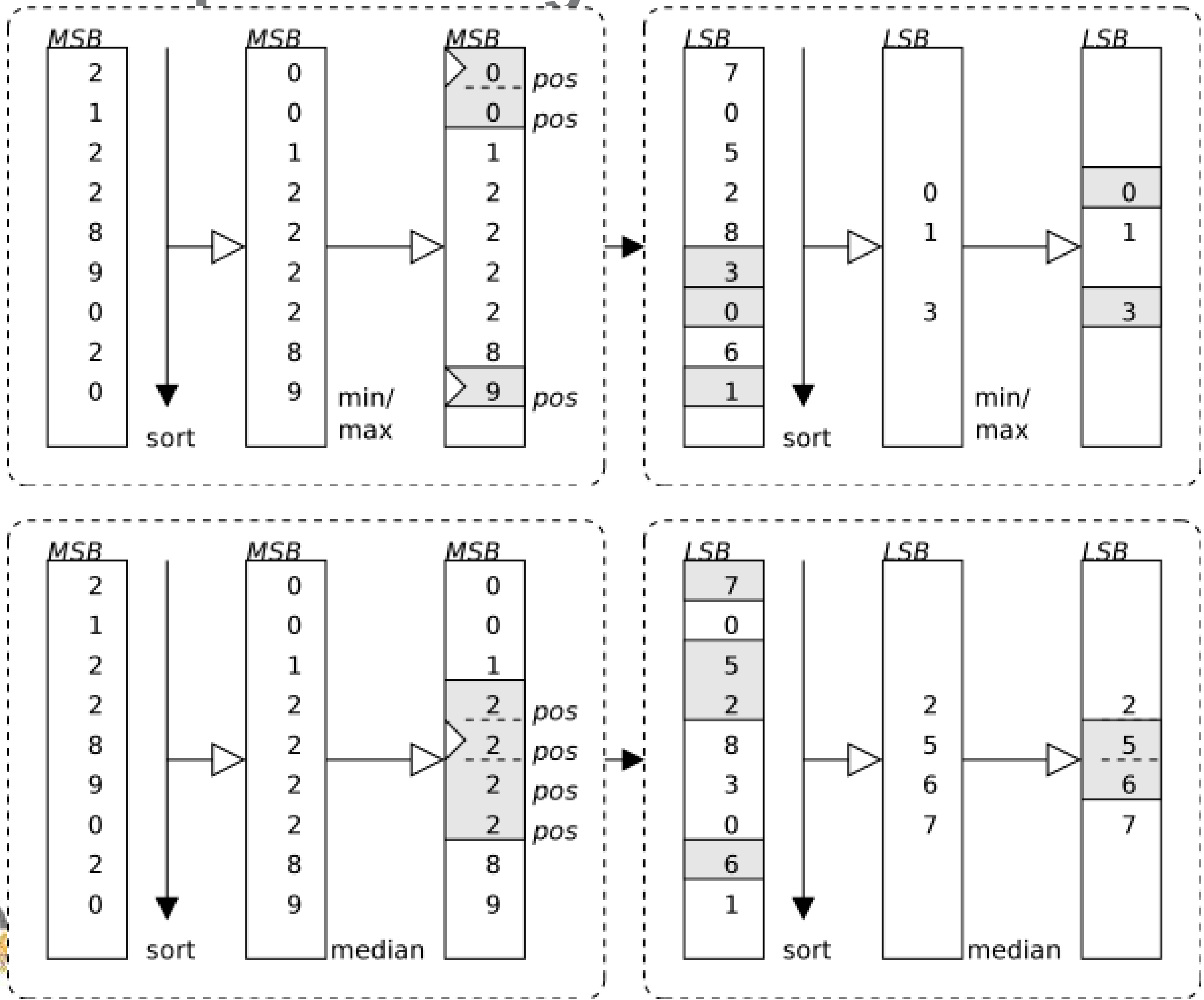


2 models:
- convergence
- incident

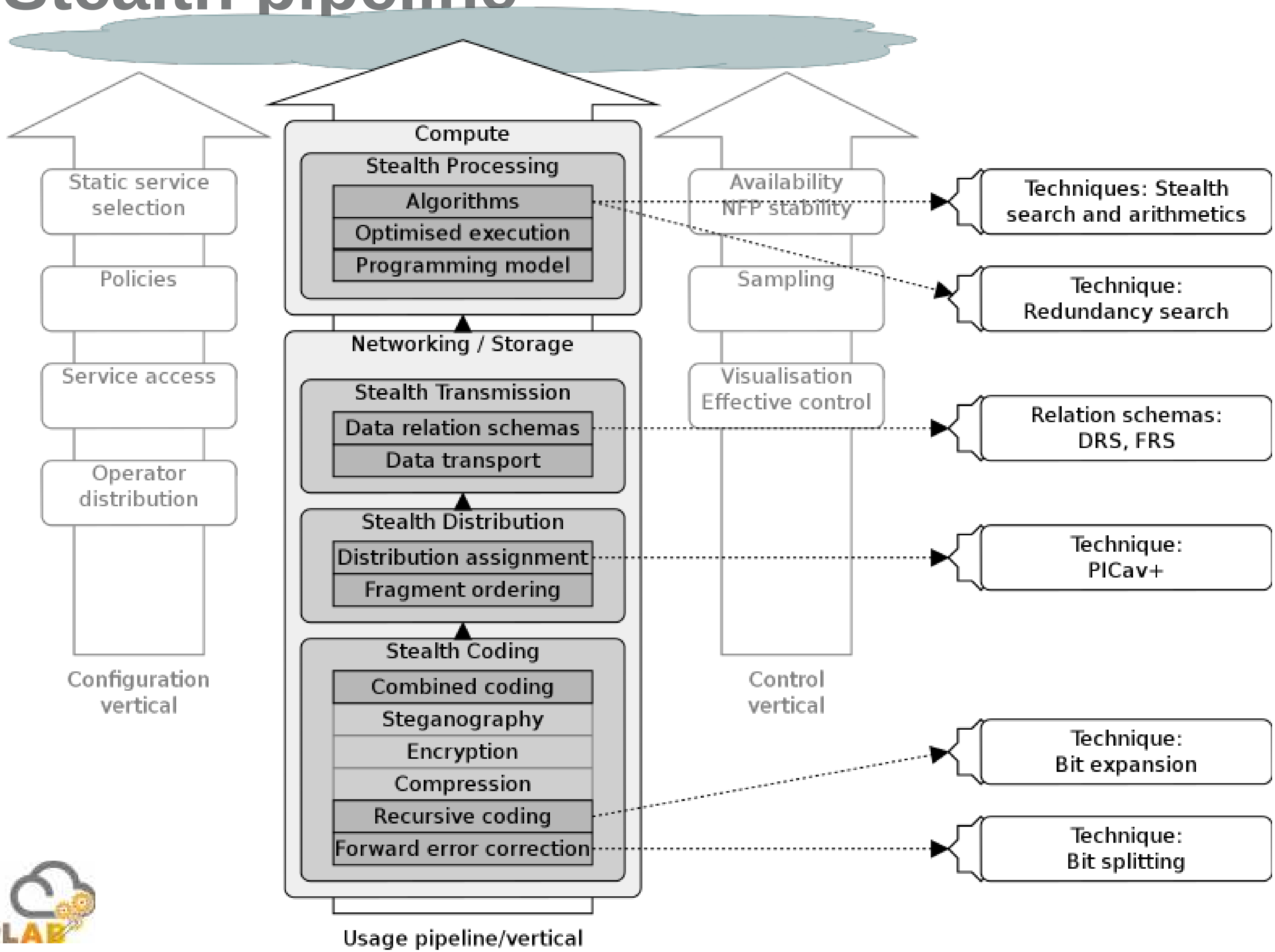
