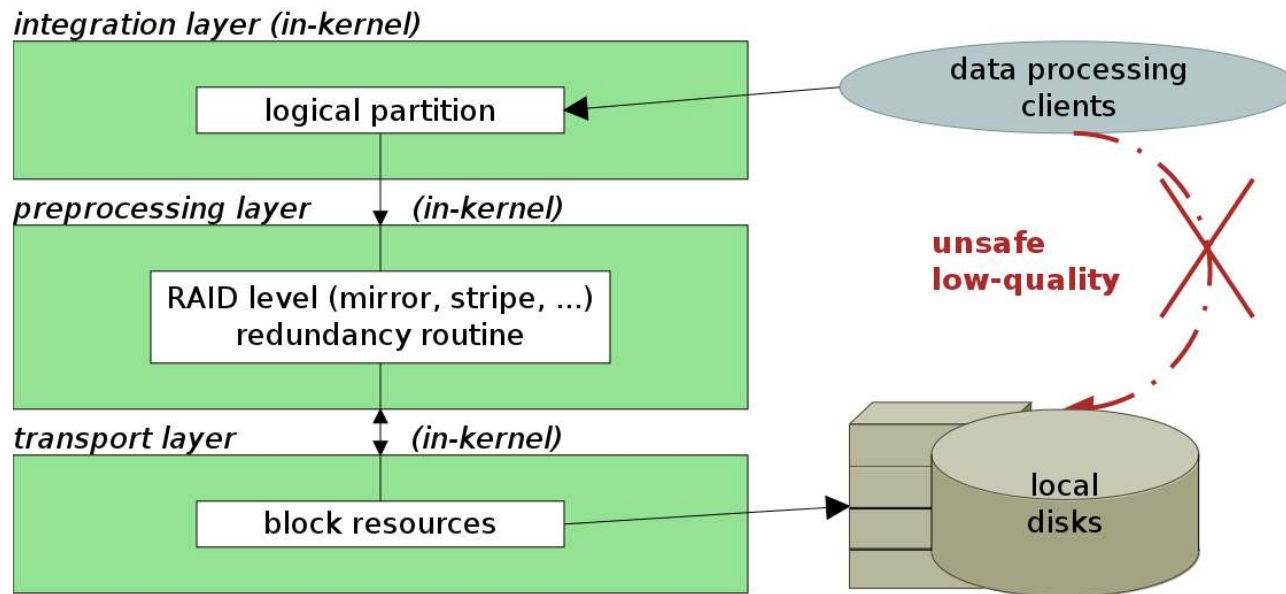


# Information Dispersion over Redundant Arrays of Optimal Cloud Storage for Desktop Users

**Josef Spillner, Gerd Bombach, Steffen Matthischke,  
Johannes Müller, Rico Tzschichholz, Alexander Schill**

**<mailto:josef.spillner@tu-dresden.de>  
<xmpp:josef.spillner@jabber.org>**

4th IEEE/ACM International Conference on Utility and Cloud Computing  
December 5, 2011, Melbourne, Australia



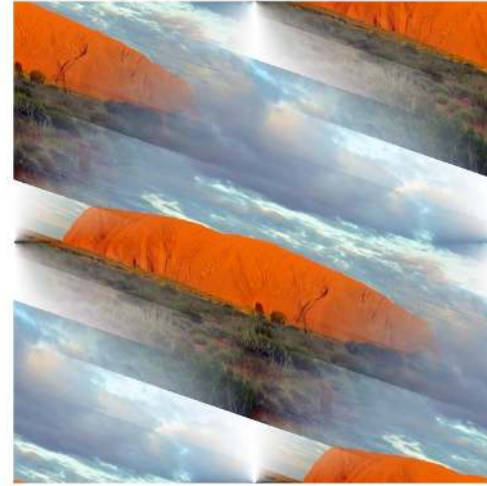


Note: black surface proportional to free offers capacity; red surface typical capacity

