CLOSER 2015
5th INTERNATIONAL CONFERENCE ON CLOUD COMPUTING AND SERVICES SCIENCE
20 - 22 MAY, 2015
LISBON, PORTUGAL

CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD

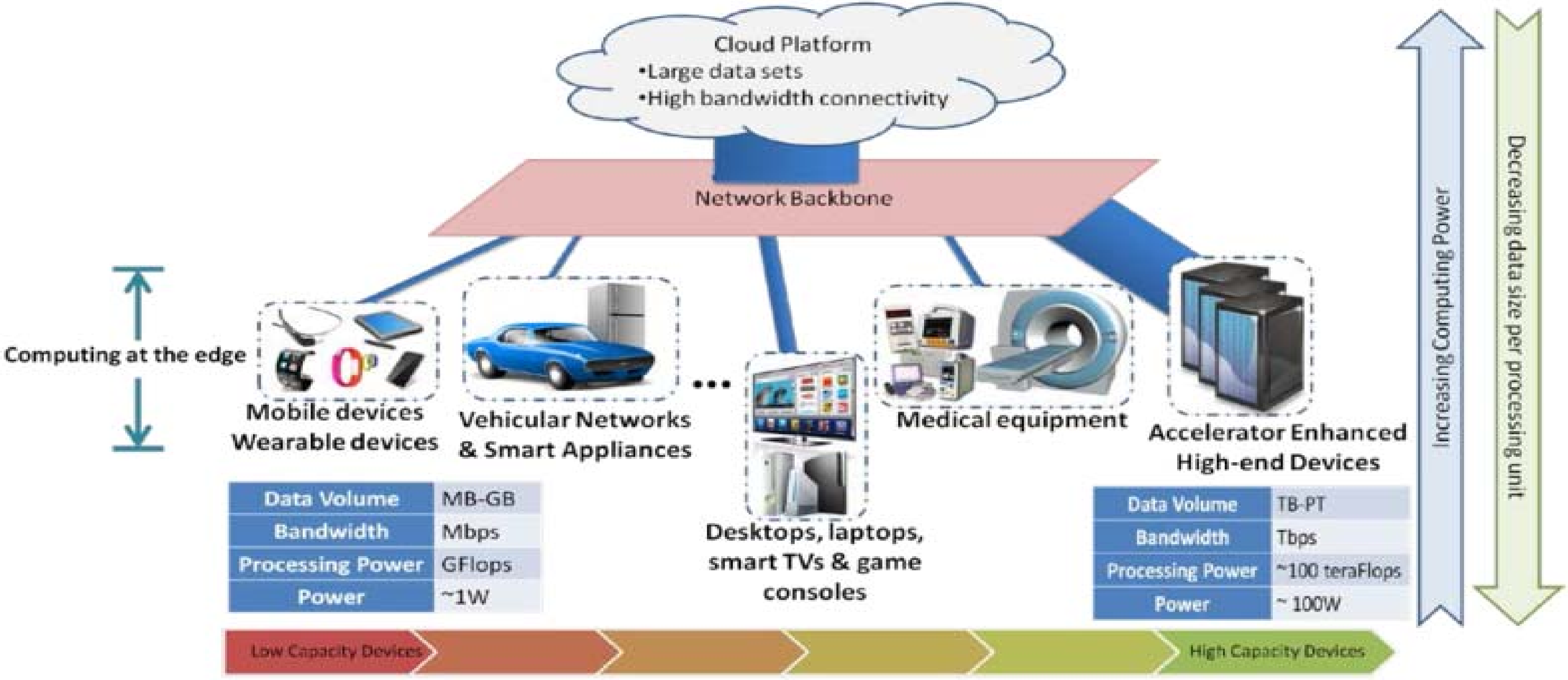
In-transit analytics on distributed Clouds: applications and architecture

Omer F. Rana
School of Computer Science & Informatics & Data Innovation Institute
Cardiff University, UK
ranaof@cardiff.ac.uk Twitter:@omerfrana

In collaboration with:
Manish Parashar, Javier Diaz-Montes, Mengong Zhou (Rutgers University, USA),
Ioan Petri, Tom Beach, Yacine Rezgui (Cardiff University),
Rafael Tolosana-Calasanz, Jose Banares (Univ. of Zaragoza, Spain)
Congduc Pham (University of Pau, France)

Cloud/Data Eco-System

- Increasing sensing capability closer to phenomenon being measured & increase volumes of “dynamic, distributed” data (IEEE P2413)
  - Capability to also undertake some processing on these devices
  - Increasing availability of programming support – “software defined environments”



Data Volume	MB-GB
Bandwidth	Mbps
Processing Power	GFlops
Power	~1W

Data Volume	TB-PT
Bandwidth	Tbps
Processing Power	~100 teraFlops
Power	~ 100W

Integrating Cloud Computing with Internet-of-Things

- "Cloud of Things" (CoT) and "Fog Computing"
  - Extending computing to the edges of the network
  - Overcoming latency constraints
- Real world/pervasive systems benefiting from Cloud infrastructure
  - Mobile & task off-loading (balancing energy usage with computation capability)
  - Internet-supported service convergence
- Significant heterogeneity in architectures and protocols for IoT
  - Device types and standards can vary significantly (e.g. iBeacons) -- development of "virtual sensors" (data reduction/fusion)
  - Often a data translation/mapping problem
- Projects:
  - Open Source IoTCloud (Sensors-as-a-Service): http://sites.google.com/site/opensourceiotcloud/
  - Open IoT (Middleware-oriented) – EU: http://www.openiot.eu/
- Commercial (mostly API based using HTTP/REST calls):
  - Xively (http://xively.com/), Open Sen.se/Internet of "Everything" (http://open.sen.se/), Think Speak (https://thingspeak.com/), Pacific Controls Gateway (http://pacificcontrols.net/products/galaxy.html)

Based on slides from Manish Parashar (Rutgers Univ.)

The Lure of Clouds

- An attractive platform for dynamic, real time service provisioning
  - Both for business & academia
- Cloud paradigm:
  - "Rent" resources as cloud services on-demand and pay for what you use
  - Potential for scaling-up, scaling-down and scaling-out, as well as for IT outsourcing and automation
  - Increasing support for dynamic deployment & configuration management
- Landscape of heterogeneous cloud services spans private & public clouds, data centers, etc.
  - Heterogeneous offering with different QoS, pricing models, availability, capabilities, and capacities
  - Variants: hybrid Clouds ("cloud bursting" & "cloud bridging" , Mobile off-loading, etc)
- Novel dynamic market-places where users can take advantage of different types of resources, quality of service (QoS), geographical locations, and pricing models
  - Various market models (on-demand, reservation, spot pricing, auctions, "Groupon", etc)
- Cloud federations extend as-a-service models to virtualized data-centers federations
  - Bring Your Own Cloud (cf. Bring Your Own Device)

Ideas from Rick McGreer (HP Labs/GENI Cloud)

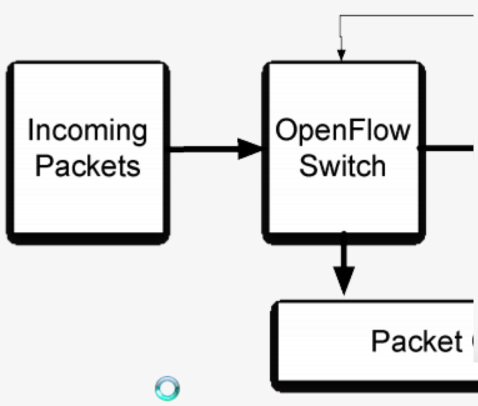
Opening the "Network" Layer

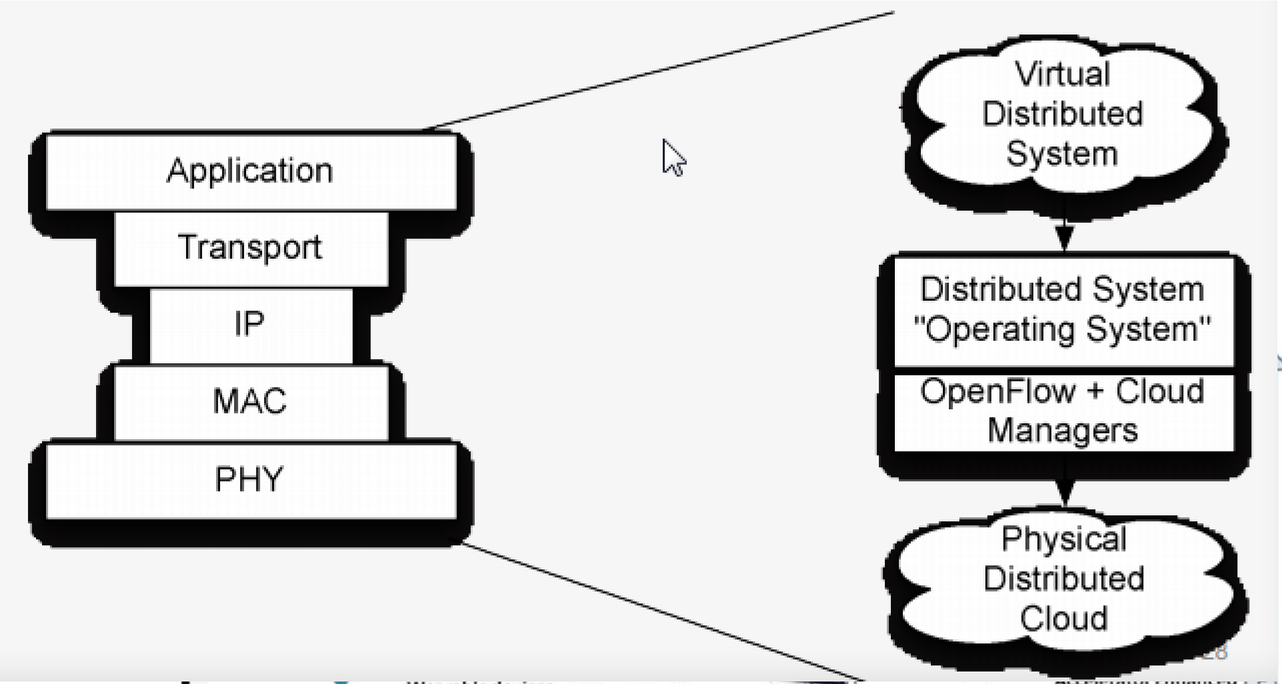
L4/L7 capability in-network, not just L2/L3 (as currently done) – i.e. application-layer admission control, security (DPI), routing, etc

OpenFlow Switches + MiddleBox Network Appliances (programmability)

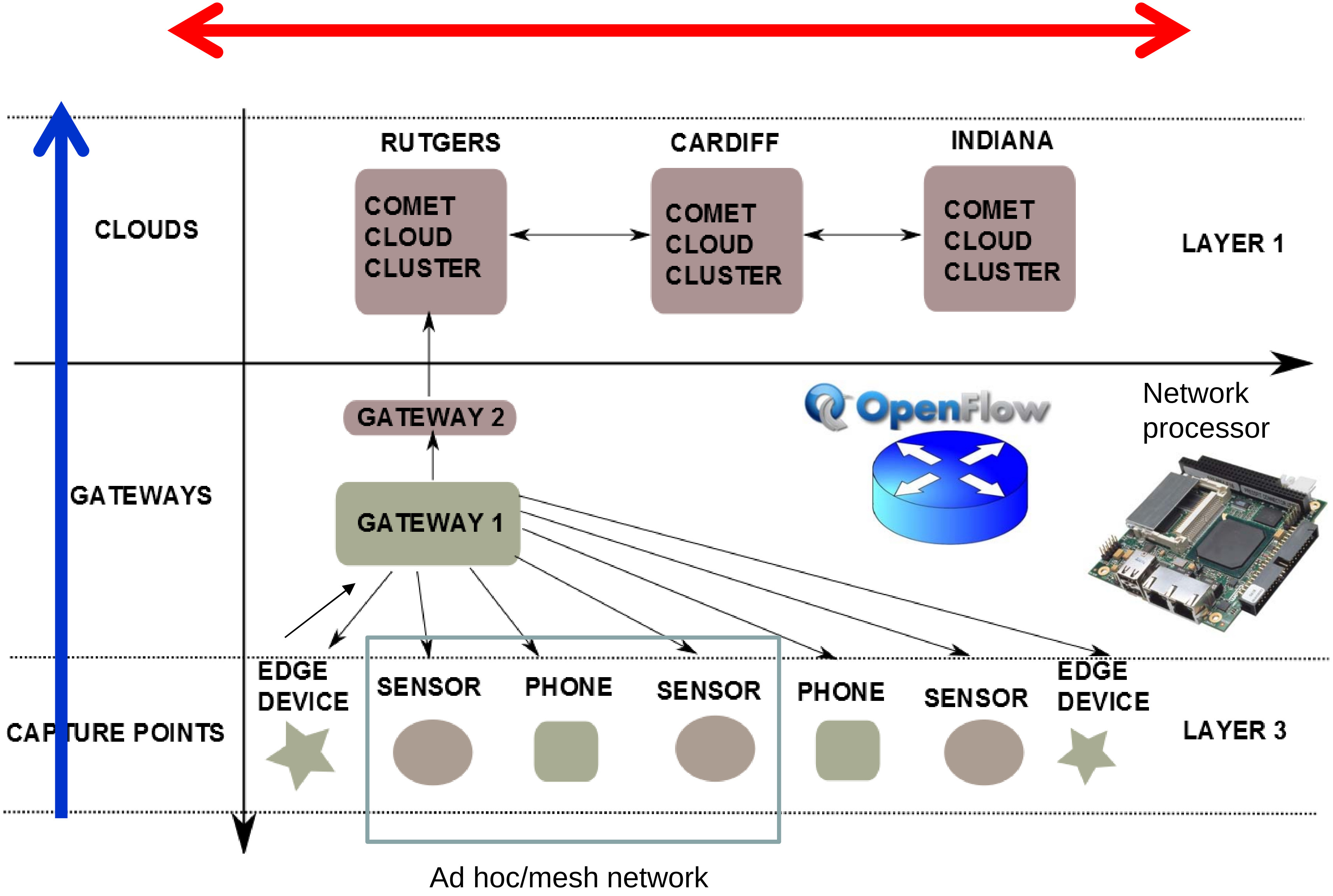
Merge the Cloud and the network through in-network dynamic,

Abstract "in-network" en services

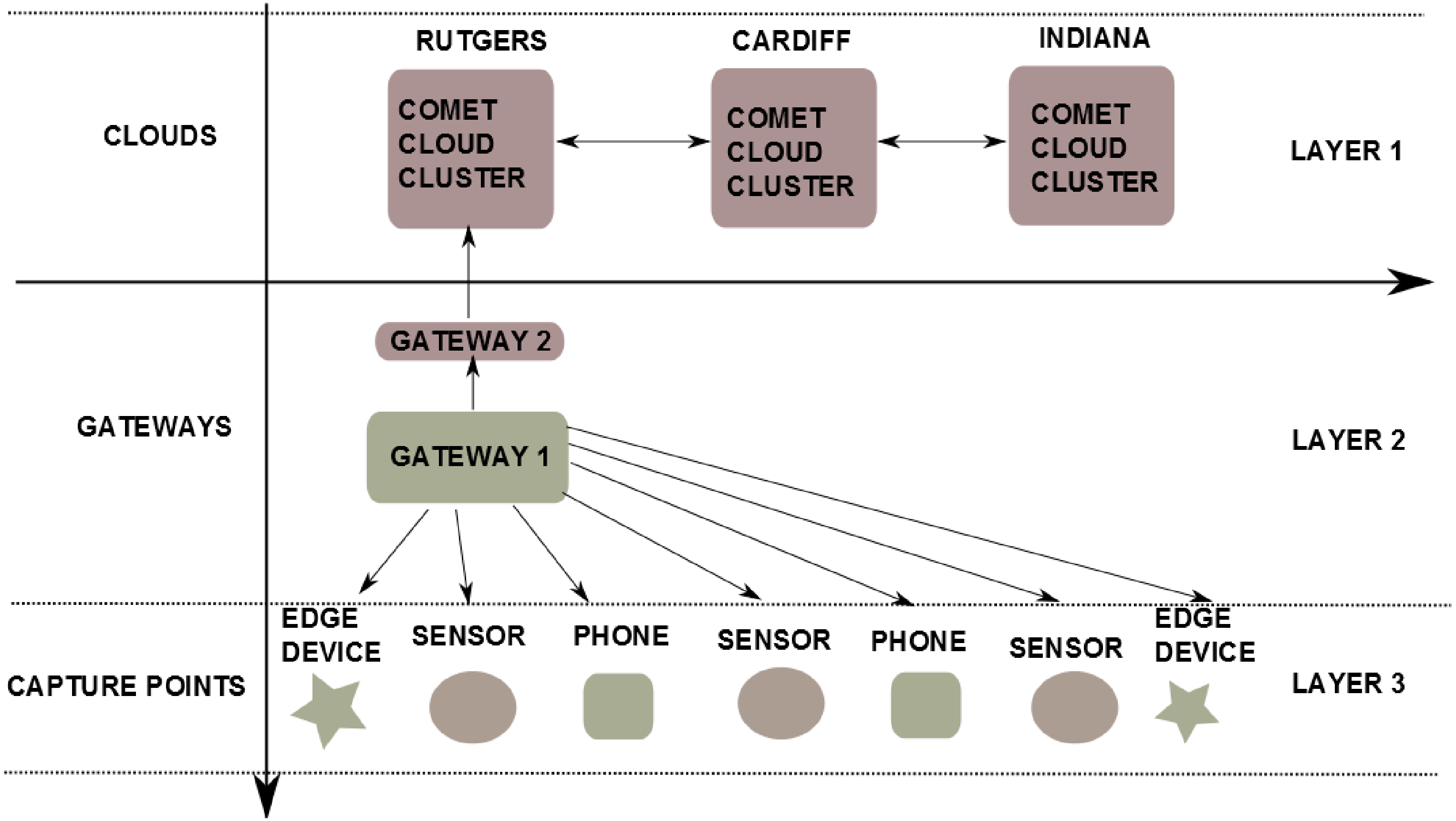




Multi-Layered Federated Clouds



Multi-Layered Federated Clouds



- Multiple data access and processing layers
- **Deciding what to do where – creation of a "decision function"**
- Different objectives: L3: power, range; L2: stream aggregation; L1: throughput
- **No need to migrate "raw" data to Cloud systems**