MCS-EMU results



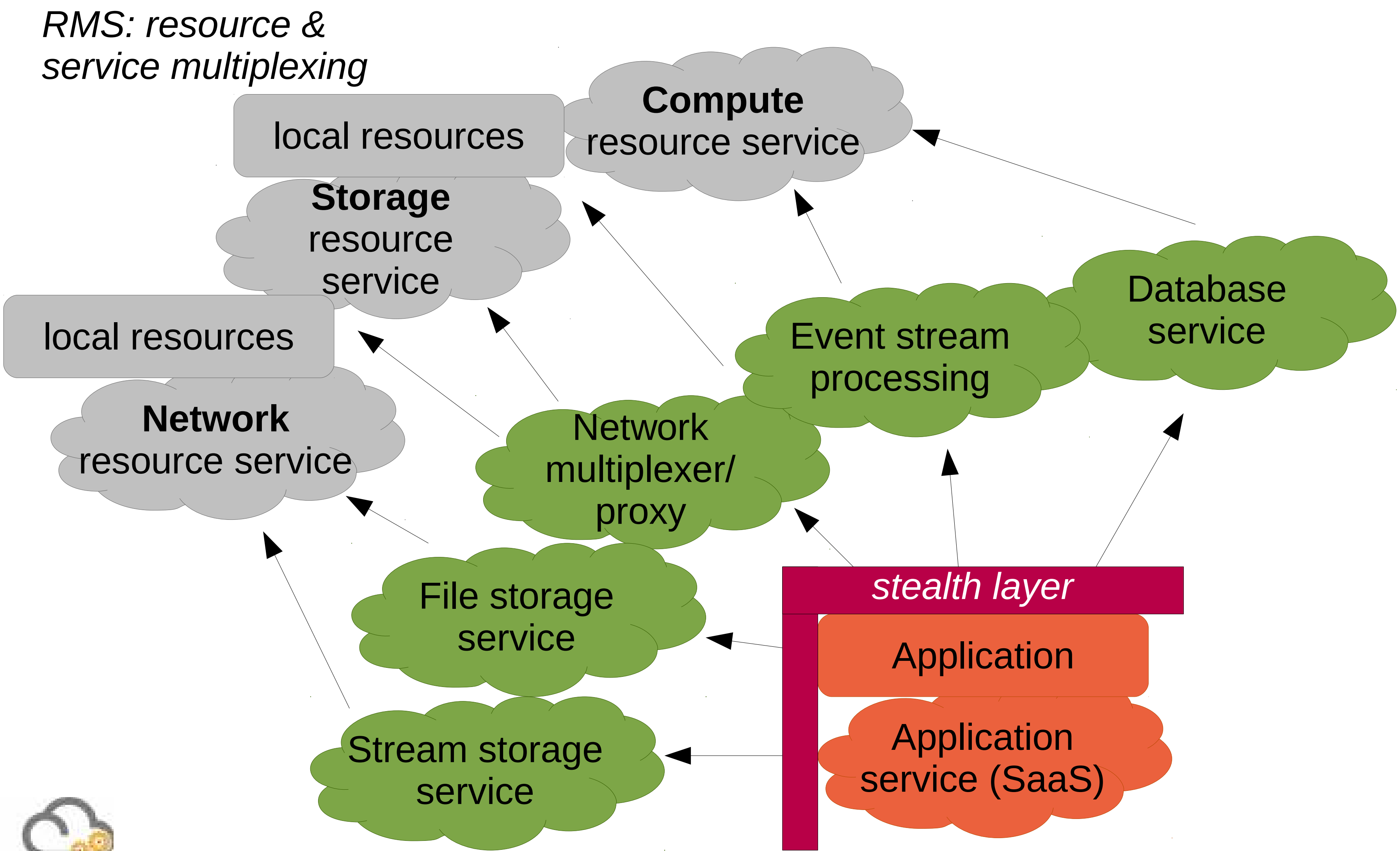
Stealth processing



Stealth pipeline

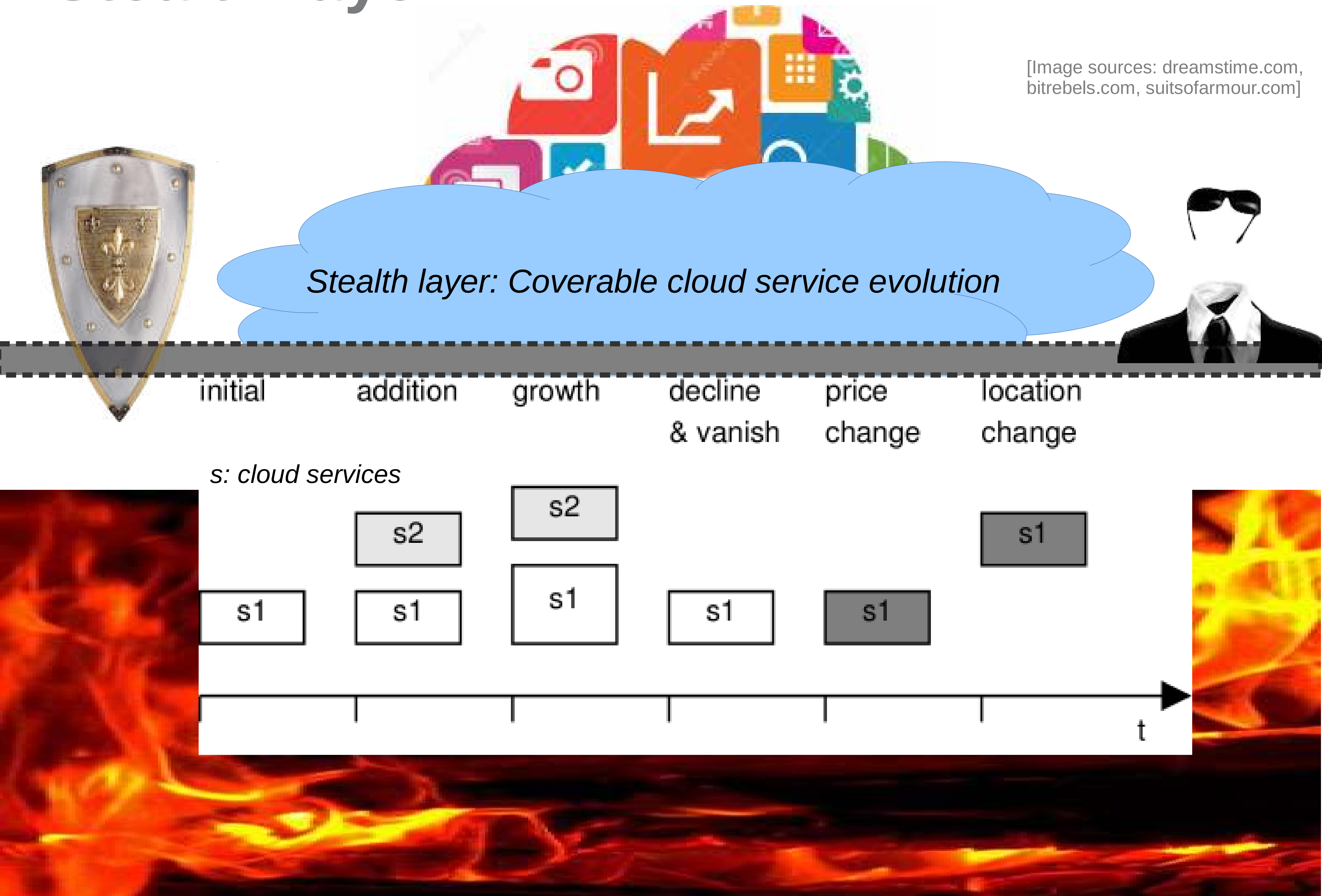


Stealth layer



Stealth layer

[Image sources: dreamstime.com, bitrebels.com, suitsofarmour.com]