(1) original file



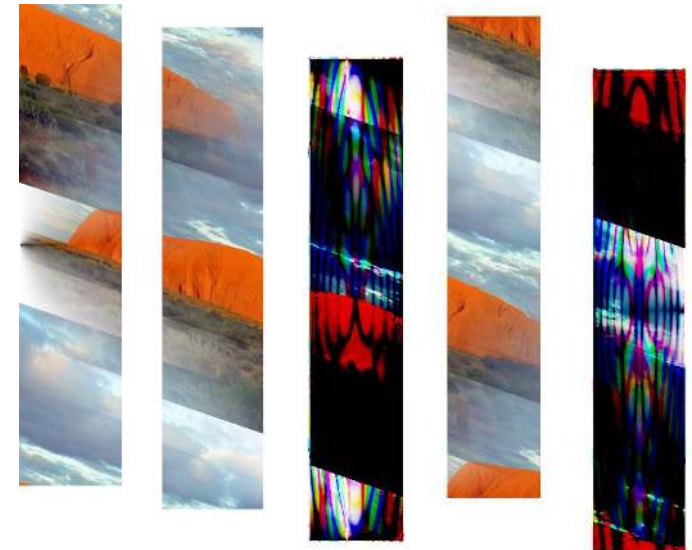
(2) added  
redundancy



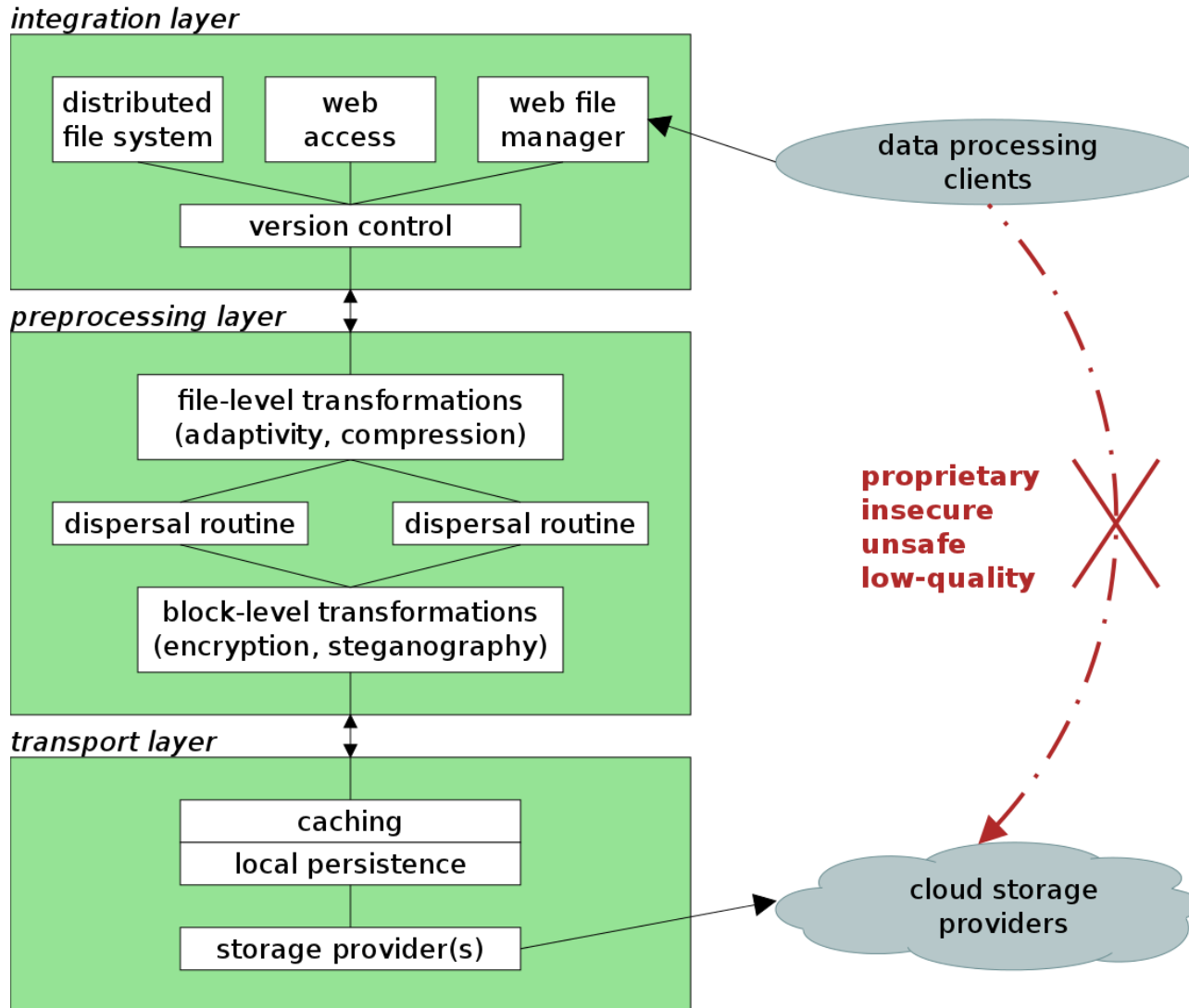
(3) fragments (k significant, n total)