Data Analytics on Multi-Layered Clouds

- In-network capability:
  - Application driven, multi-node/capability driven
- Analytics:
  - In-situ (aggregation or capture site) – most common (e.g. Apache Spark, Hadoop, other in-memory, etc)
  - Data-drop (on-demand, “elastic”) – e.g. use of shared folders
  - In-transit (distributed, partial)
- New class of analysis algorithms
  - Resource-aware analytics (capacity, capability, availability)
    - Constraints influence types of analysis undertaken
  - Influenced by resource constraints (I/O, power, cost, historic performance)
- Workflow/Pipelines across layers
  - Dynamically adapt over time
  - Scale (in/out) with resource availability
  - Operation types vary in complexity & data size

AWS Lambda -- compute nodes charged by 100ms -- not the hour. First 1M node.js exec/month for free -- a monitoring challenge (http://aws.amazon.com/lambda/)

Talk Roadmap

- Application scenarios for Federated Clouds
  - Analysis pipelines
- Modelling (abstractions) for federated Clouds
  - Use of Reference nets (a type of Petri net)
  - Model is directly executable
- Concluding scenario
  - Cloud-based building data analytics

Types of applications

- Variety of applications in multimedia streaming
  - Computational Science with sensor coupling – e.g. emergency response, security, environment, etc
  - Processing requirements vary – over different timeframes
- Not just true for physical sciences
  - increasingly social scientists also face similar challenges (e.g. tension indicators in communities)
- Increasing availability of data over the Web and from government departments
  - Data from Facebook, Twitter, Flickr (text, audio, video, etc)
    - People as sensors
  - Data from government agencies – Police API, Demographic data (ONS), etc

Application 1:

EPSRC

Technology Strategy Board
Driving Innovation

cost
EUROPEAN COOPERATION
IN SCIENCE AND TECHNOLOGY

Data Streaming and Complex Event Processing

Bañares, José Ángel, Rana, Omer, Tolosana-Calasanz, Rafael and Pham, Congduc. ”Revenue creation for rate adaptive stream management in multi-tenancy environments”. Lecture Notes in Computer Science 8193, pp. 122-137.Springer.

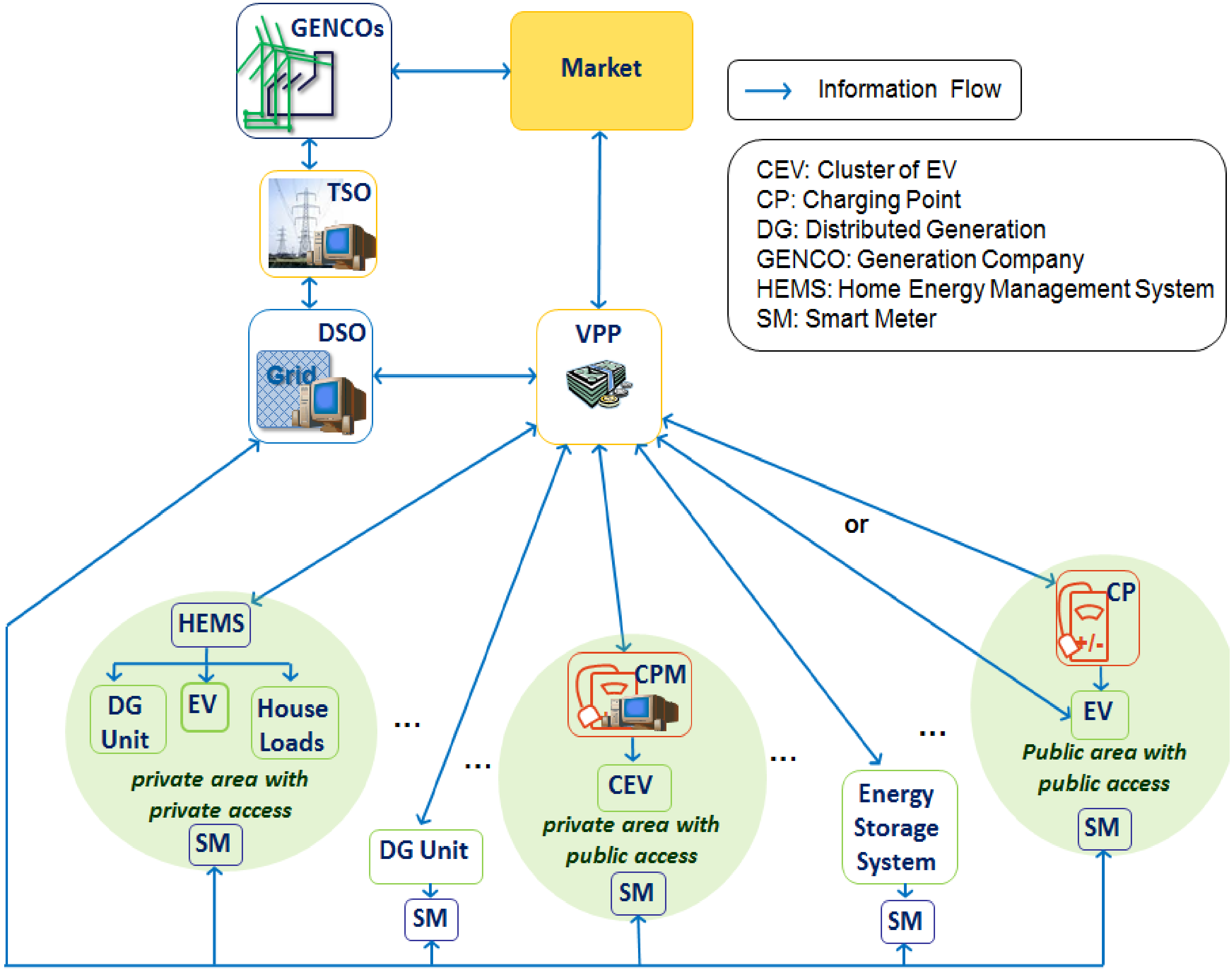
Tolosana-Calasanz, Rafael, Banares, Jose Angel, Cipcigan, Liana, Rana, Omer, Papadopoulos, Panagiotis andPham, Congduc. ”A Distributed In-Transit Processing Infrastructure for Forecasting Electric Vehicle Charging Demand”. Presented at: 2013 13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing (CCGrid), Delft, Netherlands, 13-16 May 2013.

Rafael Tolosana-Calasanz, José Á. Bañares, Omer Rana, Congduc Pham, Erotokritos Xydas, Charalampos Marmaras, Panagiotis Papadopoulos and Liana Cipcigan, “Enforcing QoS on OpenNebula-based Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams”, to be presented at DPMSS workshop (from Sensor Networks to Clouds), at 14th IEEE/ACM Int. Symp. On Cluster, Cloud and Grid Computing (CCGrid), Chicago, May 2014.

Virtual Power Plants & Electric Vehicles

http://www.eandfes.co.uk/

A small sized
Smart Grid



⟶ Information Flow

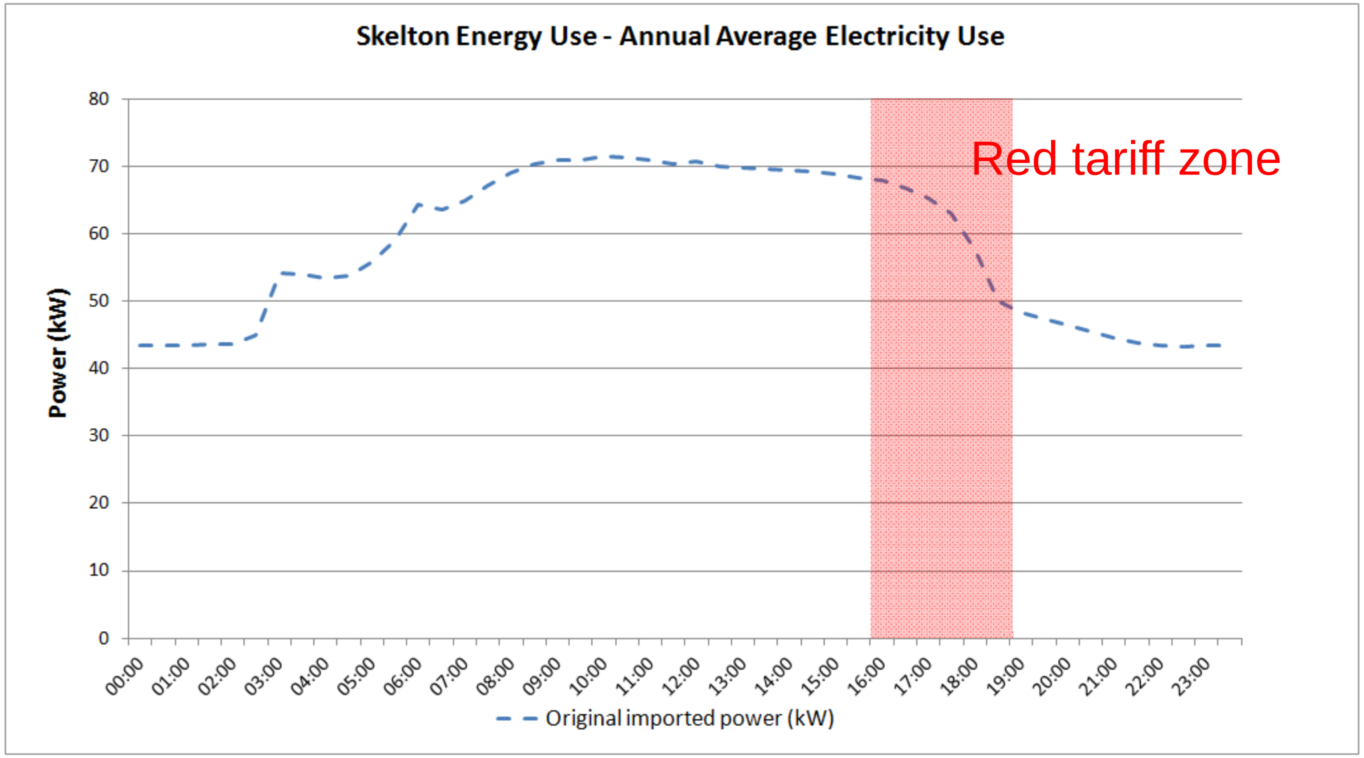
CEV: Cluster of EV
CP: Charging Point
DG: Distributed Generation
GENCO: Generation Company
HEMS: Home Energy Management System
SM: Smart Meter

Innovate UK
Technology Strategy Board
KAMFutures
innovative solutions
cenex
CARDIFF UNIVERSITY PRIFYSGOL CAERDYDD
energy saving trust | ymddiriedolaeth arbed ynni

http://www.eandfes.co.uk/

Case study scenario

- Skelton building annual average electricity use per day

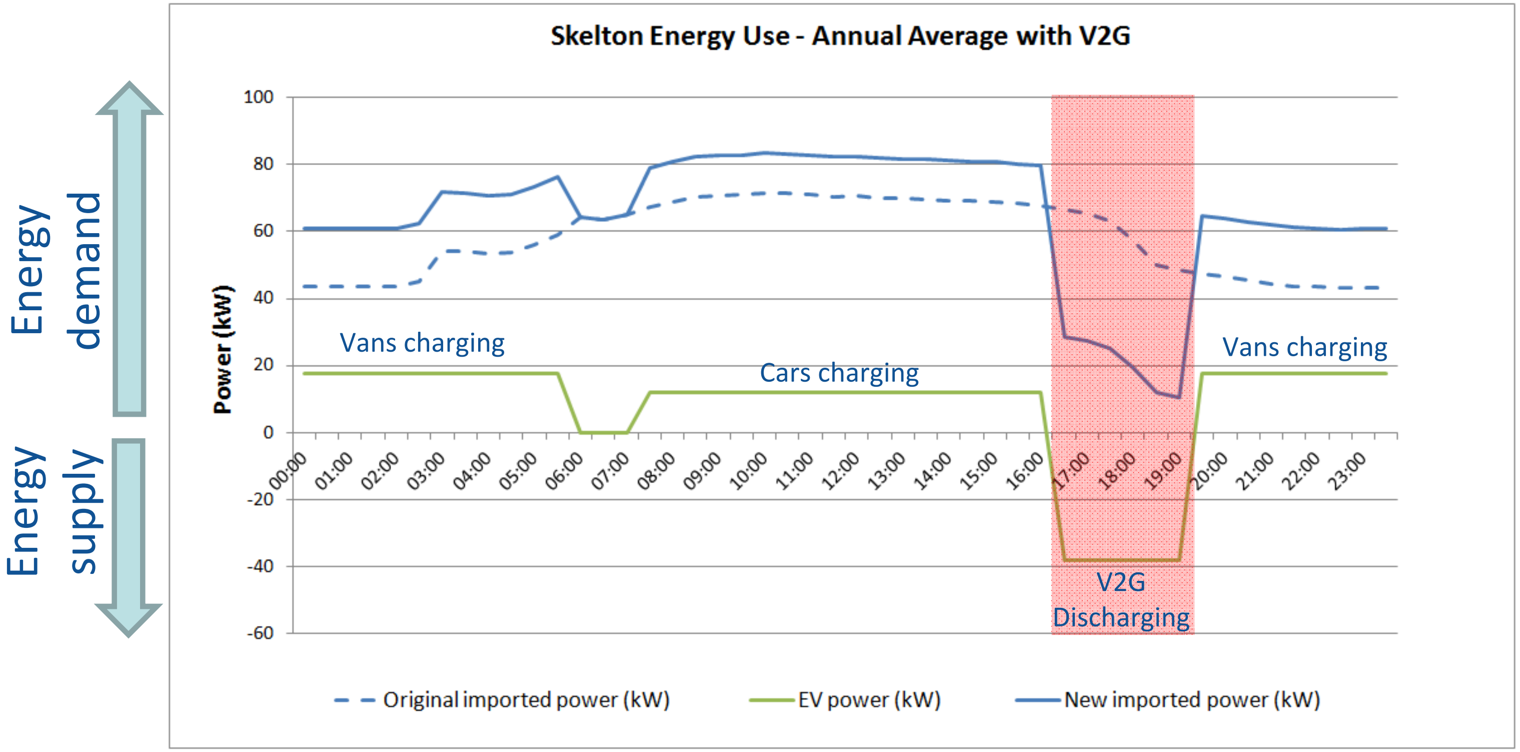


- 10 passenger cars (24 kWh battery) , 5 maintenance vans (55kWh battery)
- 20% battery SoC available for V2G
- Recharging for 20 miles per day in car EV and 40 miles per day in van
- Solar scenarios 0 kWp, 30kWp, 60 kWp, 90 kWp

http://www.eandfes.co.uk/

Case study scenario

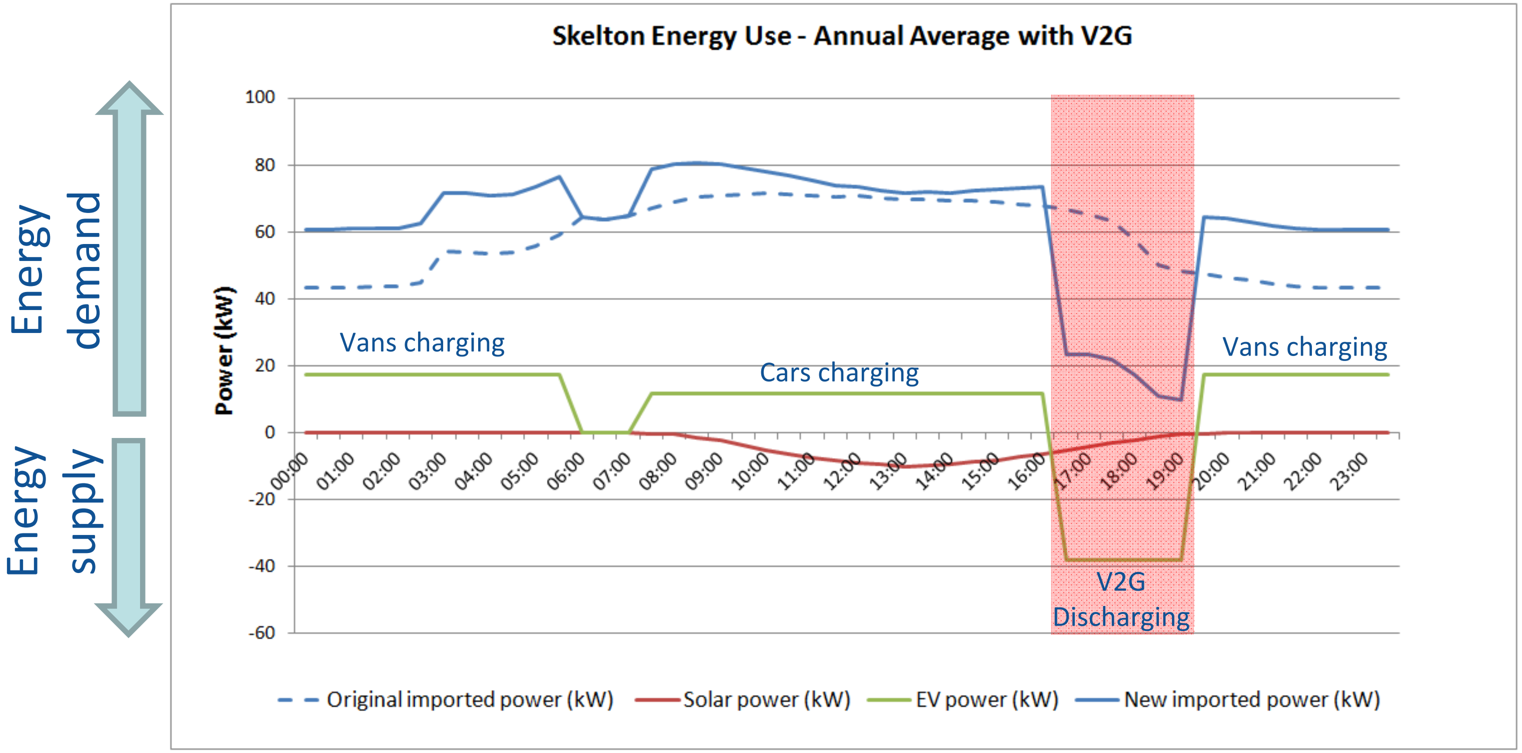
- Skelton building annual average electricity use per day