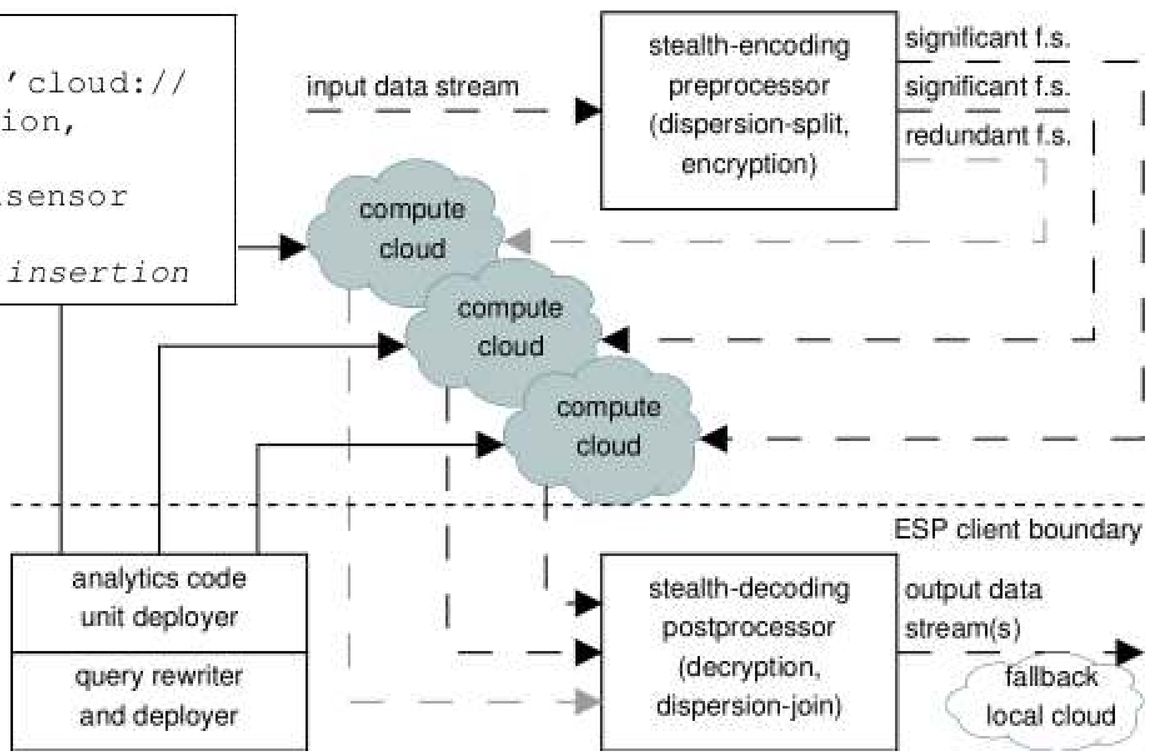
Showcase: StealthDB

Listing 1. Sender commands

```
\$ ./stealthdb sender
>>> USE CLOUDS 'cloud://cloud1' AND 'cloud://
    cloud2' WITH 'encryption,dispersion,
    ordered';
>>> CREATE TABLE heartsensor (frequency REAL);
>>> INSERT INTO heartsensor (frequency) VALUES
     (102.4);
... -- more insertions follow
```

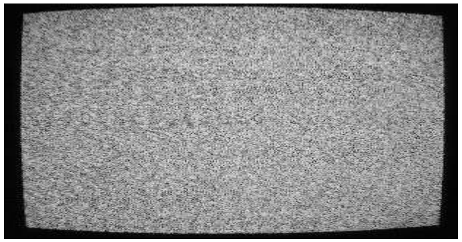
Listing 2. Receiver commands

```
\$ ./stealthdb receiver
>>> USE CLOUDS 'cloud://cloud1' AND 'cloud://
    cloud2' WITH 'encryption,dispersion,
    ordered';
>>> SELECT AVG(frequency) FROM heartsensor
    WHERE frequency > 0.0 FOREVER;
... -- query gets triggered on each insertion
```



Showcase: StealthDB

A data fragment in the cloud...

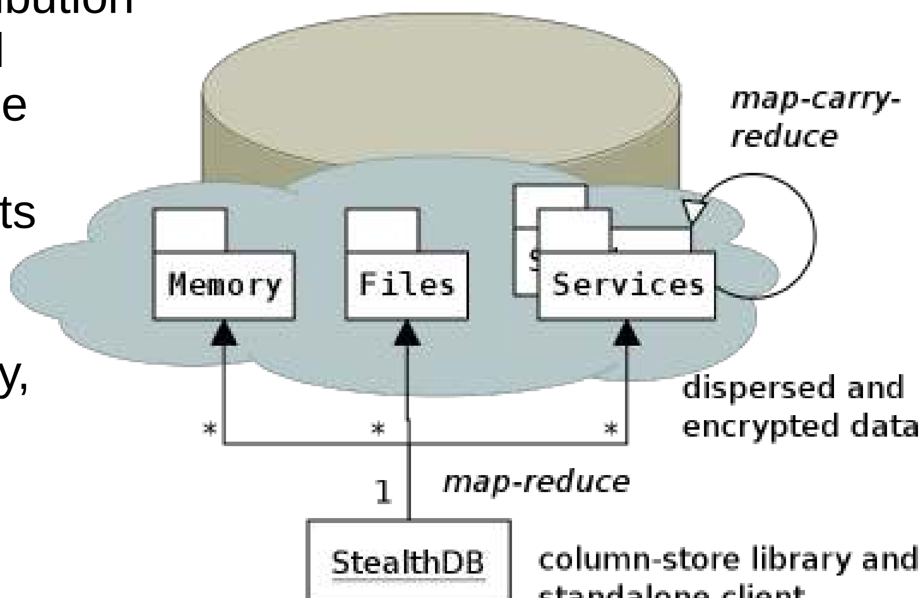


... what can we do with it?

Dispersed Processing:
- structure-preserving bitsplitting
  => search (any data)
  => arithmetics, statistics (structured data)

Encrypted Dispersed Processing (Stealth Processing):
- homomorphic encryption
- order-preserving encryption
- searchable encryption

Features:
- per-column distribution
- migration control
- map-carry-reduce operations
- user requirements optimisation (performance, energy-efficiency, reliability, ...)



StealthDB in Action: overview

```
josef@rumba:/repos/space-universe/dispersedalgorithms/db$ ./stealthdb
~~ StealthDB ›master ›Wed May 20 16:14:37 2015 +0200 ~~
Type HELP; to get started.
Using database 'stealthdb'.
Storing all data and performing all procedures on ['mem://localhost'] with ['replication'].
>>> HELP;
StealthDB Quickhelp
 HELP [<topic>]
 SHOW DATABASES|TABLES
 CREATE TABLE <table> [(<column> <column-type>, ...)]
 DESCRIBE <table>
 DROP TABLE [IF EXISTS] <table>
 CREATE DATABASE <database>
 USE DATABASE <database>
 DROP DATABASE <database>
 [EXPLAIN ANALYZE] SELECT [DISTINCT] */<column>/<aggregate>(*/<column>)/<predicate>, ... [FROM <table>]
  [WHERE <column> LIKE/=/... <value>] [ORDER BY <column> [ASC|DESC]] [OPTIMIZE FOR <goal>] [FOREVER]
 INSERT INTO <table> (<column>, ...) VALUES (<value>, ...)
 DELETE FROM <table>
 USE CLOUDS <cloud> [AND <cloud>...][WITH <distribution>]
 ALTER TABLE <table> [ALTER COLUMN <column>] USE CLOUDS ...
 MODE <mode>
>>>
```