(4) selective fragment encryption

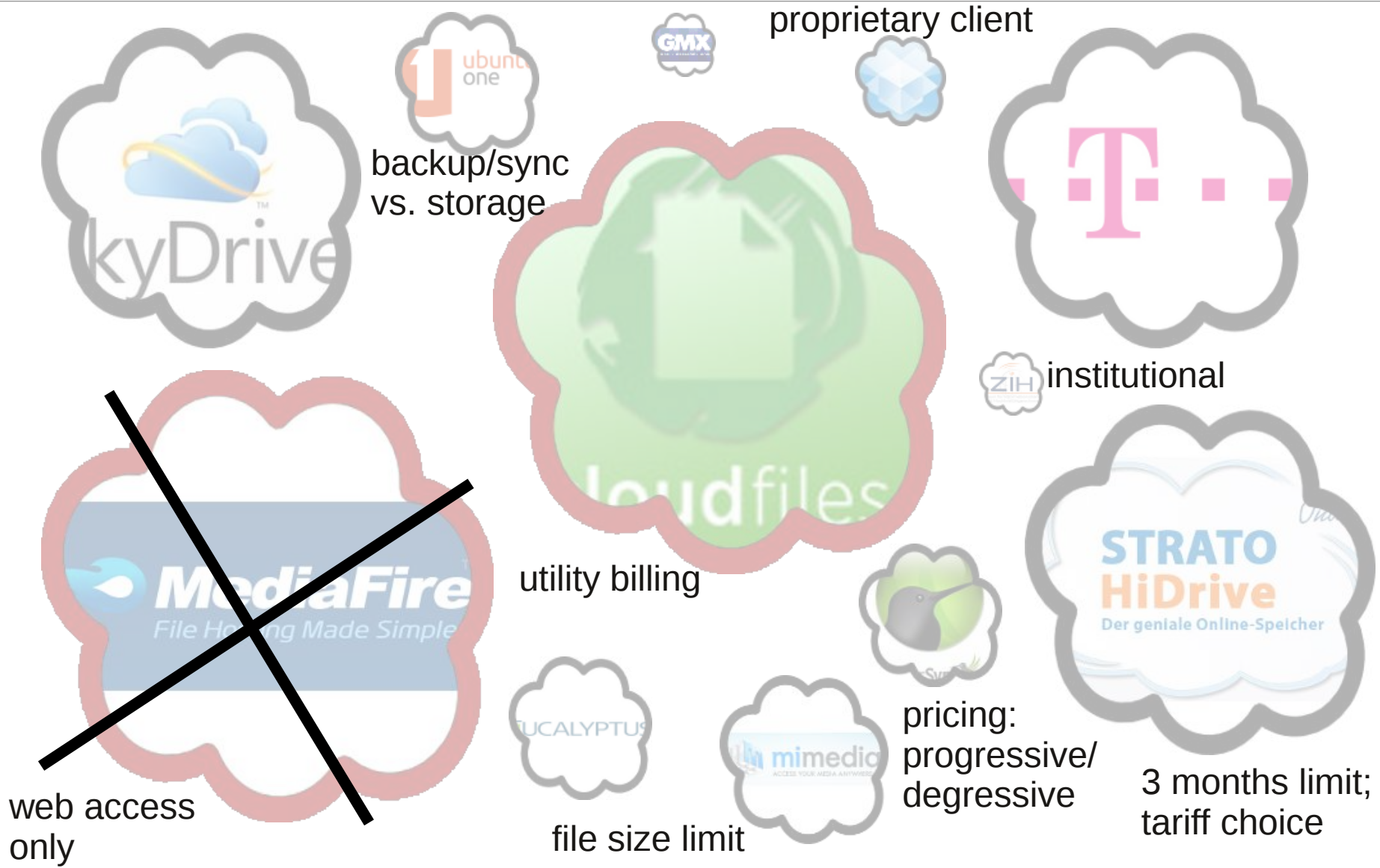


# RAIC - Redundant Arrays of Independent Clouds



Prototype	Origin	Authors	Approach	Integration	Availability
Dependable Sky ('11)	T-Clouds, Univ. of Lisbon 	Bessani, Correia, Quaresma, André, Sousa	data units, byzantine fail, Rabin erasure	application library	no
('11)	HPI, Potsdam 	Maxim Schnjakin, Christoph Meinel	NFPs, propr. platform, Liberation erasure	web interface	no
TrustedSafe ('10)	eGovCD, Berlin/ FhG FOKUS 	(no publication)	n fragments: one on USB, others remote	Windows plugin	no
Cloud Shredder ('11)	Chinese Academy of Sciences	Nan Zang, Jiwu Jing, Peng Liu	Two fragments: one local, one remote	(not yet known)	no
SecCSIE ('11)	T-Systems MMS, Dresden 	Ronny Seiger, Stephan Groß	Cachy Reed-Solomon erasure	CIFS proxy	no





## 1. base ontology: QoS

```
concept MeasurementUnit
  conversionFactor impliesType _double

concept SpaceUnit subConceptOf MeasurementUnit

instance GB memberOf SpaceUnit
  conversionFactor hasValue 1024.0
```

```
wsmIvariant _"http://www.wsmo.org/wsml/wsml-syntax/wsml-flight"
```

```
namespace { _"urn:ontology:conqo/CloudQoS.wsml#",
  qos _"urn:ontology:conqo/QoSBase.wsml#" }
```

```
ontology CloudQoS
  importsOntology { _"urn:ontology:conqo/QoSBase.wsml#" }
```

```
nonFunctionalProperties
  qosdefinition hasValue "Cloud Storage"
endNonFunctionalPropertie
```

```
concept PricePerMonth subConceptOf {qos#Quality, qos#LowerBetter}
  qos#unit impliesType qos#CurrencyUnit
```

```
concept Capacity subConceptOf {qos#Quality, qos#HigherBetter}
  qos#unit impliesType qos#SpaceUnit
```

```
concept CloudStorage
```