- 0 kWp Solar installation

Case study scenario

- Skelton building annual average electricity use per day

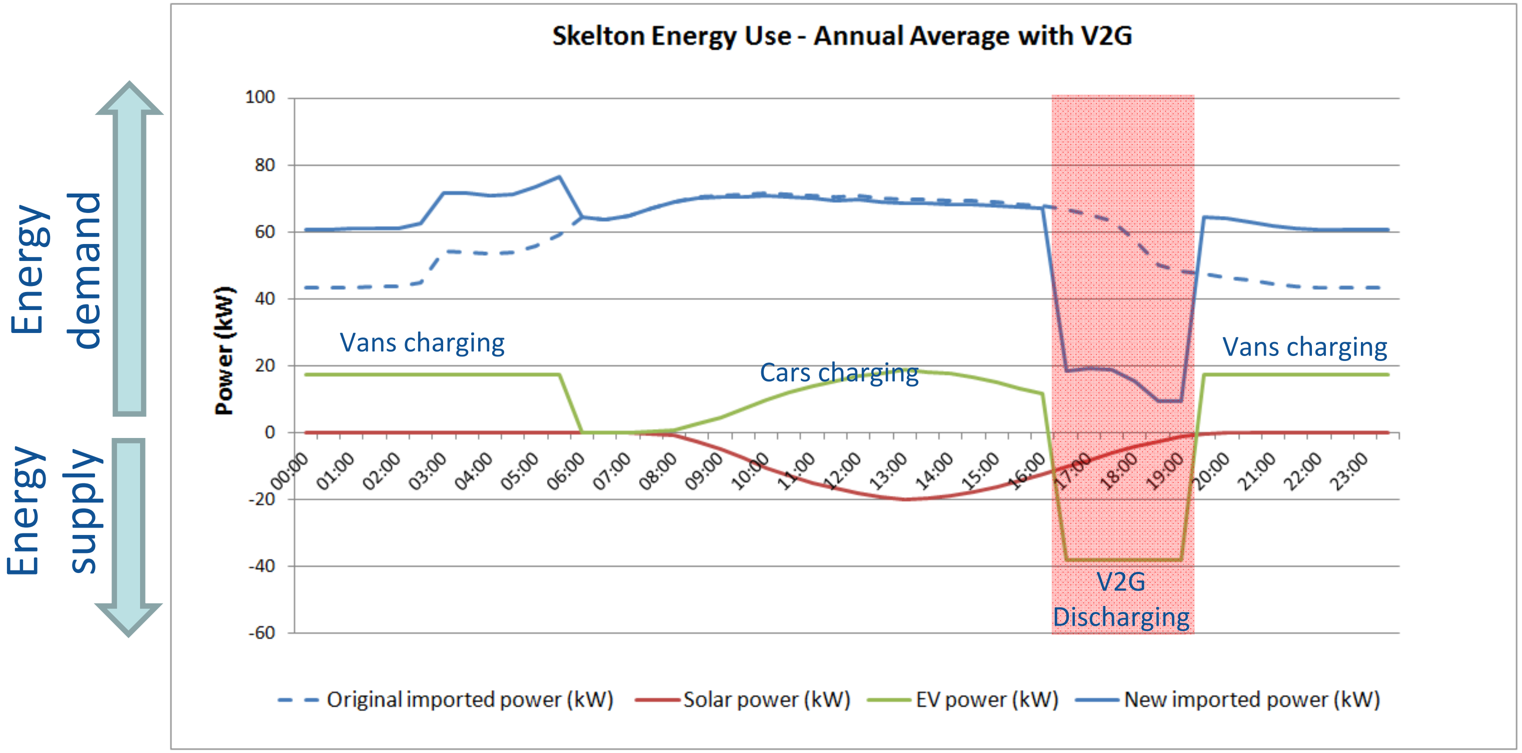


- 30 kWp Solar installation

http://www.eandfes.co.uk/

Case study scenario

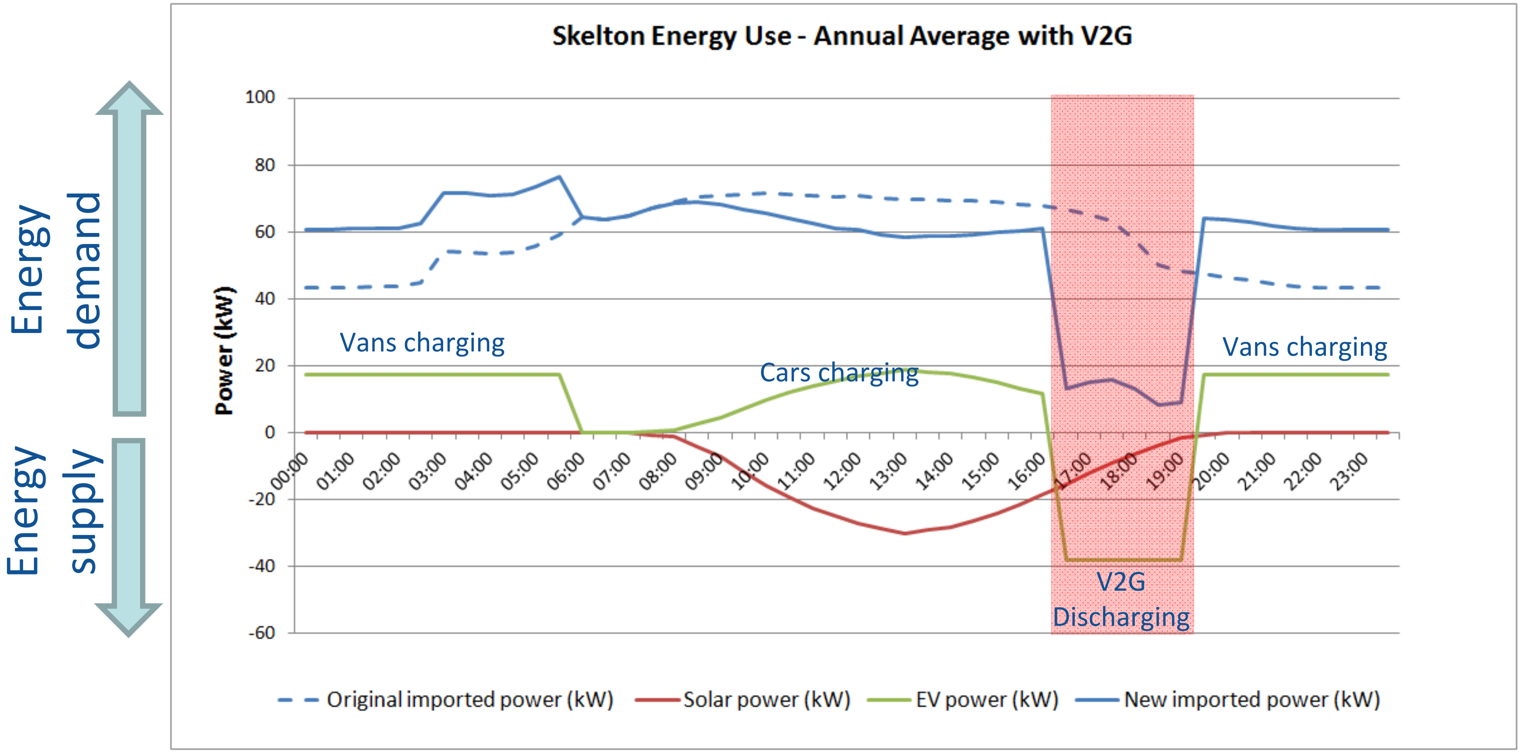
- Skelton building annual average electricity use per day



- 60 kWp Solar installation
- Solar PV tracking

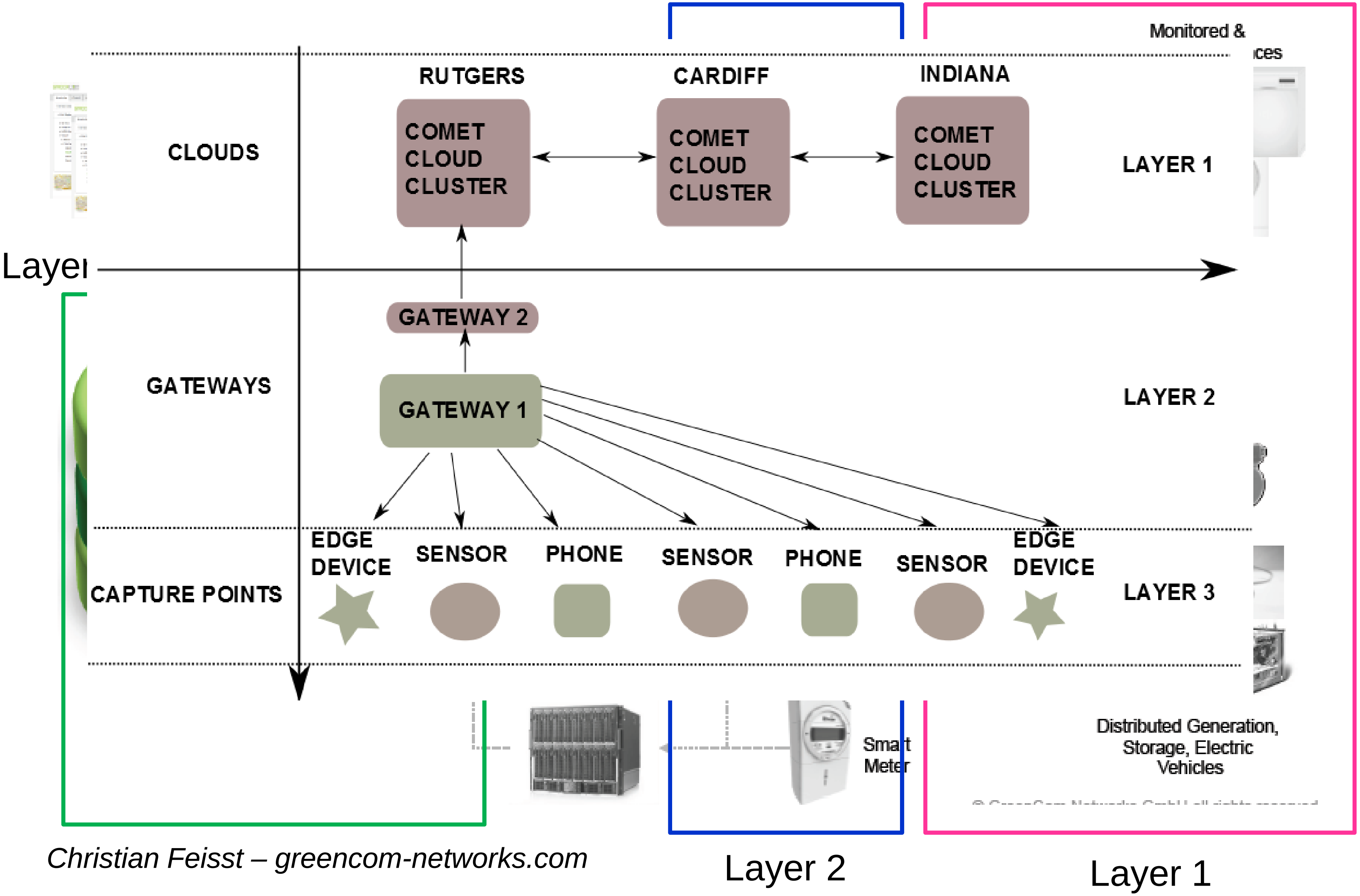
Case study scenario

- Skelton building annual average electricity use per day



- 90 kWp Solar installation
- Solar PV tracking

Data Collection for Brokerage



Monitored & ...ces

CLOUDS

RUTGERS
COMET CLOUD CLUSTER

CARDIFF
COMET CLOUD CLUSTER

INDIANA
COMET CLOUD CLUSTER

LAYER 1

Layer

GATEWAYS

GATEWAY 2

GATEWAY 1

LAYER 2

CAPTURE POINTS

EDGE DEVICE

SENSOR

PHONE

SENSOR

PHONE

SENSOR

EDGE DEVICE

LAYER 3

Smart Meter

Distributed Generation, Storage, Electric Vehicles

Layer 2

Layer 1

Christian Feisst – greencom-networks.com

Cloud-based Regional Brokerage (Virtual Power Plants)

Technology Partners

cenex

energy saving trust

KAM Futures

moixa TECHNOLOGY

CARDIFF UNIVERSITY PRIFYSGOL CAERDYDD

msp manchester science parks

POTENZA TECHNOLOGY

WMG Innovative Solutions

Demonstration Sites;

msp manchester science parks

rjcls

Funded by;

EPSRC Engineering and Physical Sciences Research Council

Innovate UK Technology Strategy Board

Advisory Board;

ecotricity

electricity north west

e.on

CISCO

MITSUBISHI MOTORS

Sir Robert McALPINE

WESTERN POWER DISTRIBUTION Serving the Midlands, South West and Wales

Application 2:
Analysing social media data

Conejero, Javier, Rana, Omer, Burnap, Peter, Morgan, Jeffrey, Carrion, Carmen and Caminero, Blanca,"Characterising the power consumption of Hadoop Clouds: A social media analysis case study". CLOSER 2013: 3rd International Conference on Cloud Computing and Services Science, Aachen, Germany, 8-10 May 2013.

Sloan, Luke, Morgan, Jeffrey, Housley, William, Williams, Matthew Leighton, Edwards, Adam Michael, Burnap, Peter and Rana, Omer, "Knowing the Tweeters: Deriving sociologically relevant demographics from Twitter". Sociological Research Online 18 (3) , 7, 2013.

Conejero, Javier, Burnap, Peter, Rana, Omer and Morgan, Jeffrey. "Scaling archived social media data analysis using a Hadoop Cloud". Presented at: IEEE 6th International Conference on Cloud Computing (CLOUD), Santa Clara, CA, USA, 27 June - 2 July 2013.

Burnap, Peter, Rana, Omer, Avis, Nicholas John, Williams, Matthew Leighton, Housley, William, Edwards, Adam Michael, Morgan, Jeffrey and Sloan, Luke. "Detecting tension in online communities with computational Twitter analysis". Technological Forecasting & Social Change, 2013. 10.1016/j.techfore.2013.04.013.