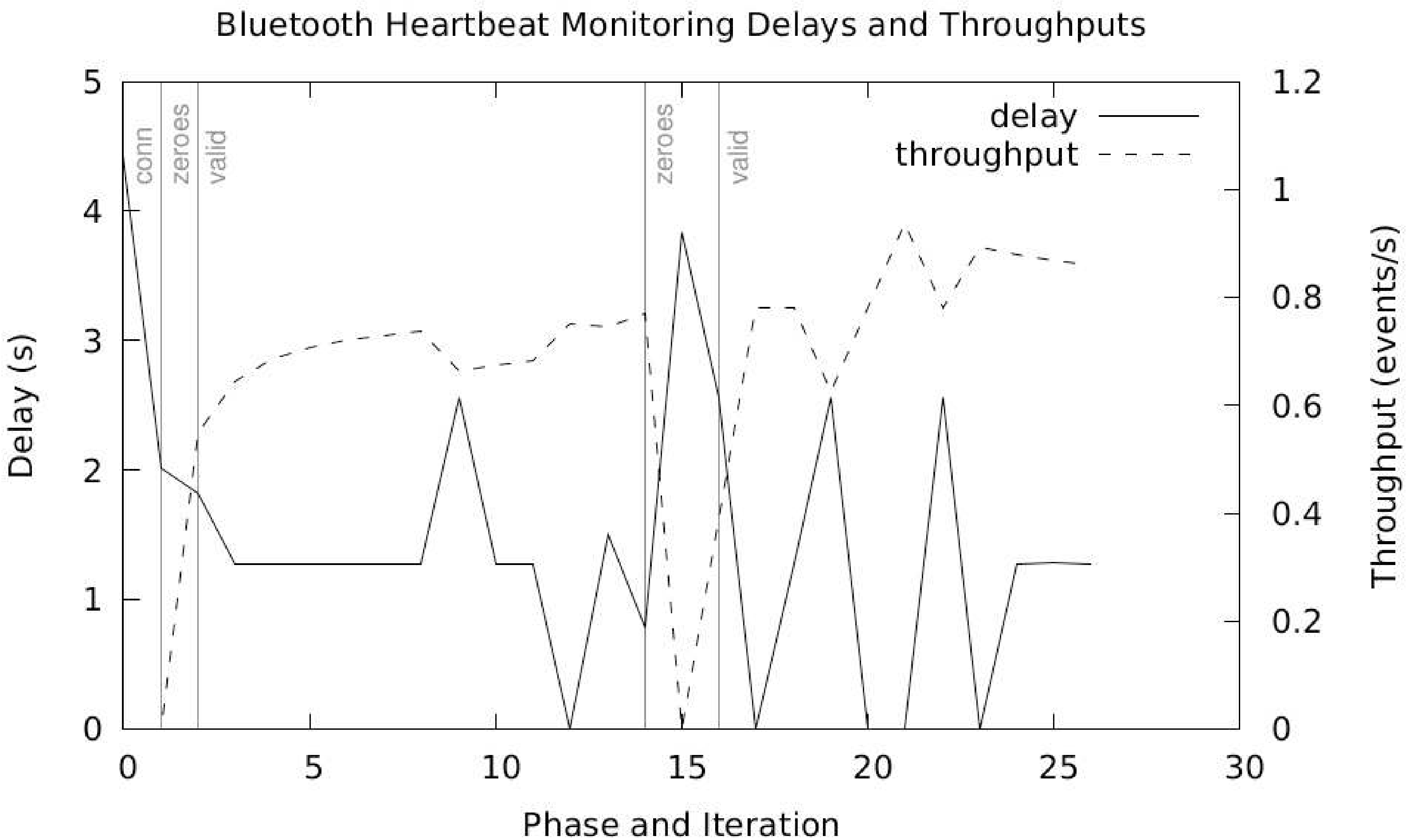
StealthDB in Action: multi-service

```
>>> USE CLOUDS 'mem://fastram' AND 'file:///globalfs/record' AND
... 'cloud://googleappengine-23' WITH 'hashring,encryption,ordered';
Using database 'stealthdb'.
Storing all data and performing all procedures on ['mem://fastram', 'file:///globalfs/record
', 'cloud://googleappengine-23'] with ['hashring', 'encryption', 'ordered'].
>>> CREATE TABLE cloudsalary (person TEXT, chf INT);
Created table cloudsalary.
(DEBUG:notifier:watch /globalfs/record/cloudsalary/person)
(DEBUG:info=::info::stealthdb-cloud::googleappengine-23::)
Added column person of type TEXT.
(DEBUG:notifier:watch /globalfs/record/cloudsalary/chf)
(DEBUG:info=::info::stealthdb-cloud::googleappengine-23::)
Added column chf of type INT.
>>>
```