## 2. domain ontology: Cloud Storage

## 3. instance ontology: Google Storage

```
webService GoogleStorage
  importsOntology {
    _"urn:ontology:conqo/CloudQoS.wsml#" }

  capability ServiceCapability
    postcondition definedBy ?serviceType memberOf
      cloud#CloudStorage .

  instance PricePerData memberOf { cloud#PricePerData,
    qos#ServiceSpec}

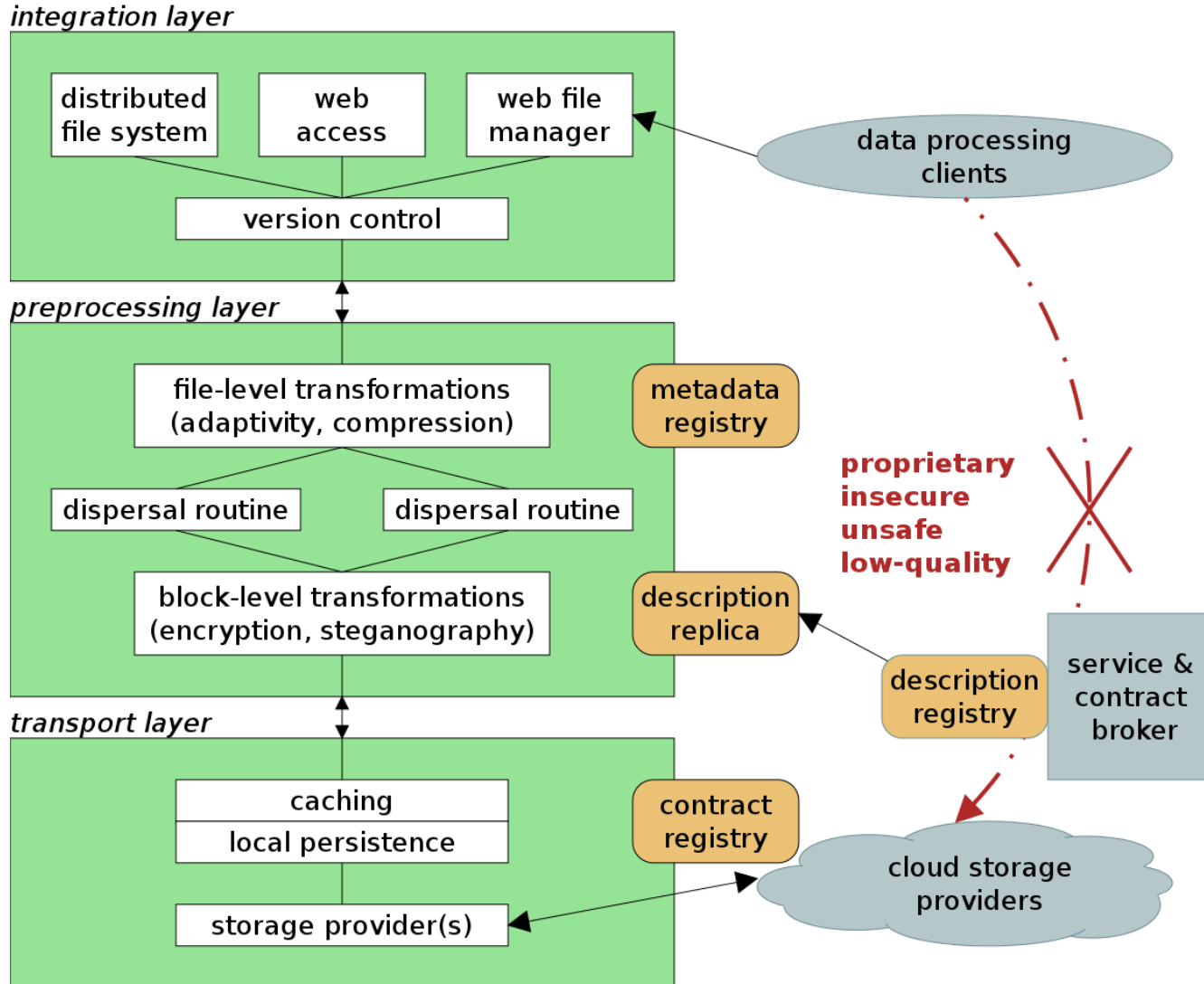
    qos#value hasValue 0.17
    qos#unit hasValue qos#Euro
```

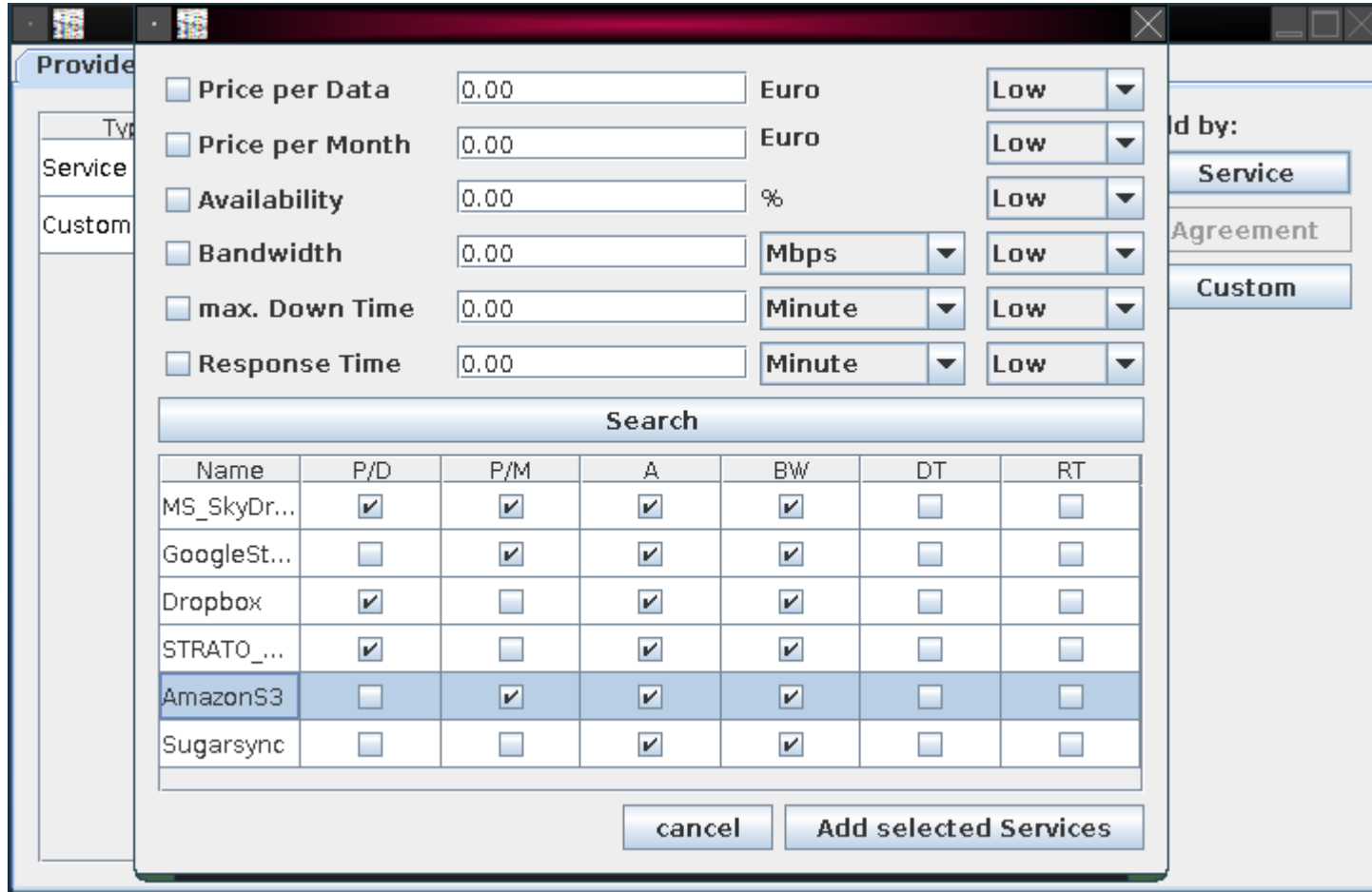
## 4. public catalogues

WSMO  
for IoS

WSMO  
for Cloud







The dialog box is titled "Provider Selection" and contains the following configuration options:

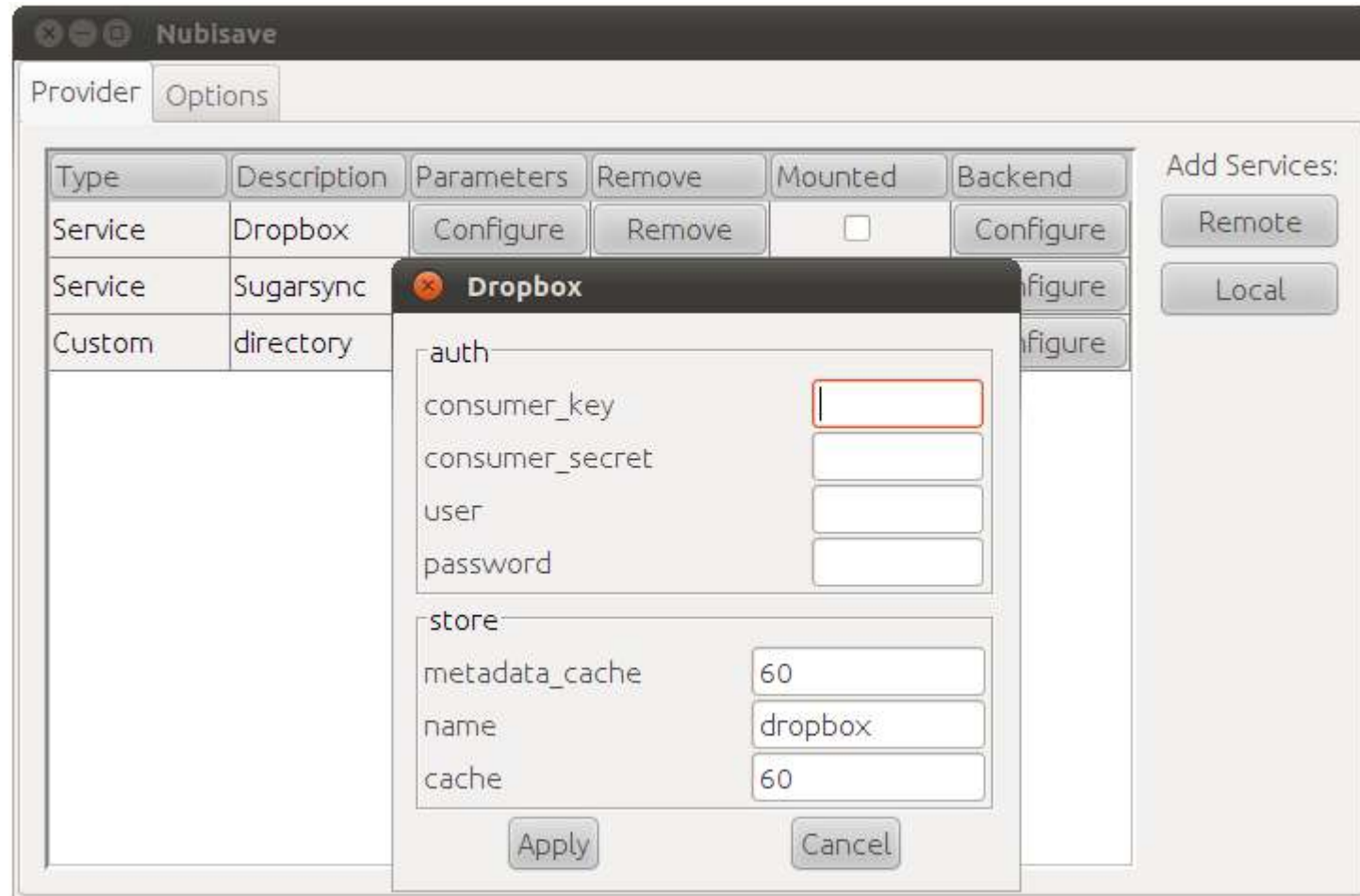
- Price per Data: 0.00 Euro, Low
- Price per Month: 0.00 Euro, Low
- Availability: 0.00 %, Low
- Bandwidth: 0.00 Mbps, Low
- max. Down Time: 0.00 Minute, Low
- Response Time: 0.00 Minute, Low

Below the configuration options is a search bar and a table of providers:

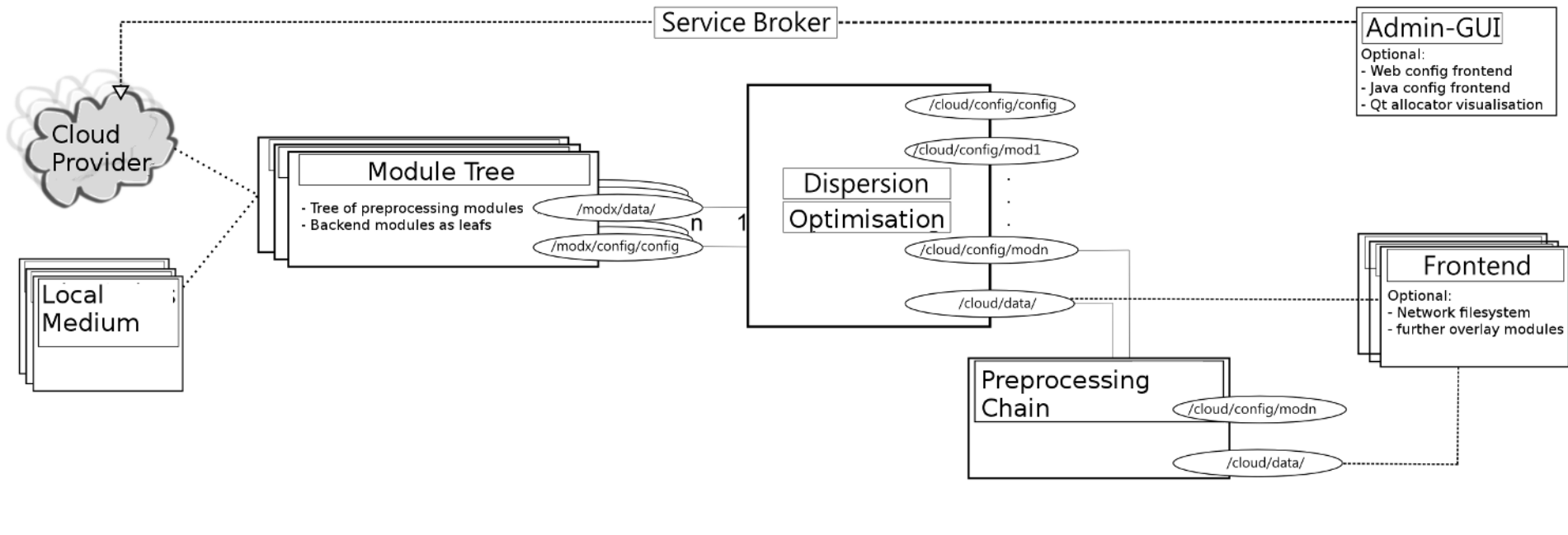
Name	P/D	P/M	A	BW	DT	RT
MS_SkyDr...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
GoogleSt...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Dropbox	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
STRATO_...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
AmazonS3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Sugarsync	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

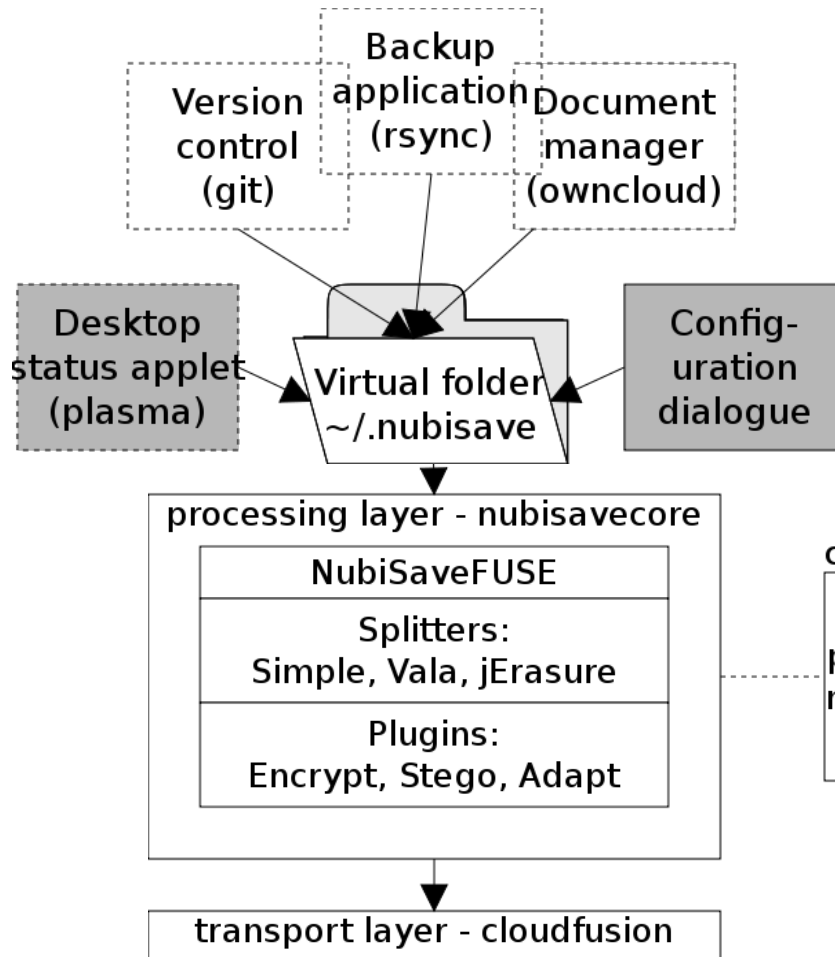
Buttons: cancel, Add selected Services

ConQo - Service registry/discovery with support for non-functional properties (NFPs)



Legend			
<b>Local Media:</b>	<b>Cloud Providers:</b>	<b>Backend Modules:</b>	<b>Preprocessing / Frontend:</b>
-NFS, -RAM -Hard disk -SAMBA	-Google -Amazon -Dropbox -SugarSync -OpenStack	-CloudFusion } Cloud -s3fs } Providers -davfs2 } -FTPFS } Local -SSHFS } Media	-WebDAV } Alternative -FTP } Frontend -HTTP } Interfaces -gzip FS } Compression -EncFS } Encryption -Splitter } Fragment creation -GenFuse } Compression, Steganography