Burnap, Peter, Avis, Nicholas John and Rana, Omer. "Making sense of self-reported socially significant data using computational methods". International Journal of Social Research Methodology 16 (3) , pp. 215-230. 2013. 10.1080/13645579.2013.774174

Williams, Matthew Leighton, Edwards, Adam Michael, Housley, William, Burnap, Peter, Rana, Omer, Avis, Nicholas John, Morgan, Jeffrey and Sloan, Luke. "Policing cyber-neighbourhoods: Tension monitoring and social media networks". Policing and Society, 2013. 10.1080/10439463.2013.780225

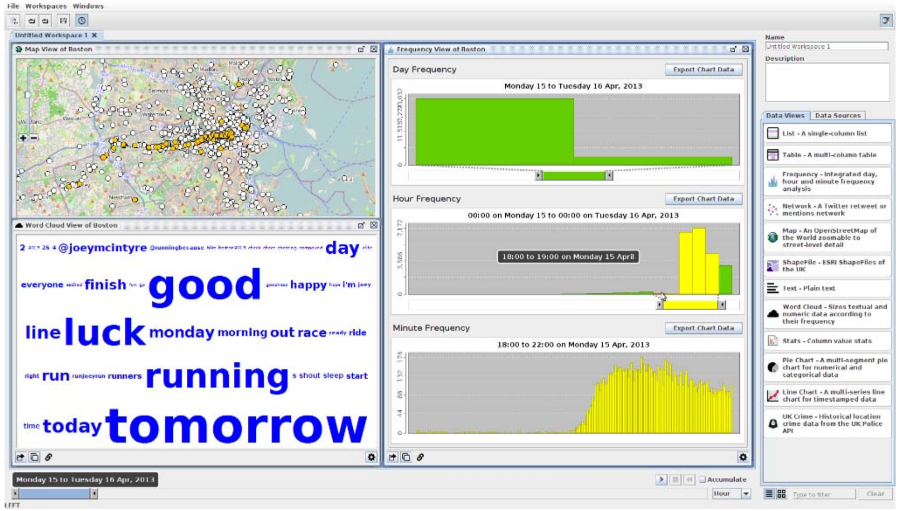
Burnap, P., Rana, O., Williams, M., Housley, W., Edwards, A., Morgan, J, Sloan, L. and Conejero, J. (2014) 'COSMOS: Towards an Integrated and Scalable Service for Analyzing Social Media on Demand', International Journal of Parallel, Emergent and Distributed Systems, Taylor & Francis

Social Media Analysis

- Significant quantities of data generated from social media (… but "ethical" usage important)
  - Twitter: "firehose" (100%), "gardenhose" (10%), "spritzer" (1%)
  - Facebook status updates
- Integrating this data with other sources
  - ONS (in the UK) + other curated data
  - Maps related: (various options: Open Street Maps, Google Maps, Yahoo! Placefinder etc)
- Raw data not significant
  - Looking for particular types of "events" of interest
- Common analysis types
  - Sentiment and Opinion analysis
  - Connectivity between content generators
- Collaborative On-line Social Media Observatory (COSMOS)
  - "Tension" indicators in terrestrial and on-line communities
  - Integrating data with other (conventional) indicators

E·S·R·C ECONOMIC & SOCIAL RESEARCH COUNCIL Google AIRBUS GROUP *http://www.cosmosproject.net/* NCRM National Centre for Research Methods JISC inspiring innovation

COSMOS Web Observatory



Name: Untitled Workspace 1

- List - A single-column list
- Table - A multi-column table
- Frequency - Integrated day, hour and minute frequency analysis
- Network - A Twitter retweet or mentions network
- Map - An OpenStreetMap of the World zoomable to street-level detail
- ShapeFile - ESRI ShapeFiles of the UK
- Text - Plain text
- Word Cloud - Sizes textual and numeric data according to their frequency
- Stats - Column value stats
- Pie Chart - A multi-segment pie chart for numerical and categorical data
- Line Chart - A multi-series line chart for timestamped data
- UK Crime - Historical location crime data from the UK Police API

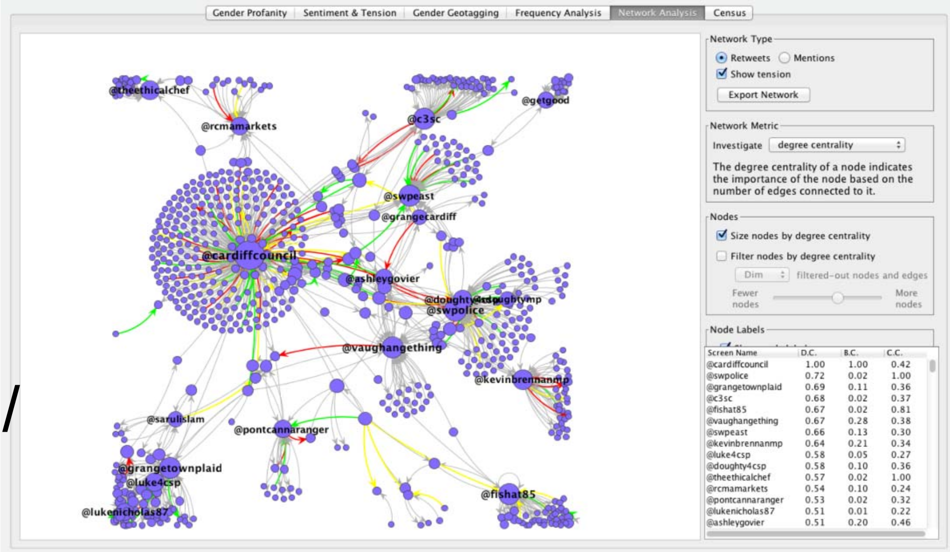
Usable – *developed with social scientists for social scientists*

Reproducible/Citable Research - *export/share workflow*

Integrated

Open ("plug and play")

Scalable (MongoDB data stores/ Hadoop Back End)



Network Type
● Retweets ○ Mentions
☑ Show tension

Network Metric
Investigate: degree centrality

The degree centrality of a node indicates the importance of the node based on the number of edges connected to it.

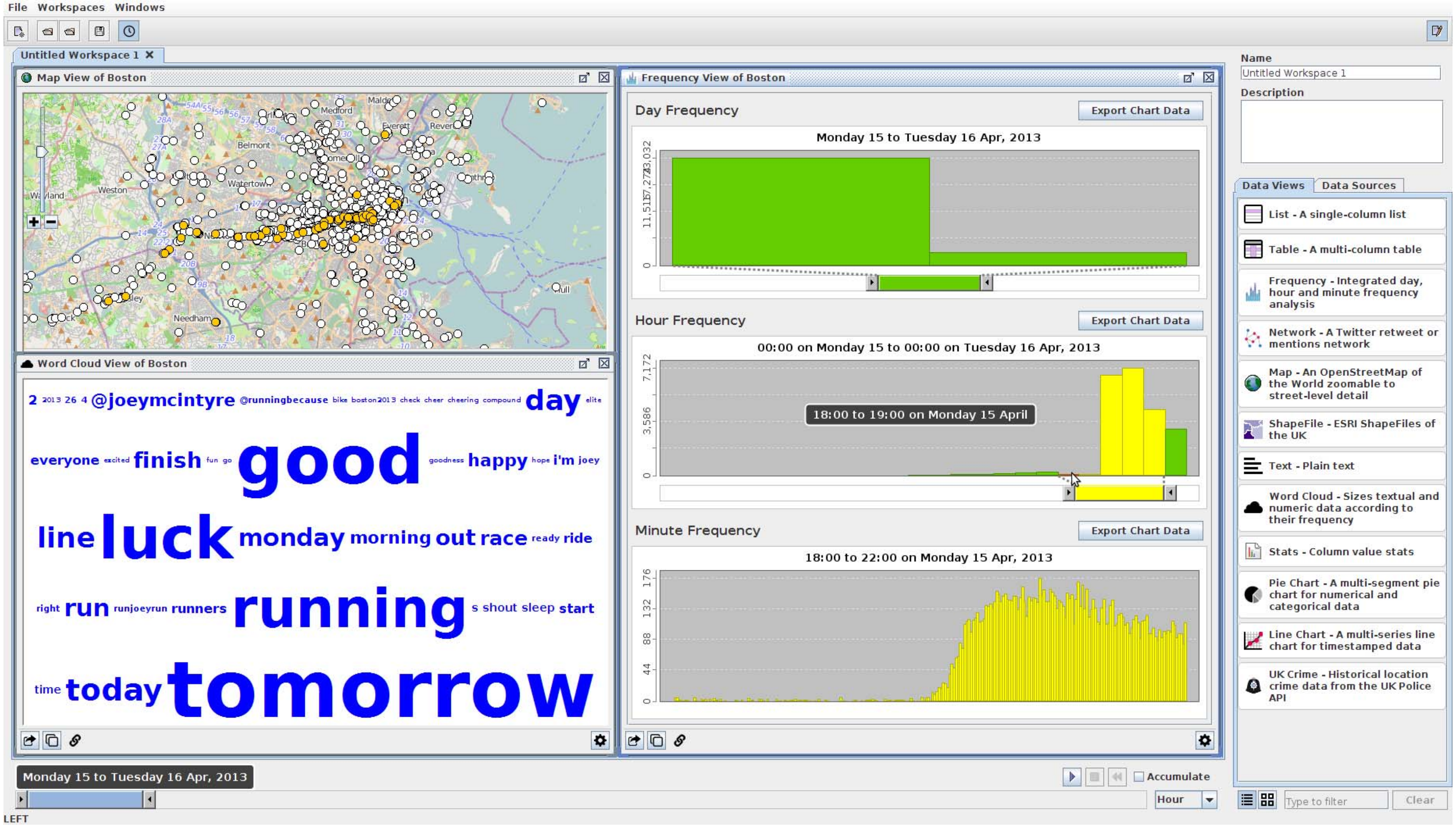
Nodes
☑ Size nodes by degree centrality
☐ Filter nodes by degree centrality
Dim filtered-out nodes and edges

Node Labels

Screen Name	D.C.	B.C.	C.C.
@cardiffcouncil	1.00	1.00	0.42
@swpolice	0.72	0.02	1.00
@grangetownplaid	0.69	0.11	0.36
@c3sc	0.68	0.02	0.37
@fishat85	0.67	0.02	0.81
@vaughangething	0.67	0.28	0.38
@swpeast	0.66	0.13	0.30
@kevinbrennanmp	0.64	0.21	0.34
@luke4csp	0.58	0.05	0.27
@doughty4csp	0.58	0.10	0.36
@theethicalchef	0.57	0.02	1.00
@rcmamarkets	0.54	0.10	0.24
@pontcannaranger	0.53	0.02	0.32
@lukenicholas87	0.51	0.01	0.22
@ashleygovier	0.51	0.20	0.46

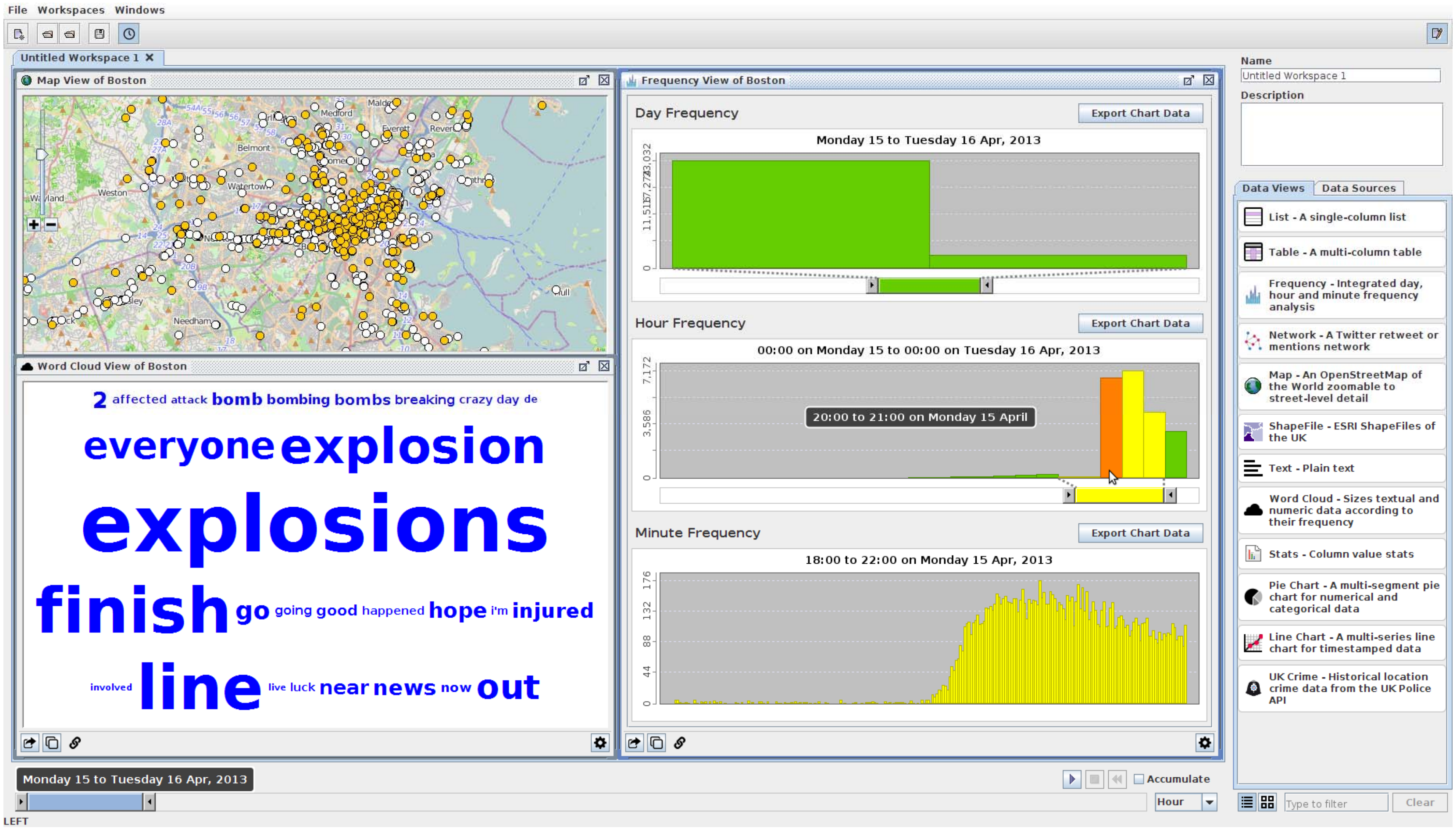
**Burnap, P. et al. (2014) '*COSMOS: Towards an Integrated and Scalable Service for Analyzing Social Medi*
on Demand', International Journal of Parallel, Emergent and Distributed Systems**

Observing Events (Boston)



http://www.cosmosproject.net/

Observing Events



Name: Untitled Workspace 1

Description:

- List - A single-column list
- Table - A multi-column table
- Frequency - Integrated day, hour and minute frequency analysis
- Network - A Twitter retweet or mentions network
- Map - An OpenStreetMap of the World zoomable to street-level detail
- ShapeFile - ESRI ShapeFiles of the UK
- Text - Plain text
- Word Cloud - Sizes textual and numeric data according to their frequency
- Stats - Column value stats
- Pie Chart - A multi-segment pie chart for numerical and categorical data
- Line Chart - A multi-series line chart for timestamped data
- UK Crime - Historical location crime data from the UK Police API

http://www.cosmosproject.net/

COSMOS Infrastructure

COSMOS Desktop

- Small local datasets
- Users' API credentials
- Local analysis
- Sept '14 launch
(>100 dl's in 17 countries)

COSMOS Cloud

- Scalable storage
  - Massive datasets (MongoDB)
- Scalable compute
  - On-demand nodes
  - Fast search & retrieve
  - Fast analysis
  - Indexing challenge
- Workflow management
- Collaboration support
- 2015 launch

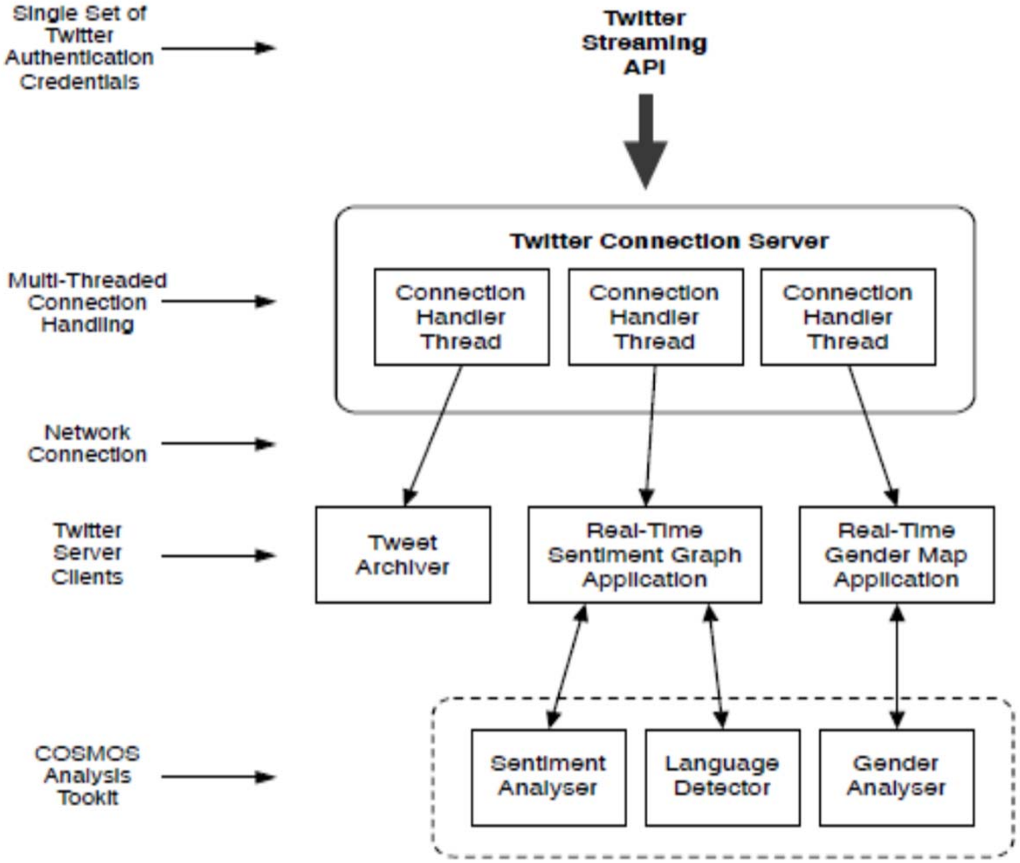
Data Collection

Persistent connection to
Twitter 1% Stream (~4 billion)
ONS/Police API
Drag and drop RSS
Import CSV/JSON