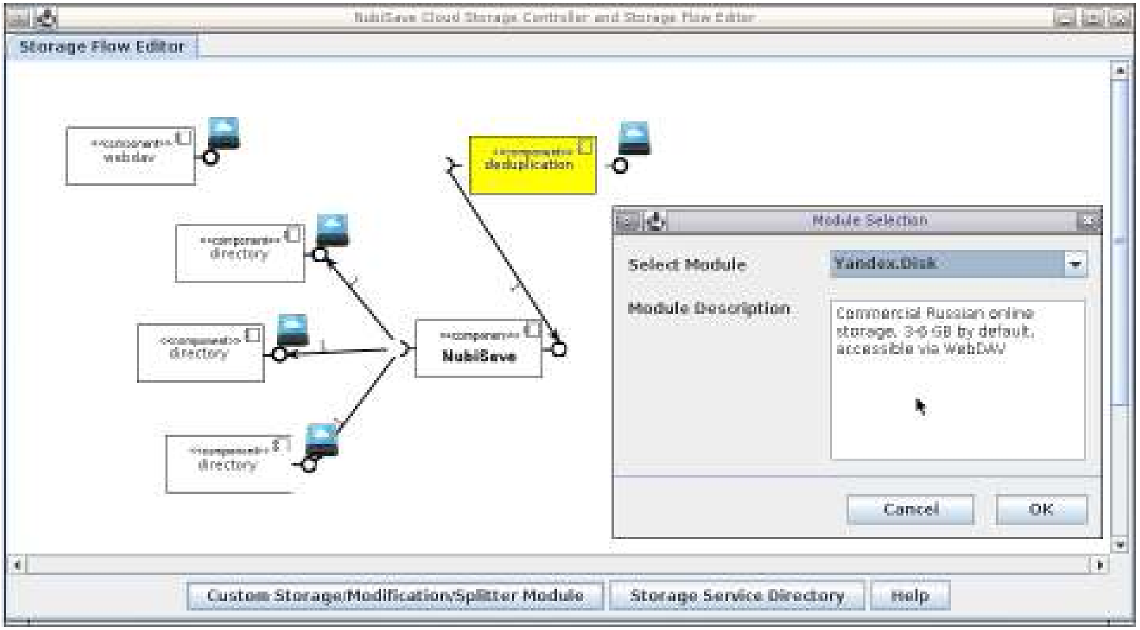
StealthDB in Action: protection goals

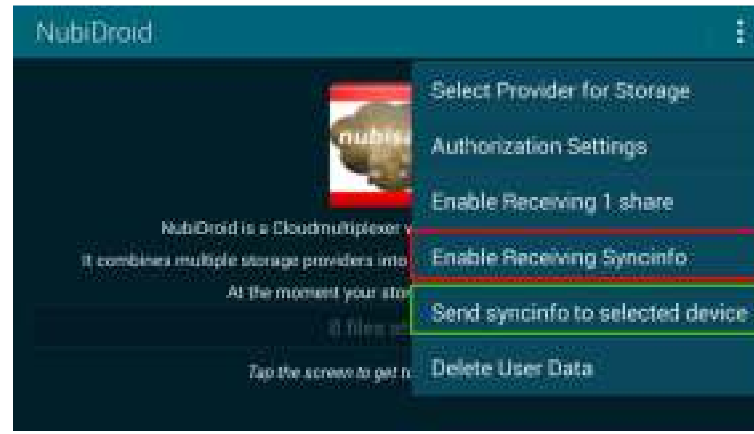
```
>>> SELECT MIN(chf) FROM cloudsalary OPTIMIZE FOR 'performance,precision';
(DEBUG:aggregate cloud:'mem://fastram')
(DEBUG:decryption:privkey=PrivateKey(l=276206633765143932185133659008305273600,m=77799637863
792083604813961003219226636),pubkey=PublicKey(n=276206633765143932218448800529509760119,n_sq
=76290104535872348142321164364999188369598776886438439468947400461280922894161,g=27620663376
5143932218448800529509760120),entry=74457922202228058251911302434069182833979044730019820823
958984398650514208122)
(DEBUG:encryption:decrypt-safe)
(DEBUG:decryption:result=250000)
(DEBUG:decryption:privkey=PrivateKey(l=276206633765143932185133659008305273600,m=77799637863
792083604813961003219226636),pubkey=PublicKey(n=276206633765143932218448800529509760119,n_sq
=76290104535872348142321164364999188369598776886438439468947400461280922894161,g=27620663376
5143932218448800529509760120),entry=29962522632047879125545174731291639553194534076196749551
474505932153721290035)
(DEBUG:encryption:decrypt-safe)
(DEBUG:decryption:result=180000)
(DEBUG:decryption:privkey=PrivateKey(l=276206633765143932185133659008305273600,m=77799637863
792083604813961003219226636),pubkey=PublicKey(n=276206633765143932218448800529509760119,n_sq
=76290104535872348142321164364999188369598776886438439468947400461280922894161,g=27620663376
5143932218448800529509760120),entry=40819492330637723794130390345172761293210144530235730019
692667468160790251811)
(DEBUG:encryption:decrypt-safe)
(DEBUG:decryption:result=24999)
(DEBUG:aggregate sampling:1.0000%)
Minimum of chf: 24999
>>>
```

Performance (depends on protection!)