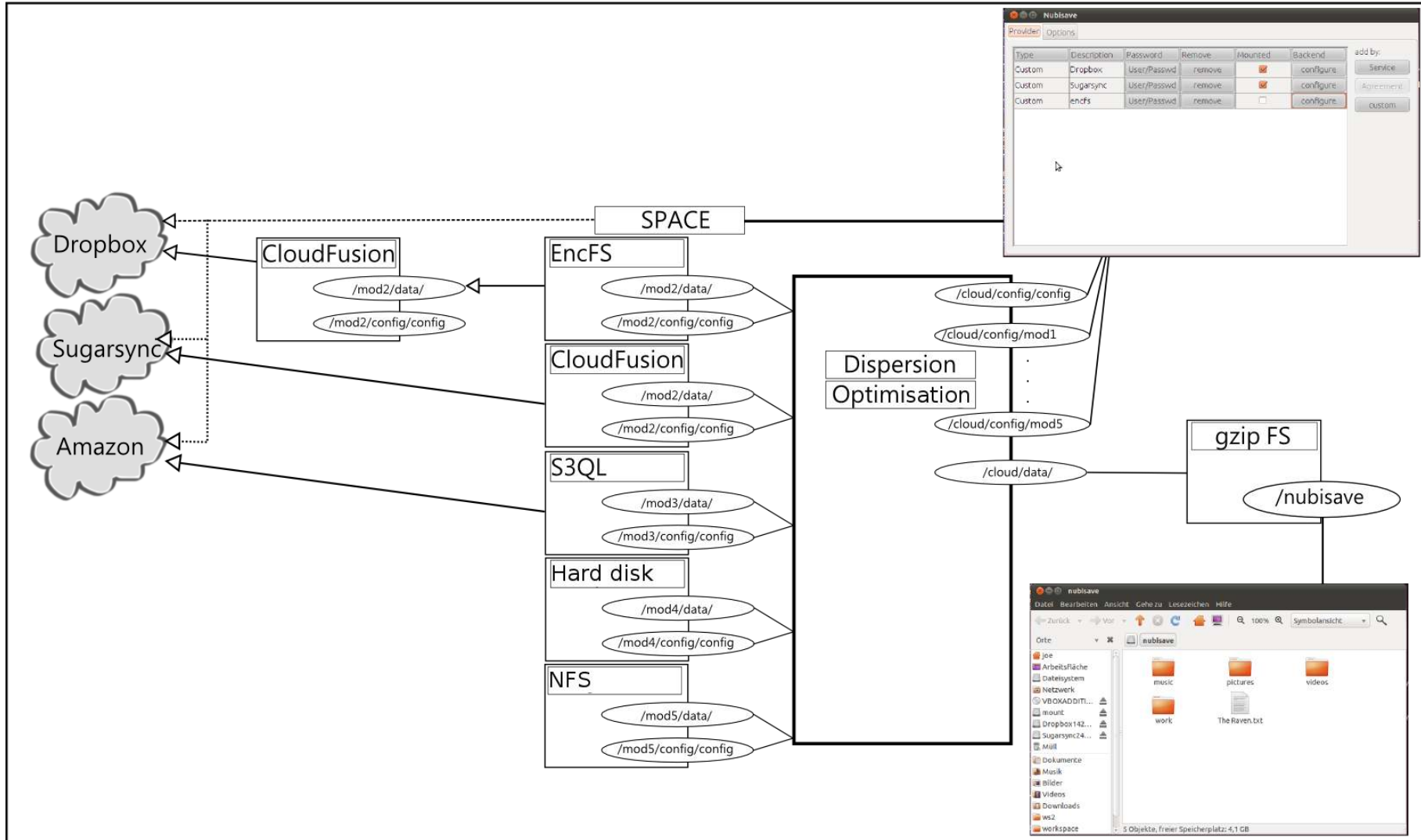
## Extension module: steganography

```

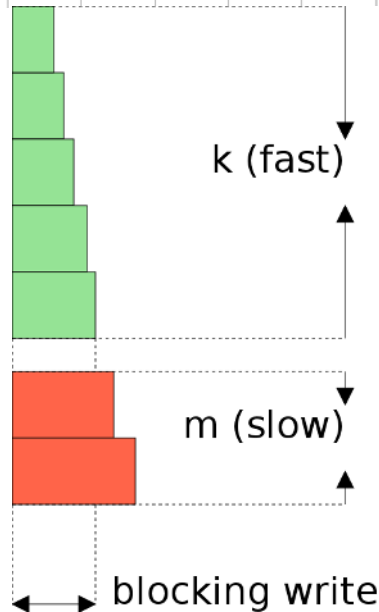
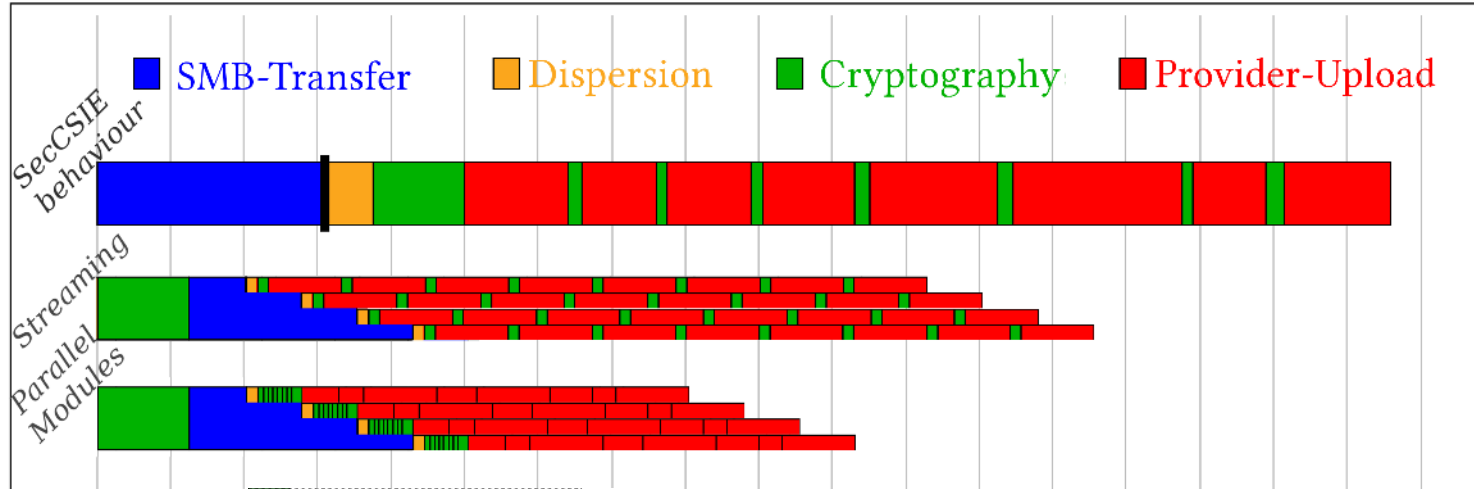
from util import linux
import tempfile

def decode(data):
    open("gzipfs.gz", "w").write(data)
    return linux.pipe_with_input_file(['gunzip -c $IN'],
        "gzipfs.gz")

def encode(data, path):
    open("gzipfs.gz", "w").write(data)
    return linux.pipe_with_input_file(['gzip -cf9 $IN'],
        "gzipfs.gz")
    
```