Data Transformation

Word Frequency
Point data frequency over time
Social Network Analysis
Geospatial Clustering
Sentiment Analysis
…API to plug new modules
and benchmark tools

COSMOS: Architecture



Layer 1: Data generation (twitter feed) – can reduce captured data

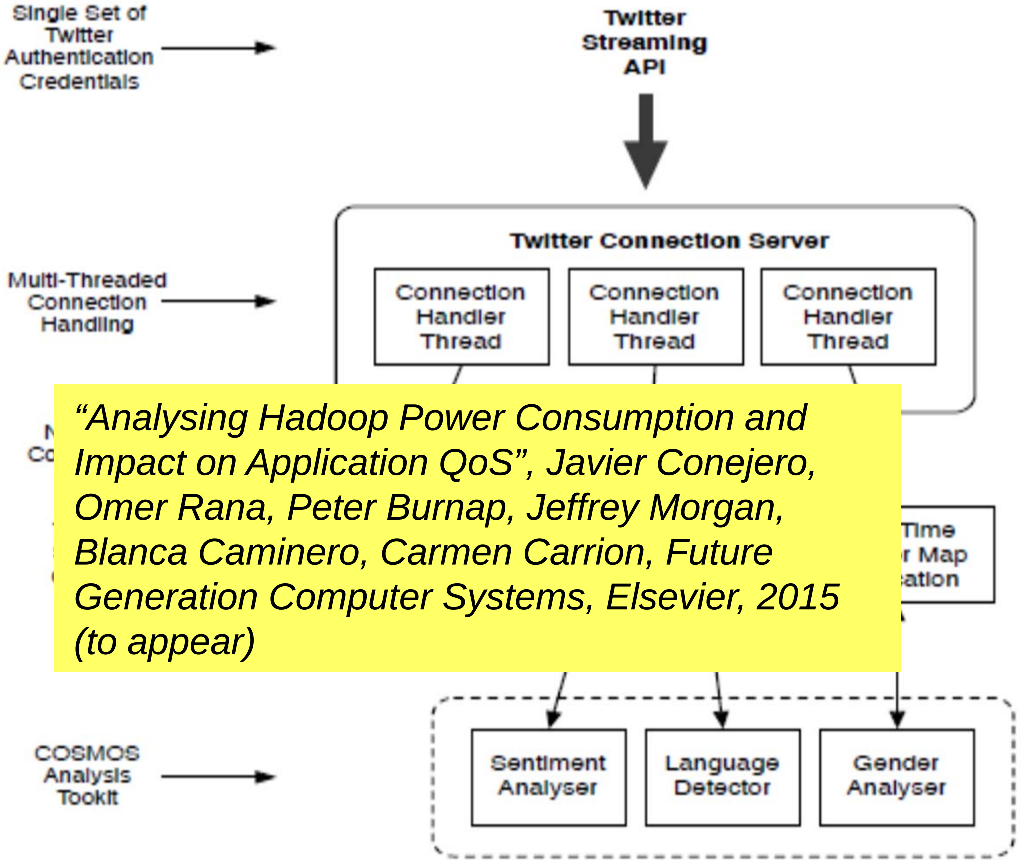
Layer 2: COSMOS filters (gender or sentiment analysis)

Layer 3: Data analysis and integration with other sources (Police API, Demographic data (ONS), etc)

- COSMOS integrates a variety of different services:

  Gender analysis, sentiment analysis, Open Street maps
- Can be integrated with user supplied services

COSMOS: Architecture



"Analysing Hadoop Power Consumption and Impact on Application QoS", Javier Conejero, Omer Rana, Peter Burnap, Jeffrey Morgan, Blanca Caminero, Carmen Carrion, Future Generation Computer Systems, Elsevier, 2015 (to appear)

Layer 1: Data generation (twitter feed) – can reduce captured data

Layer 2: COSMOS filters (gender or sentiment analysis)

Layer 3: Data analysis and integration with other sources (Police API, Demographic data (ONS), etc)

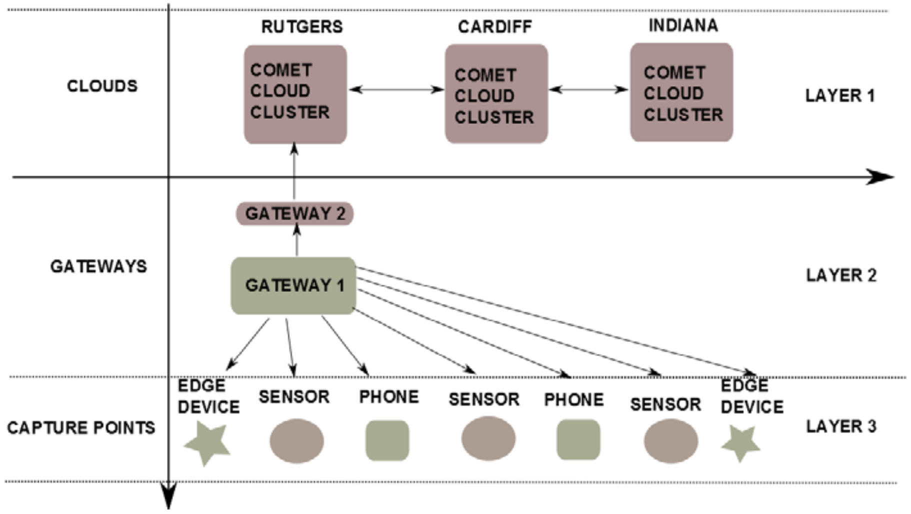
- COSMOS integrates a variety of different services:
  Gender analysis, sentiment analysis, Open Street maps
- Can be integrated with user supplied services

Burnap, P. et al. (2014) '*COSMOS: Towards an Integrated and Scalable Service for Analyzing Social Media on Demand*', International Journal of Parallel, Emergent and Distributed Systems, Taylor & Francis

MODELLING COORDINATION IN MULTI-DIMENSIONAL PIPELINES

Common Theme: Pipelines

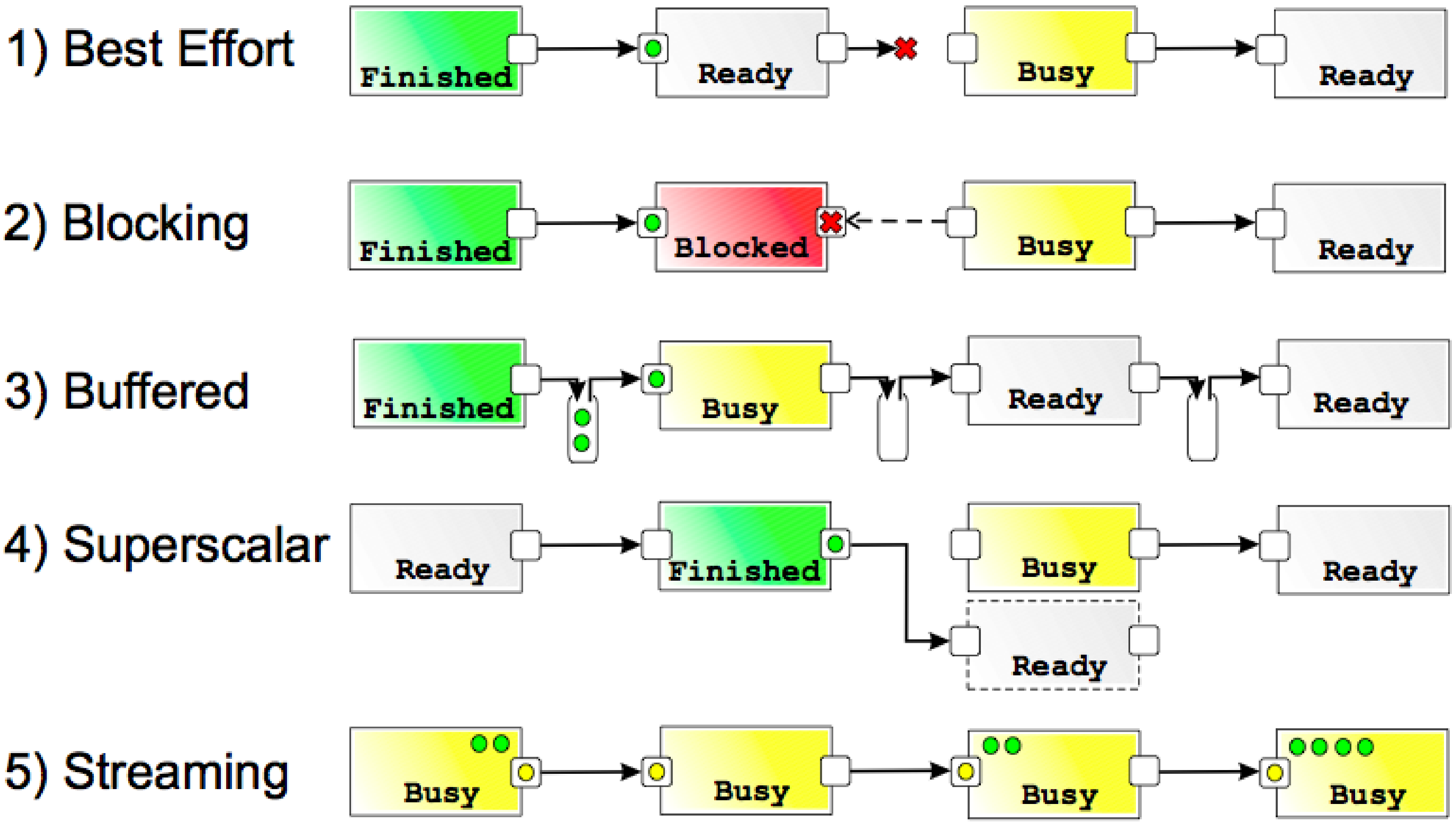
- **Existence of “pipelines” – across multiple layers**
  - Stream/In-Memory analysis
- **Pipeline stages have different emphasis**
  - Pre-Collect and store, data reduction, partial analysis, etc
- **Data-driven pipeline execution**
  - Inclusion of “sensing” into the pipeline
- **Multiple, co-existing, concurrent pipelines**
  - Superscalar pipelines
- **Pipeline capability differs depending on Layer 1, 2 or 3**
  - Resource availability & constraints



Abstractions: Data flow “process networks” (actors and firing rules); Coordination: Pub-Sub + Events, Tuple Space models; Implementations: Yahoo Pipes!, Storm (bolts and spouts, stream groups and topology), Pachube/Xively Cloud (Rate limited); Functional approaches: SCALA;Streamflow and Xbaya; Databases: EVE, Dequob, Calder; SummingBird (used with Storm and Scalding). Commercial: Amazon Kinesis/Lambda; Samza, Cascading, S4, Spark Cluster/Streaming, Google DataFlow/Millwheel; In Memory: Druid, VoltDB, MemSQL, NuoDB

Pipelined Execution Semantics

Based on the notion of a “pipeline collision”



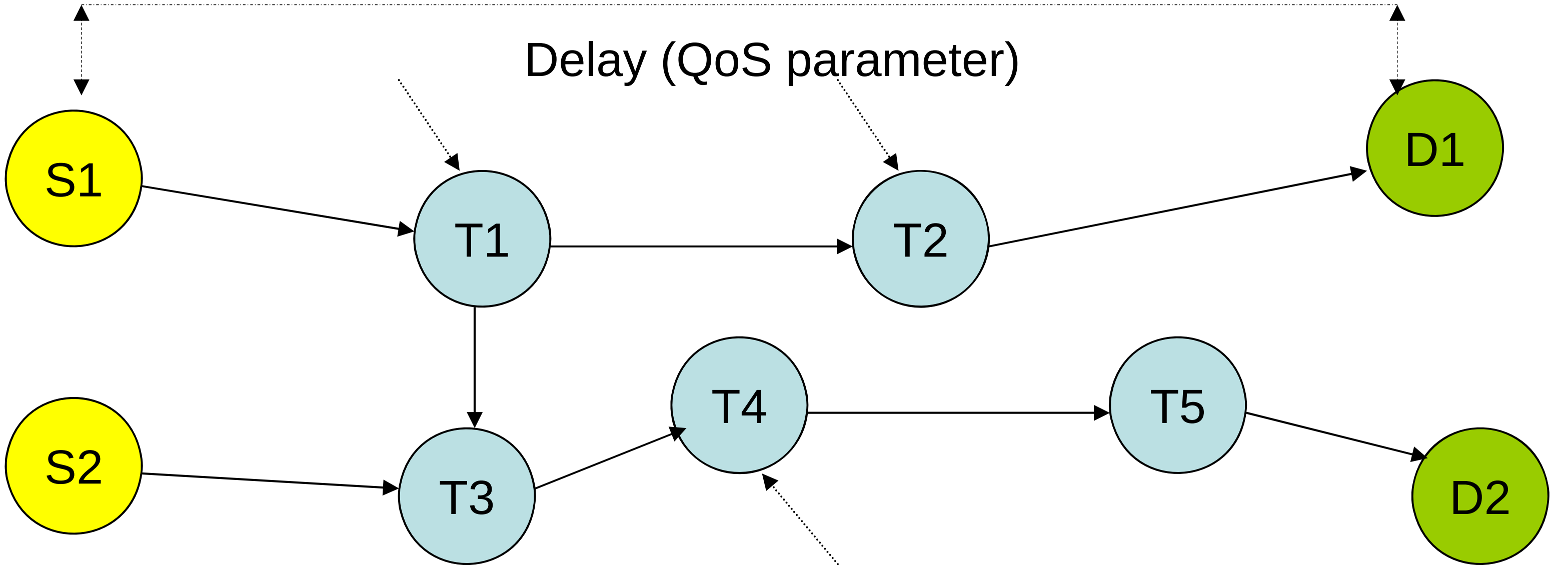
“Parallel Computing Patterns for Grid Workflows”

Cesare Pautasso, Gustavo Alonso, Proceedings of WORKS 2006, alongside SC 2006

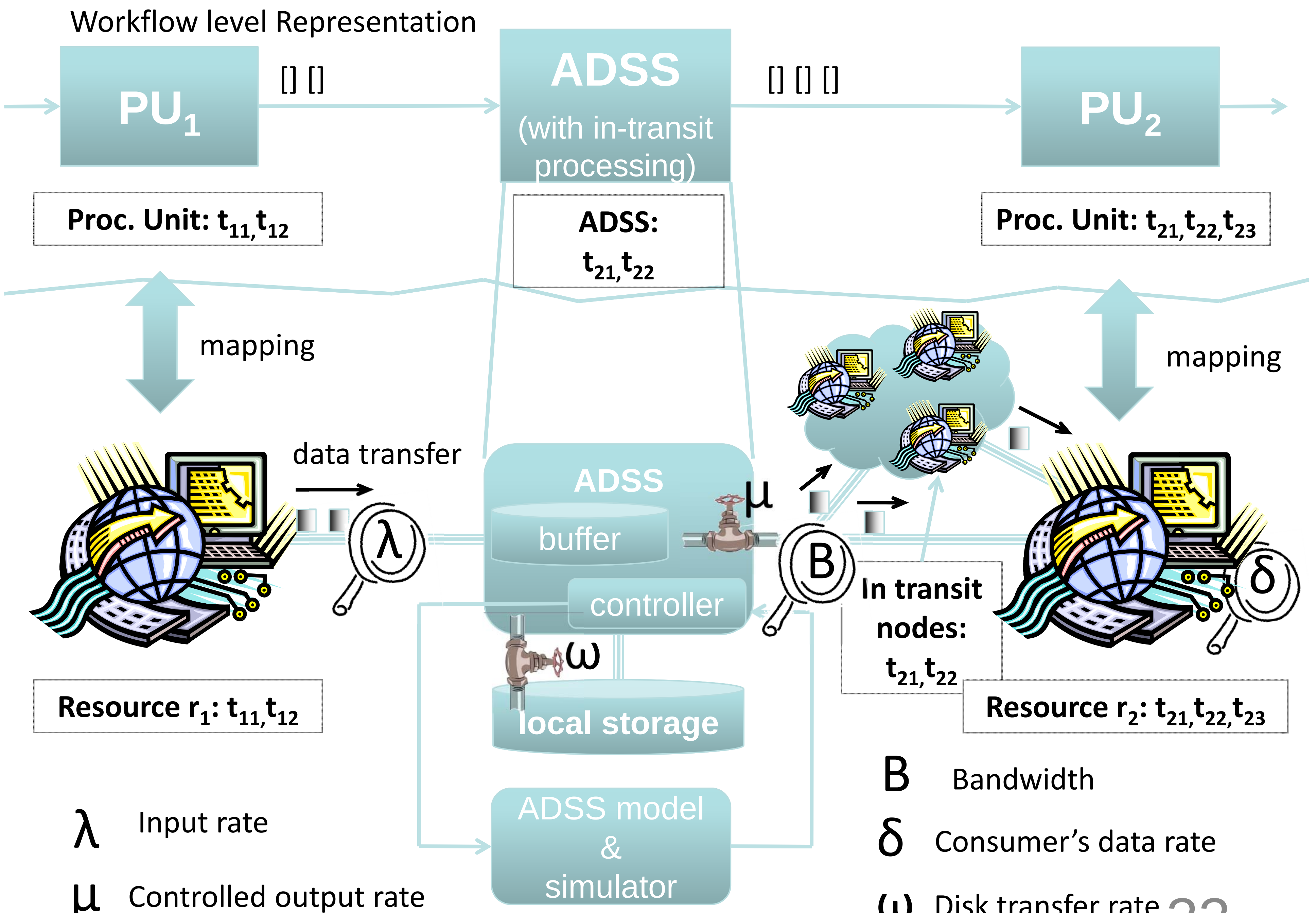
Autonomic Streaming Pipeline

- **Streaming pipeline**
  - No “blocking” semantics
  - Continuous data transmission as a stream
  - Data processing order: arrival order (implicit) or time stamp (explicit)
  - After processing – result elements form the stream
- **Autonomic streaming**
  - Data stream “reacts” to changes in (operating) environment and producer/consumer data generation/consumption rate mismatch
  - Network congestion → alter transmission data rate
  - Alternative modes of analysis: in-transit, at-source, at-sink, etc