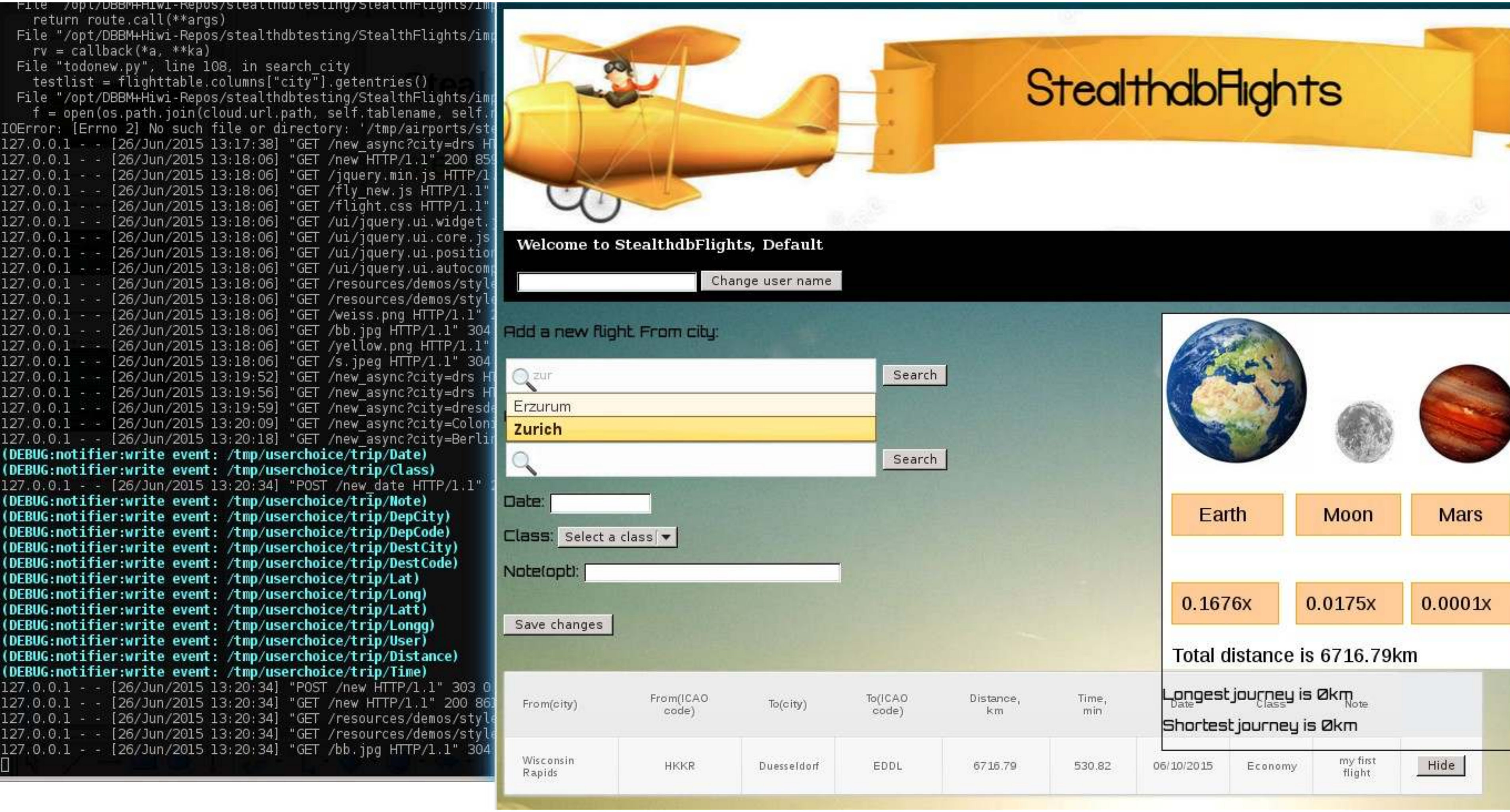


File Service Multiplexing





DB Service Multiplexing

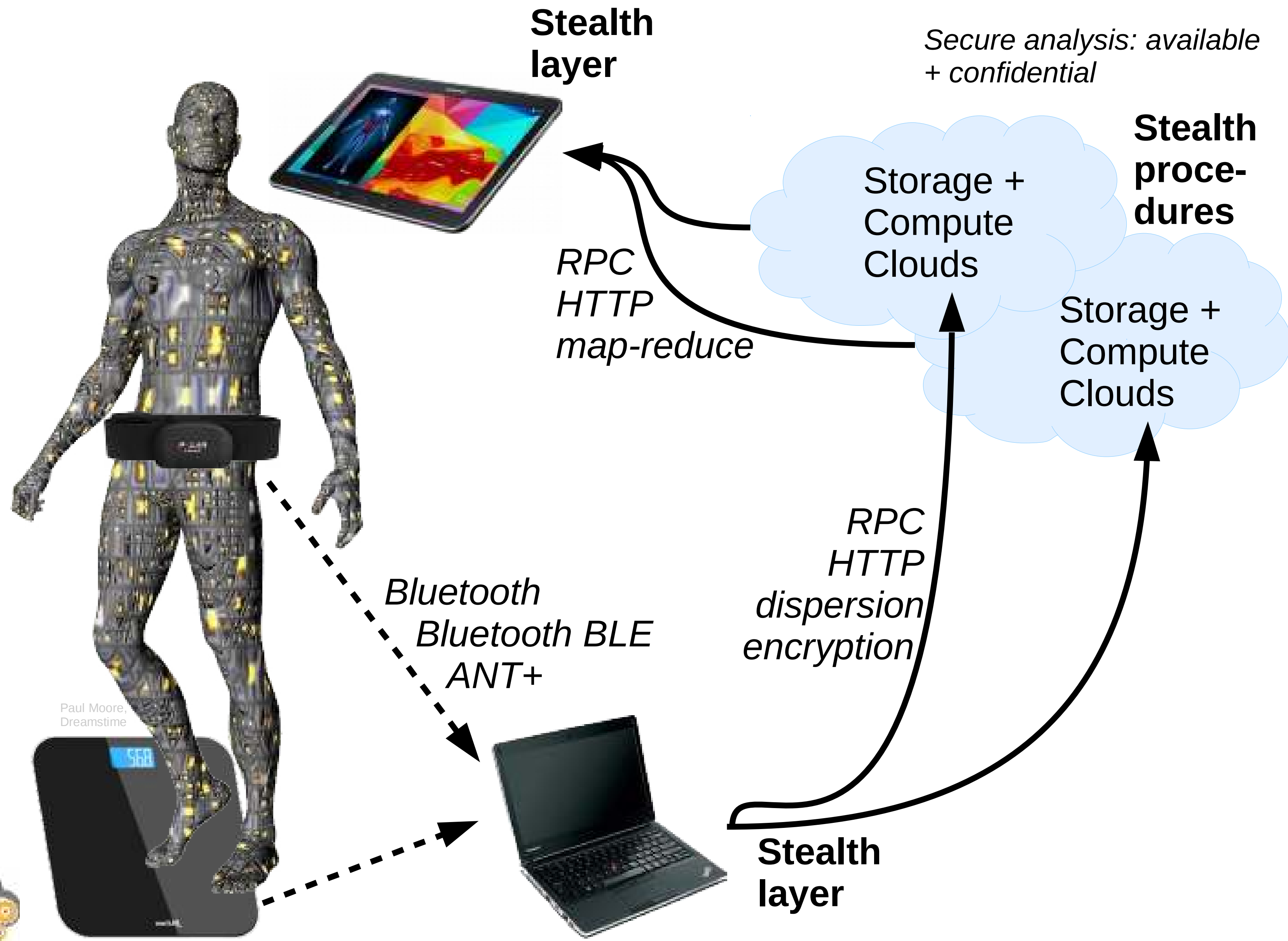


ESP Service Multiplexing

Writer:
USE CLOUDS 'cloud://cloud1' AND 'cloud://cloud2' WITH 'encryption,dispersion,ordered';
DROP TABLE IF EXISTS heartsensor;
CREATE TABLE heartsensor (frequency REAL);
↨ INSERT INTO heartsensor (frequency) VALUES (97.4);