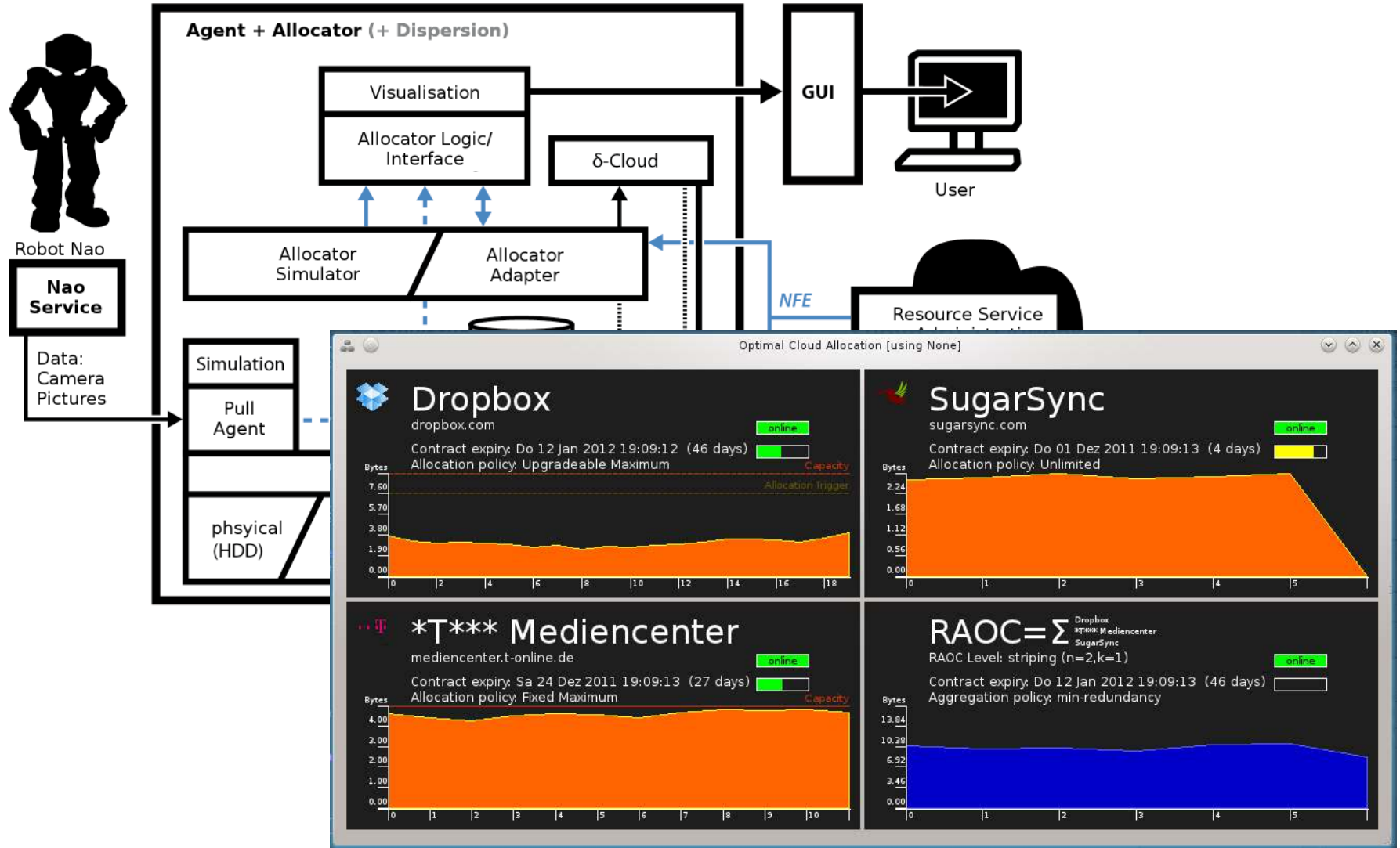






## Scheduling strategies

- round-robin
- fastest first



## Results

- Techniques for Redundant Arrays of Optimal Cloud Storage
- Blueprint architecture for building RAIC/RAOC systems
- Modular open source prototype NubiSave

## Next Steps

- Enhanced cloud storage provider catalogue/ontology
- Autonomous dynamic scaling
- High-throughput scenarios



## Further Material:

*nubisave.org*

- NubiSave web page, git repository, ...

*flexcloud.eu*

- EU/ESF-funded project: Flexible Service Architectures for Cloud Computing