In-transit Analysis



- Data processing while data is in movement from source to destination
- Question: what to process where and when
- Use of “slack” in network to support partial processing
- Application types:
  - Streaming & Data Fusion requirement



Workflow level Representation

PU$_1$

[] []

ADSS

(with in-transit processing)

[] [] []

PU$_2$

Proc. Unit: t_{11}, t_{12}

ADSS: t_{21}, t_{22}

Proc. Unit: t_{21}, t_{22}, t_{23}

mapping

mapping

data transfer

λ

ADSS

buffer

controller

μ

ω

B

δ

local storage

ADSS model & simulator

In transit nodes: t_{21}, t_{22}

Resource r_1: t_{11}, t_{12}

Resource r_2: t_{21}, t_{22}, t_{23}

λ	Input rate
μ	Controlled output rate
B	Bandwidth
δ	Consumer’s data rate
ω	Disk transfer rate

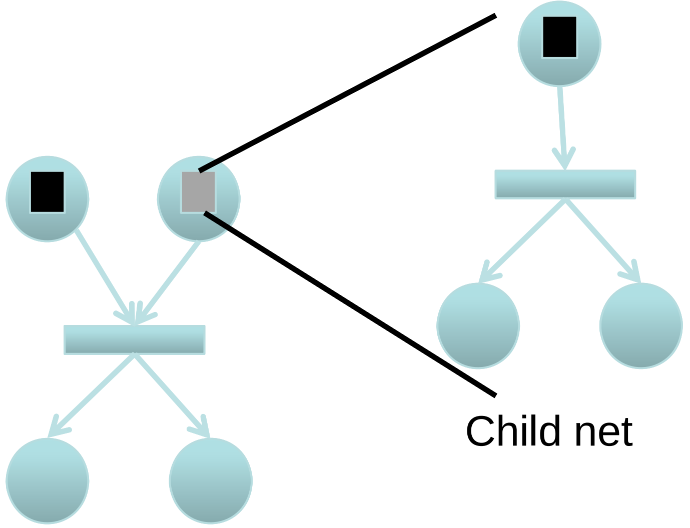
Reference Nets + Renew + DVega

- “Nets-within-nets”
  - Systems net and an object net
  - Net can express creation of new net instances (“creational inscriptions”) – enabling dynamic self-modification of structure
  - Interaction via “synchronous channels”
  - Channel can contain variables whose binding is based on unification
  - Timed reference nets:
    - Time stamp attached to tokens
    - Use of timed inscriptions on arcs (control time stamps and firing delays)
- Renew
  - Java-based interpreter of Reference nets (an executable formalism)
  - Use tuples and Java expression as the inscription language
  - Objects nets can be Java objects

Reference nets

- **Petri nets**
  - directed bipartite graph
  - 2 types of nodes: places and transitions
  - arcs: place-transition, transition-place
  - tokens: move on the graph
  - static structure
- **Reference nets [1]**
  - tokens can be nets ➔ workflow hierarchies
  - tokens can be data ➔ data flow
  - Synchronous channels:
    - synchronise two transitions across different nets which both fire atomically at the same time
    - Both transitions must agree on the name of the channel and on a set of parameters before they can synchronise

- Places have initialisation inscriptions (initial marking)
- Arc inscriptions evaluated on transition firing

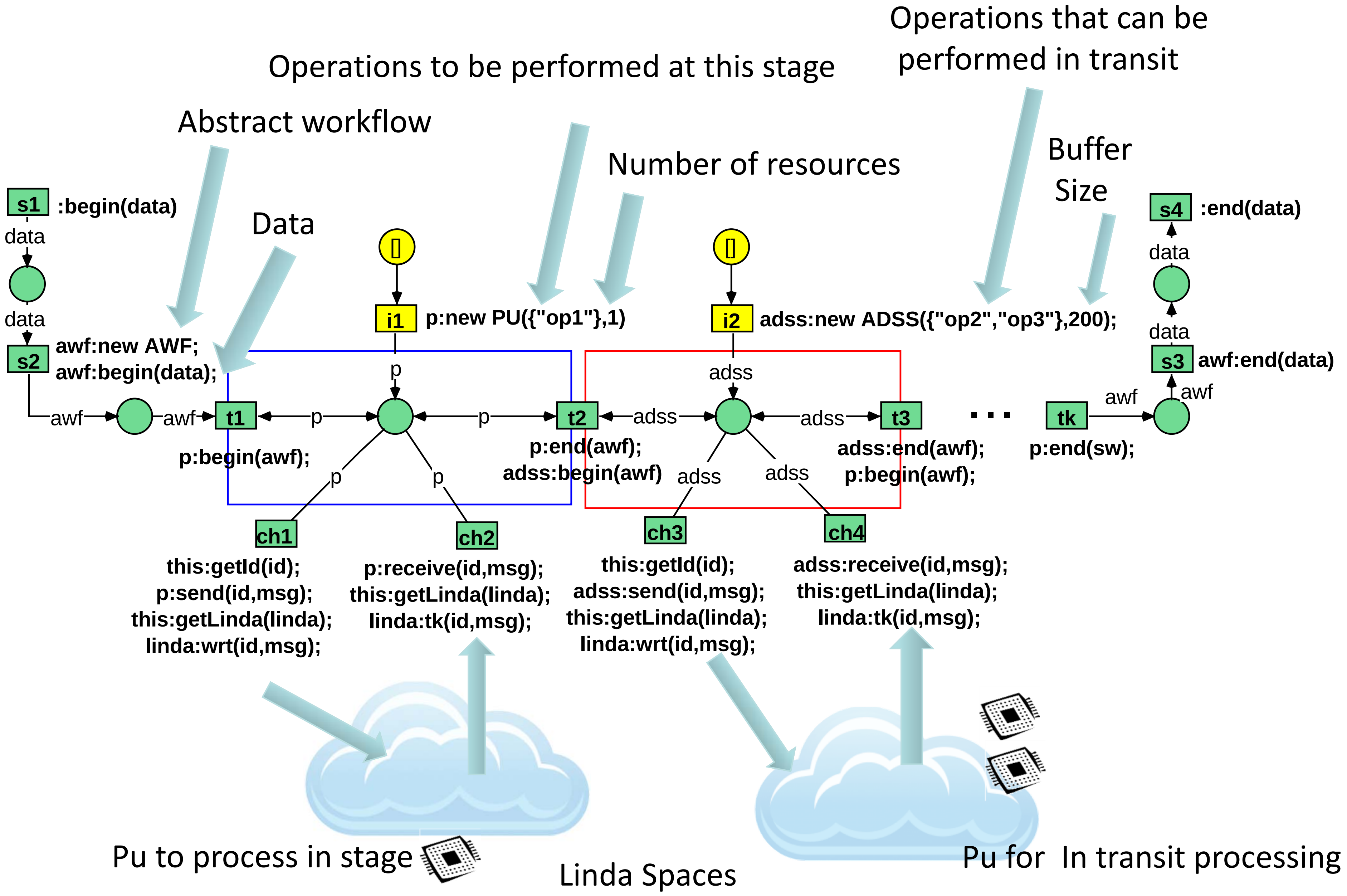


Parent net

---

[1] Referenznetze. Olaf Kummer. Logos Verlag, Berlin 2002

A stage in the pipeline – demonstrating the processing unit and the ADSS



Operations that can be performed in transit

Operations to be performed at this stage

Abstract workflow

Number of resources

Buffer Size

Data

s1 :begin(data)

s2 awf:new AWF; awf:begin(data);

i1 p:new PU({"op1"},1)

i2 adss:new ADSS({"op2","op3"},200);

s4 :end(data)

s3 awf:end(data)

t1 p:begin(awf);

t2 p:end(awf); adss:begin(awf)

t3 adss:end(awf); p:begin(awf);

tk p:end(sw);

ch1

```
this:getId(id);
p:send(id,msg);
this:getLinda(linda);
linda:wrt(id,msg);
```

ch2

```
p:receive(id,msg);
this:getLinda(linda);
linda:tk(id,msg);
```

ch3

```
this:getId(id);
adss:send(id,msg);
this:getLinda(linda);
linda:wrt(id,msg);
```

ch4

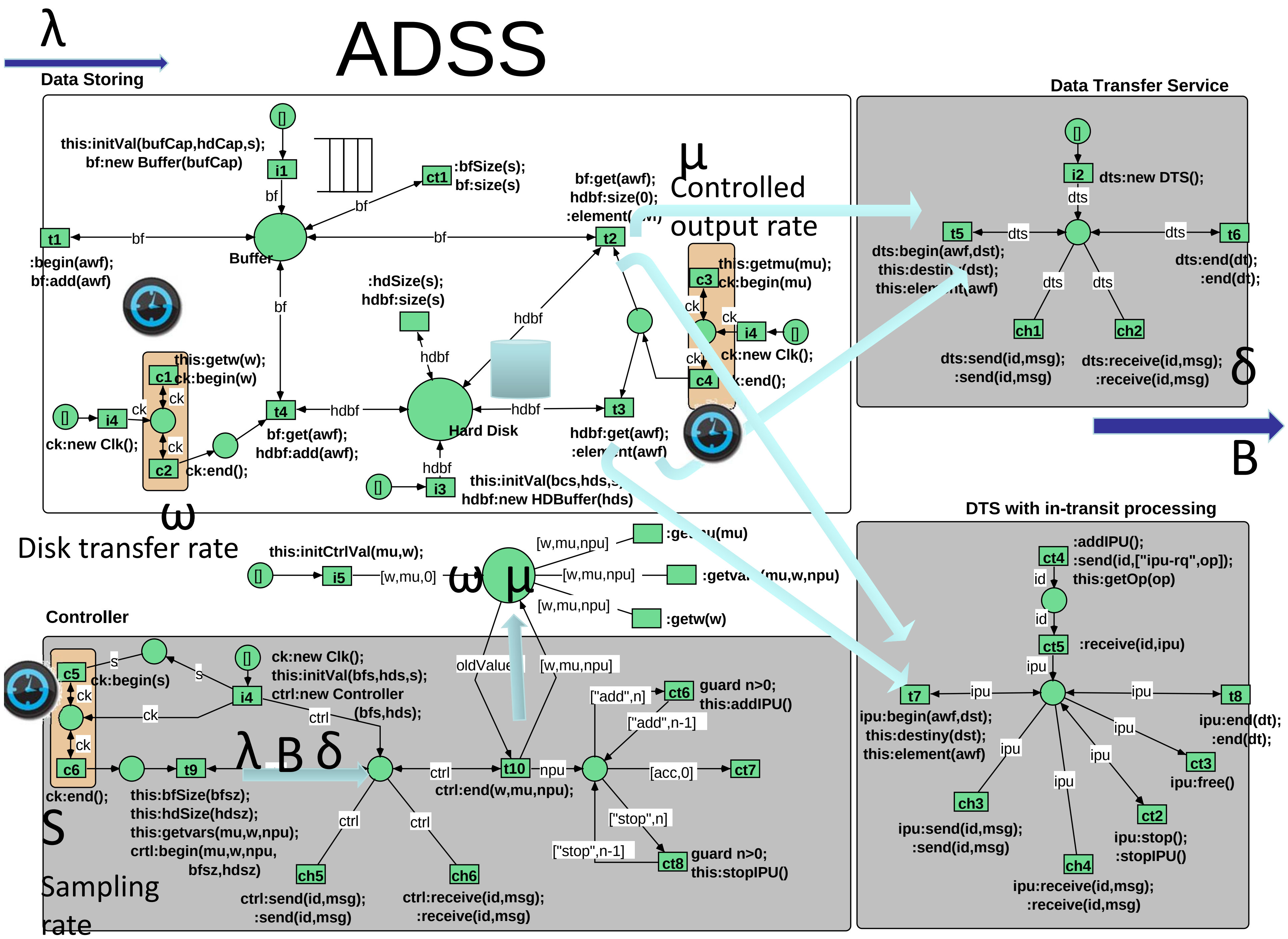
```
adss:receive(id,msg);
this:getLinda(linda);
linda:tk(id,msg);
```

Pu to process in stage

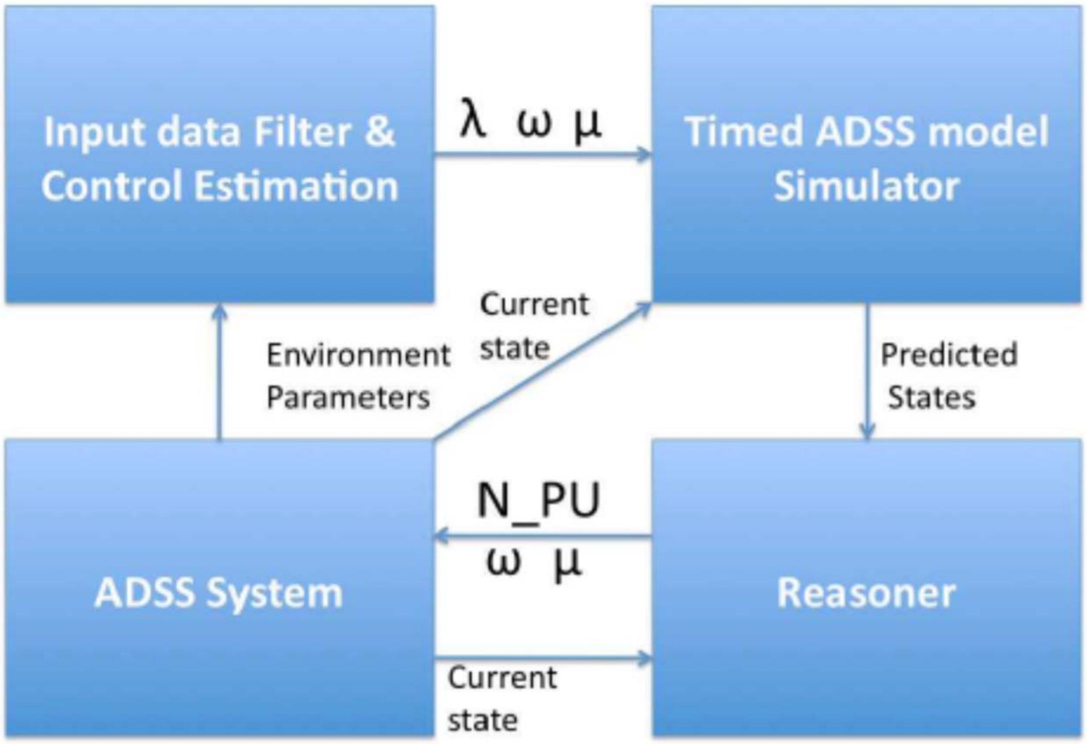
Linda Spaces

Pu for In transit processing

ADSS


λ
Data Storing
this:initVal(bufCap,hdCap,s);
bf:new Buffer(bufCap)
i1
bf
ct1
:bfSize(s);
bf:size(s)
t1
:begin(awf);
bf:add(awf)
Buffer
bf:get(awf);
hdbf:size(0);
:element(awf)
t2
μ
Controlled
output rate
:hdSize(s);
hdbf:size(s)
hdbf
this:getw(w);
ck:begin(w)
c1
ck
i4
ck:new Clk();
c2
ck:end();
t4
bf:get(awf);
hdbf:add(awf);
Hard Disk
t3
hdbf:get(awf);
:element(awf)
i3
this:initVal(bcs,hds,s);
hdbf:new HDBuffer(hds)
c3
this:getmu(mu);
ck:begin(mu)
c4
ck:end();
ω
Disk transfer rate
Data Transfer Service
i2
dts:new DTS();
dts
t5
dts:begin(awf,dst);
this:destiny(dst);
this:element(awf)
t6
dts:end(dt);
:end(dt);
ch1
dts:send(id,msg);
:send(id,msg)
ch2
dts:receive(id,msg);
:receive(id,msg)
δ
B
DTS with in-transit processing
ct4
:addIPU();
:send(id,["ipu-rq",op]);
this:getOp(op)
id
ct5
:receive(id,ipu)
ipu
t7
ipu:begin(awf,dst);
this:destiny(dst);
this:element(awf)
t8
ipu:end(dt);
:end(dt);
ct3
ipu:free()
ch3
ipu:send(id,msg);
:send(id,msg)
ct2
ipu:stop();
:stopIPU()
ch4
ipu:receive(id,msg);
:receive(id,msg)
this:initCtrlVal(mu,w);
i5
[w,mu,0]
ω μ
[w,mu,npu]
:getmu(mu)
:getvars(mu,w,npu)
:getw(w)
Controller
c5
s
ck:begin(s)
i4
ck:new Clk();
this:initVal(bfs,hds,s);
ctrl:new Controller
(bfs,hds);
ctrl
c6
ck:end();
t9
this:bfSize(bfsz);
this:hdSize(hdsz);
this:getvars(mu,w,npu);
crtl:begin(mu,w,npu,
bfsz,hdsz)
λ B δ
oldValue
t10
npu
ctrl:end(w,mu,npu);
["add",n]
ct6
guard n>0;
this:addIPU()
["add",n-1]
[acc,0]
ct7
["stop",n]
["stop",n-1]
ct8
guard n>0;
this:stopIPU()
ch5
ctrl:send(id,msg);
:send(id,msg)
ch6
ctrl:receive(id,msg);
:receive(id,msg)
S
Sampling
rate