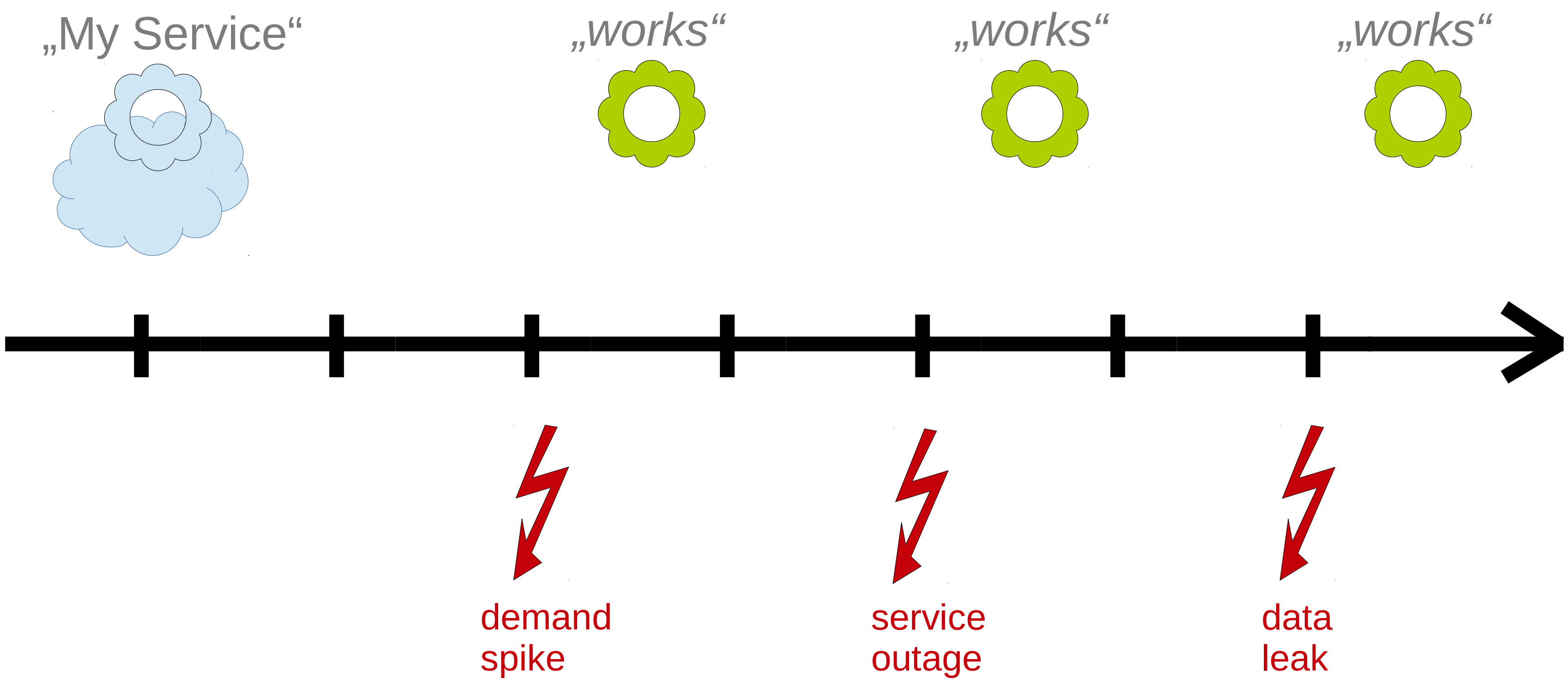
Reader:
USE CLOUDS 'cloud://cloud1' AND 'cloud://cloud2' WITH 'encryption,dispersion,ordered';
MODE debug;
SELECT AVG(frequency) FROM heartsensor WHERE frequency > 0.0 FOREVER;

ESP Service Multiplexing



Cloud-native & stealthy applications

Outlook: Tomorrow's cloud applications and services



Wrap-Up

New application/service models emerging for clouds:
- Cloud-Native Applications → resilience, scalability
- Stealth Applications → availability, confidentiality

SPLab to advance and transfer research + tooling in these areas!

UCC 2014/CLASP: workshop paper on stealth algorithms
NetSys 2015: demo paper on stealth queries
BlackSeaCom 2015: extended demo paper on stealth stream processing

UCC 2015: full paper on StealthDB √
UCC 2015/CloudAM: workshop paper on CNA evaluation
~autumn 2015: habilitation manuscript on risk minimisation in the cloud

References

[BBT+15] Sandro Brunner, Martin Blöchlinger, Giovanni Toffetti, Josef Spillner, Thomas M. Bohnert:
Experimental Evaluation of the Cloud-Native Application Design.
4th International Workshop on Clouds and (eScience) Applications Management (CloudAM), Limassol, Cyprus, December 2015. Submitted for review.

[SBS+15] Josef Spillner, Martin Beck, Alexander Schill, Thomas M. Bohnert:
Stealth Databases: Ensuring User-Controlled Queries in Untrusted Cloud Environments.
8th IEEE/ACM International Conference on Utility and Cloud Computing (UCC), Limassol, Cyprus, December 2015.

[Spi15] Josef Spillner:
Secure Distributed Data Stream Analytics in Stealth Applications. (Demo)
3rd International Black Sea Conference on Communications and Networking (BlackSeaCom), Constanţa, Romania, May 2015.

[SMS15] Josef Spillner, Lorenzo Miori, Julian Sanin:
Stealth Apps for Secure Personal Data Analytics in the Cloud. (Demo)
2nd International Conference on Networked Systems (NetSys), Cottbus, Germany, March 2015.

[SS14b] Josef Spillner, Alexander Schill:
Algorithms for Dispersed Processing.
1st International Workshop on Advances in Cloud Computing Legislation, Accountability, Security and Privacy (CLASP), London, UK, December 2014.

[SS14a] Josef Spillner, Alexander Schill:
Towards Dispersed Cloud Computing.
2nd IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Chișinău, Moldova, May 2014.