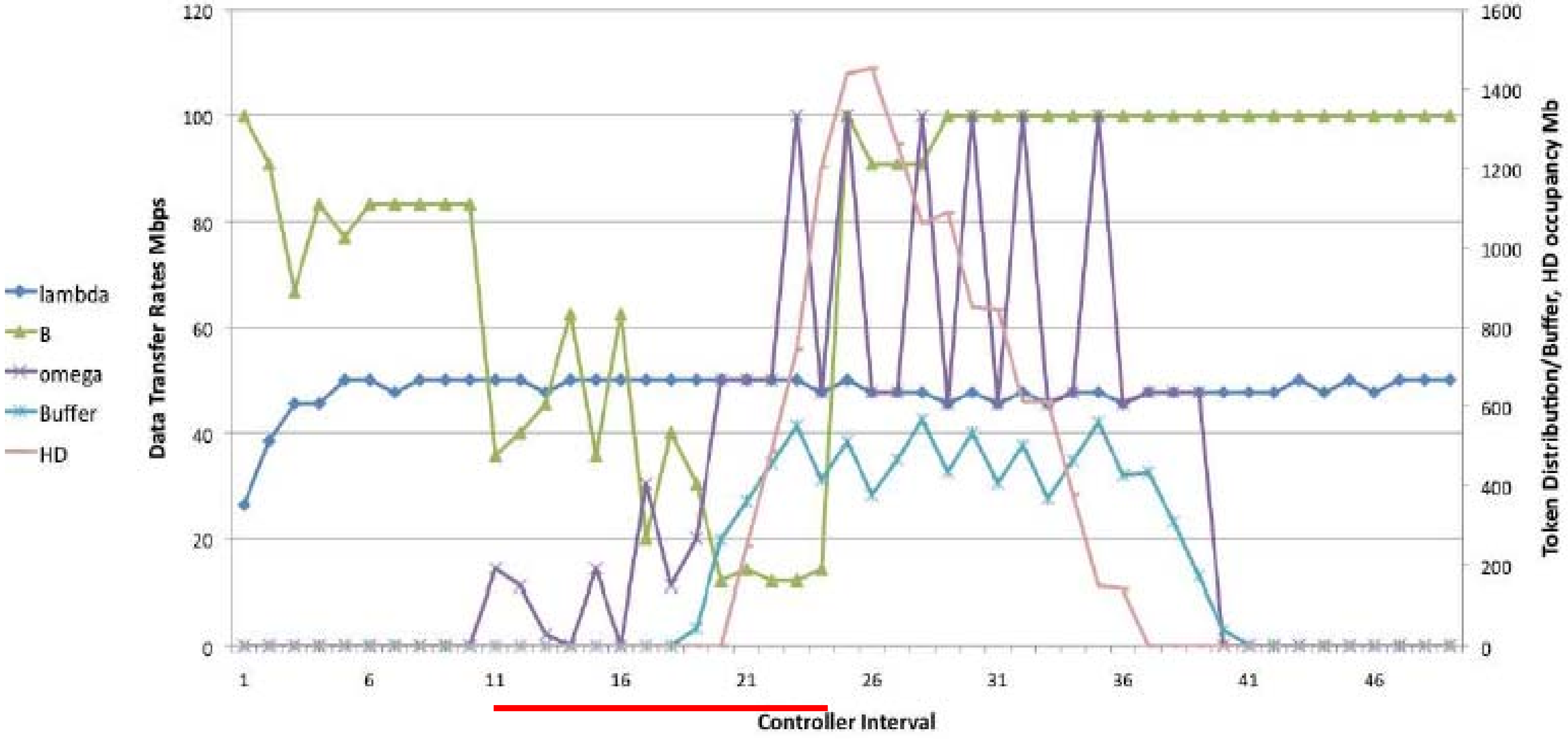
Control Strategy + Adaptation



- Reference net model executes alongside real system
- Model used to tune behaviour
- Rule-based Reasoner coupled with other machine learning strategies

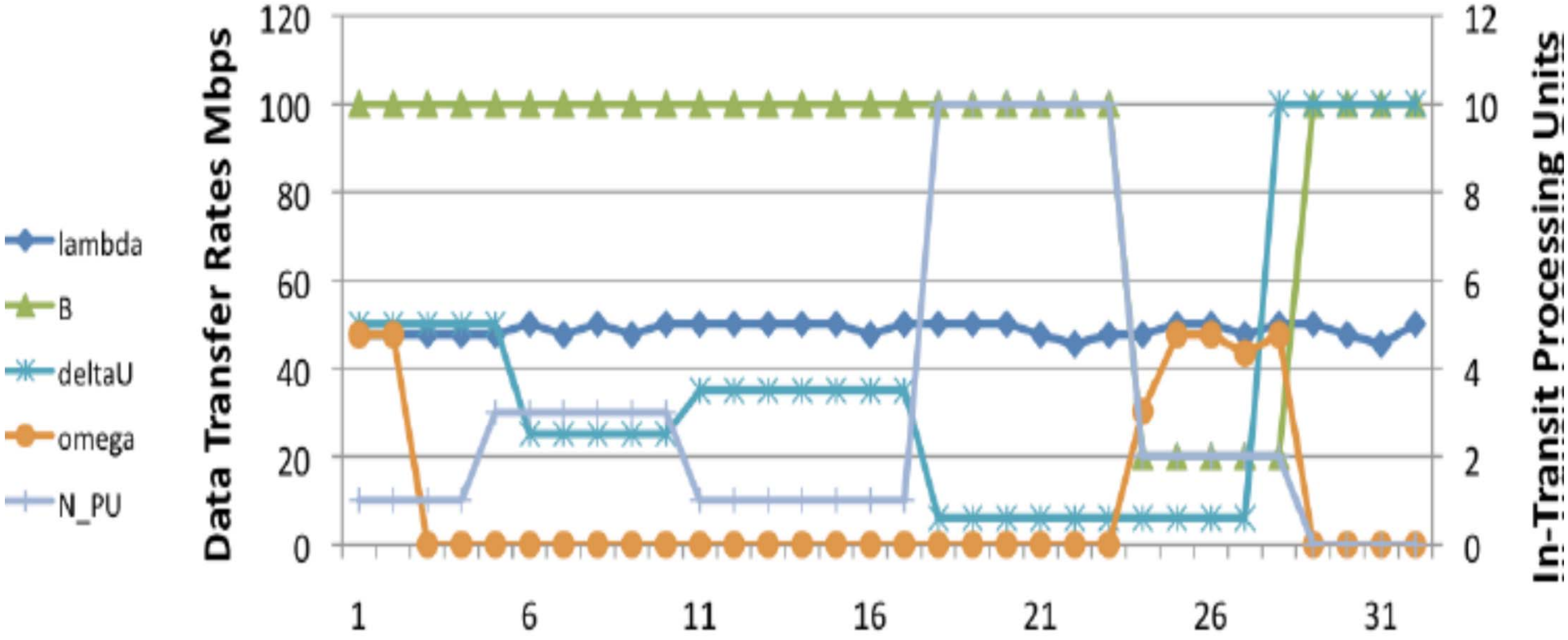
Rafael Tolosana-Calasanz, José A. Bañares, Omer F. Rana:
Autonomic streaming pipeline for scientific workflows.
Concurrency and Computation: Practice and Experience 23 (16): 1868-1892 (2011)

Adapting Transfer Rates based on Network Congestion



Lamda: data generation rate; B: bandwidth; omega: hard disk transfer rate
Network congestion added: intervals 11-24; control interval: 10 secs.

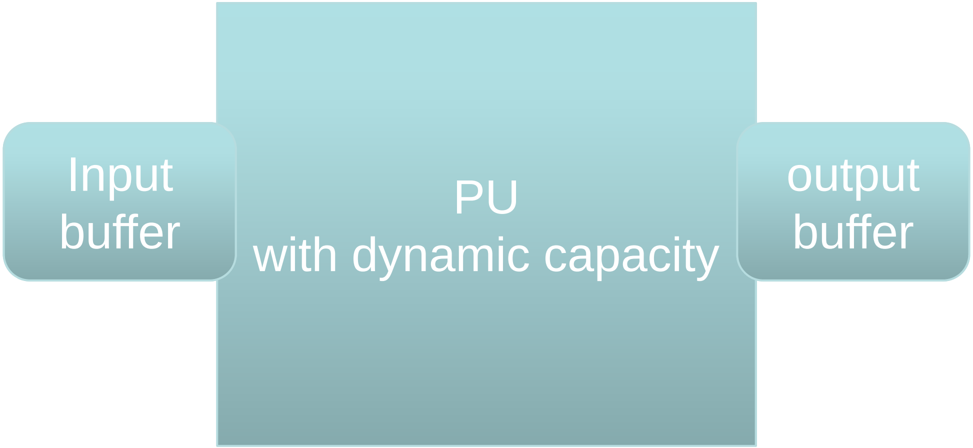
Adding in-transit processing nodes



deltaU: change in processing rate (i.e. number of data items processed/time)

Rafael Tolosana-Calasanz, José Á. Bañares, Omer Rana, Congduc Pham, Erotokritos Xydas, Charalampos Marmaras, Panagiotis Papadopoulos and Liana Cipcigan, "Enforcing QoS on OpenNebula-based Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams", to be presented at DPMSS workshop (from Sensor Networks to Clouds), at 14th IEEE/ACM Int. Symp. On Cluster, Cloud and Grid Computing (CCGrid), Chicago, May 2014.

Superscalar Pipelines and Rate Adaptation



Input
buffer

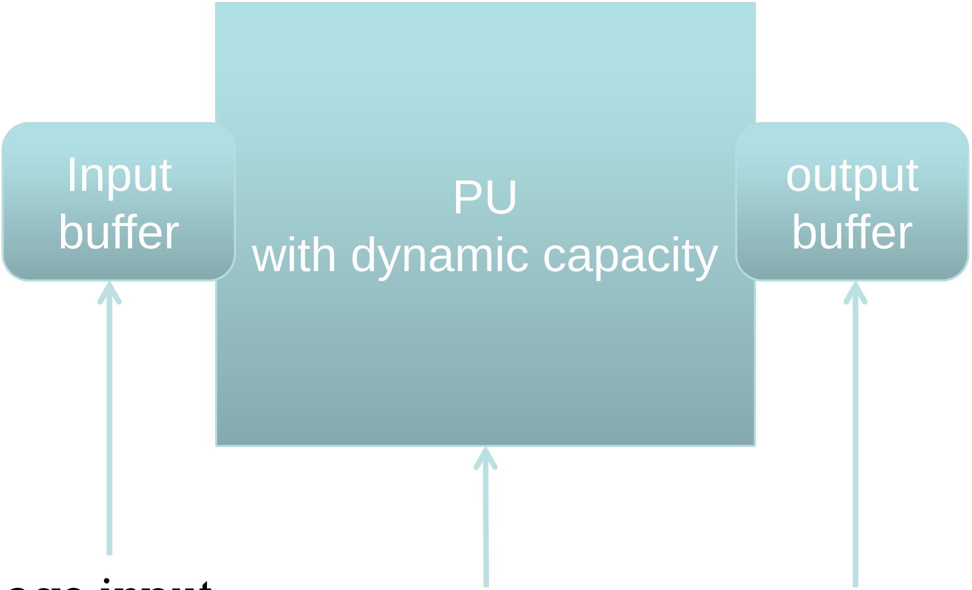
PU
with dynamic capacity

output
buffer

R. Tolosana-Calasanz, J. B. Banares. C. Pham and O. Rana
“Enforcing QoS in Scientific Workflow Systems Enacted Over Cloud Infrastructures”
Journal of Computer and System Science (Elsevier), 2012

Isolating multiple concurrent pipelines

Multiple input streams with different QoS demands



Input buffer

PU
with dynamic capacity

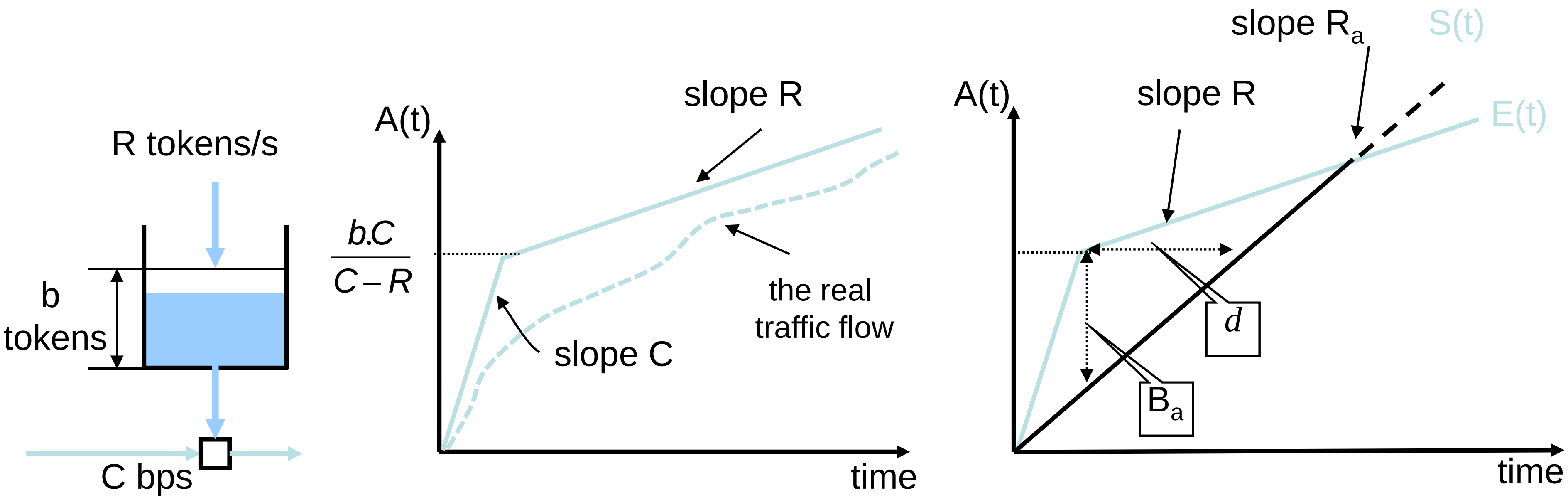
output buffer

Manage input
to guarantee
QoS

Manage local
resources of
PU to guarantee
QoS

Modify mu, omega,
select routes,
add in-transit proc.

Token Bucket for traffic shaping

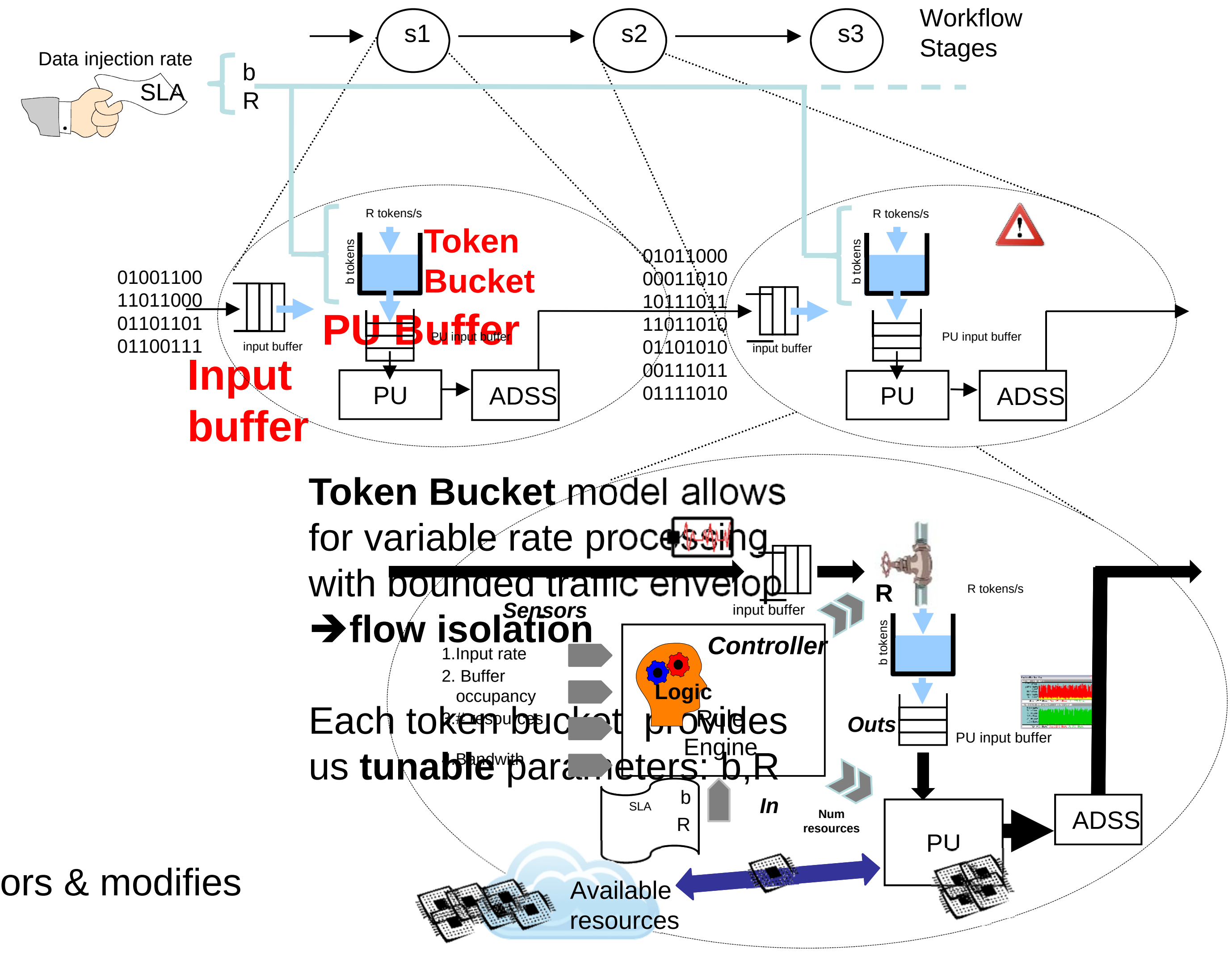


A(t): Amount of data arriving up to time t

- Two key parameters of interest:
  - R: Also called the **committed information rate** (CIR), it specifies how much data can be sent or forwarded per unit time on average
  - b: it specifies for **each burst** how much data can be sent within a given time without creating scheduling concerns. Tokens in excess are normally dropped.

R. Tolosana, J. Banares, C. Pham, O. Rana, "Enforcing QoS in Scientific Workflow Systems Enacted Over Cloud Infrastructures", Journal of Computer and System Science (JCSS), 78(5), Elsevier.

System Architecture

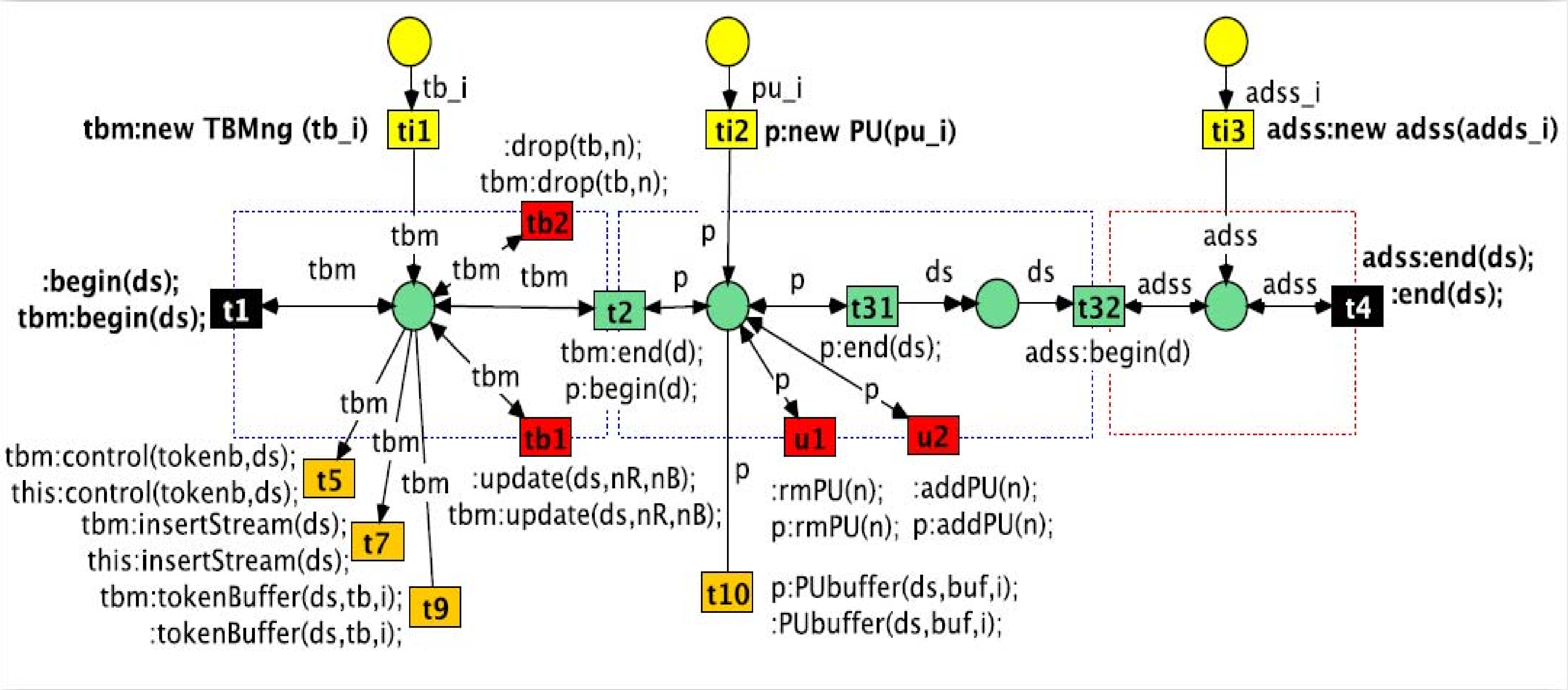


Token Bucket model allows for variable rate processing with bounded traffic envelop
➔**flow isolation**

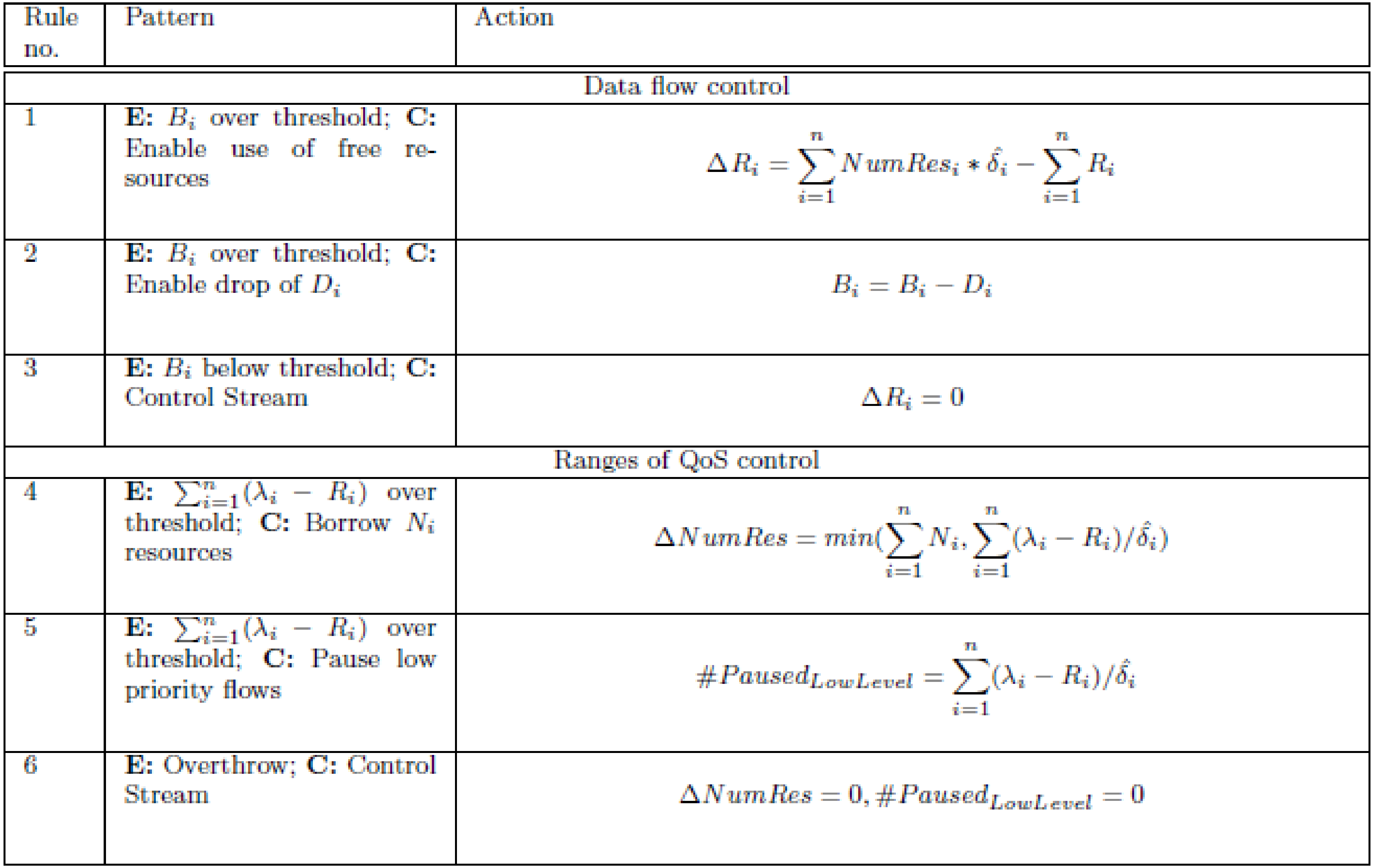
Each token bucket provides us **tunable** parameters: b,R

Controller: monitors & modifies behaviour

Integrating Token Bucket Into Model



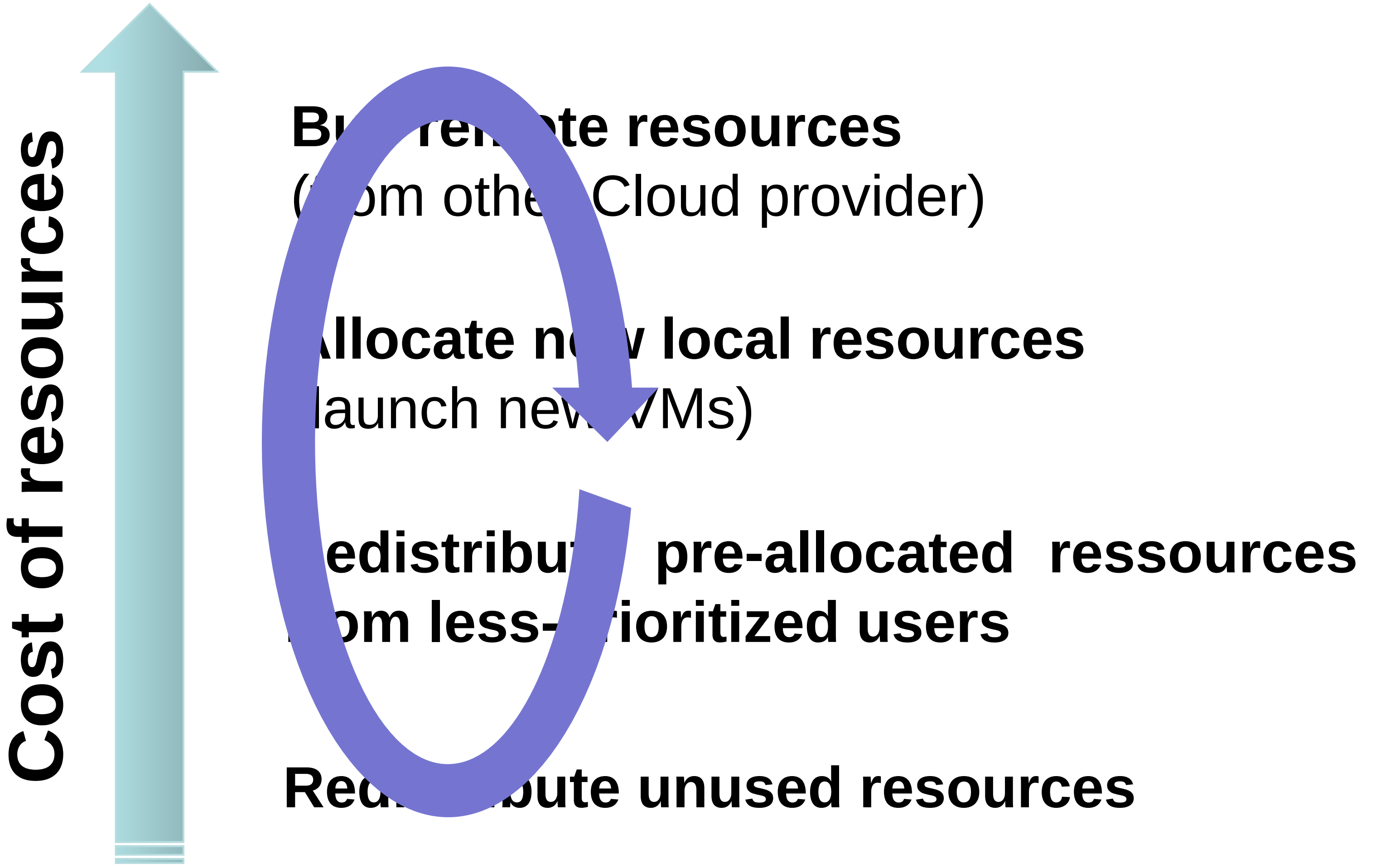
Example Rules



Rule no.	Pattern	Action
Data flow control		
1	**E:** B_i over threshold; **C:** Enable use of free resources	$\Delta R_i = \sum_{i=1}^{n} NumRes_i * \hat{\delta_i} - \sum_{i=1}^{n} R_i$
2	**E:** B_i over threshold; **C:** Enable drop of D_i	$B_i = B_i - D_i$
3	**E:** B_i below threshold; **C:** Control Stream	$\Delta R_i = 0$
Ranges of QoS control		
4	**E:** $\sum_{i=1}^{n}(\lambda_i - R_i)$ over threshold; **C:** Borrow N_i resources	$\Delta NumRes = min(\sum_{i=1}^{n} N_i, \sum_{i=1}^{n}(\lambda_i - R_i)/\hat{\delta_i})$
5	**E:** $\sum_{i=1}^{n}(\lambda_i - R_i)$ over threshold; **C:** Pause low priority flows	$\#Paused_{LowLevel} = \sum_{i=1}^{n}(\lambda_i - R_i)/\hat{\delta_i}$
6	**E:** Overthrow; **C:** Control Stream	$\Delta NumRes = 0, \#Paused_{LowLevel} = 0$

R. Tolosana, J. Banares, C. Pham, O. Rana,
"End-to-End QoS on Shared Clouds for Highly Dynamic, Large-Scale Sensing Data Streams", *Proceedings of the Workshop on Data-intensive Process Management in Large-Scale Sensor Systems (DPMSS 2012): From Sensor Networks to Sensor Clouds. Alongside IEEE/ACM CCGrid 2012, Ottawa, Canada, May 13-16, 2012*

Resource Management Strategy



Cost of resources

Buy remote resources
(from other Cloud provider)

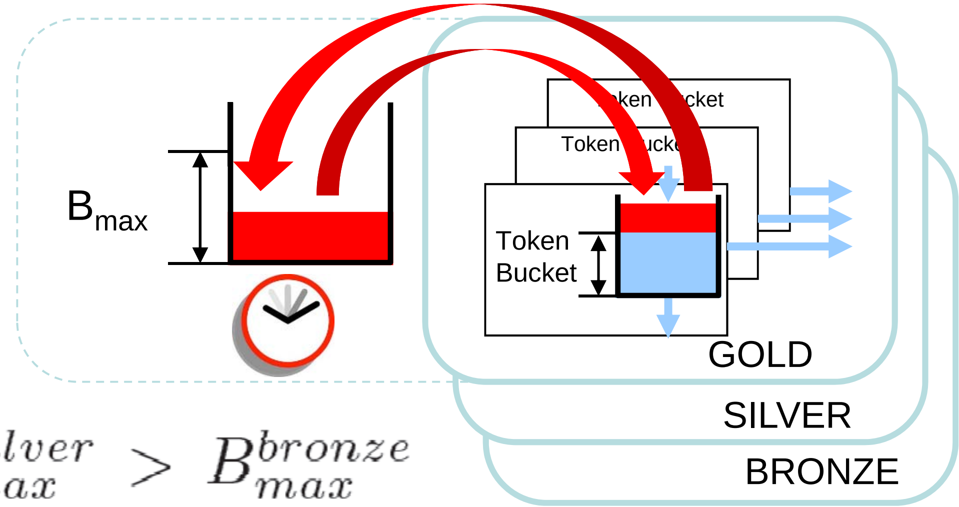
Allocate new local resources
(launch new VMs)

Redistribute pre-allocated ressources from less-prioritized users

Redistribute unused resources

José Ángel Bañares, Omer F. Rana, Rafael Tolosana-Calasanz, Congduc Pham: ***Revenue Creation for Rate Adaptive Stream Management in Multi-tenancy Environments.*** *Int. Conf on Economics of Grids, Clouds, Systems & Services (GECON 2013), pp 122-137, Zaragoza, Spain, Springer*

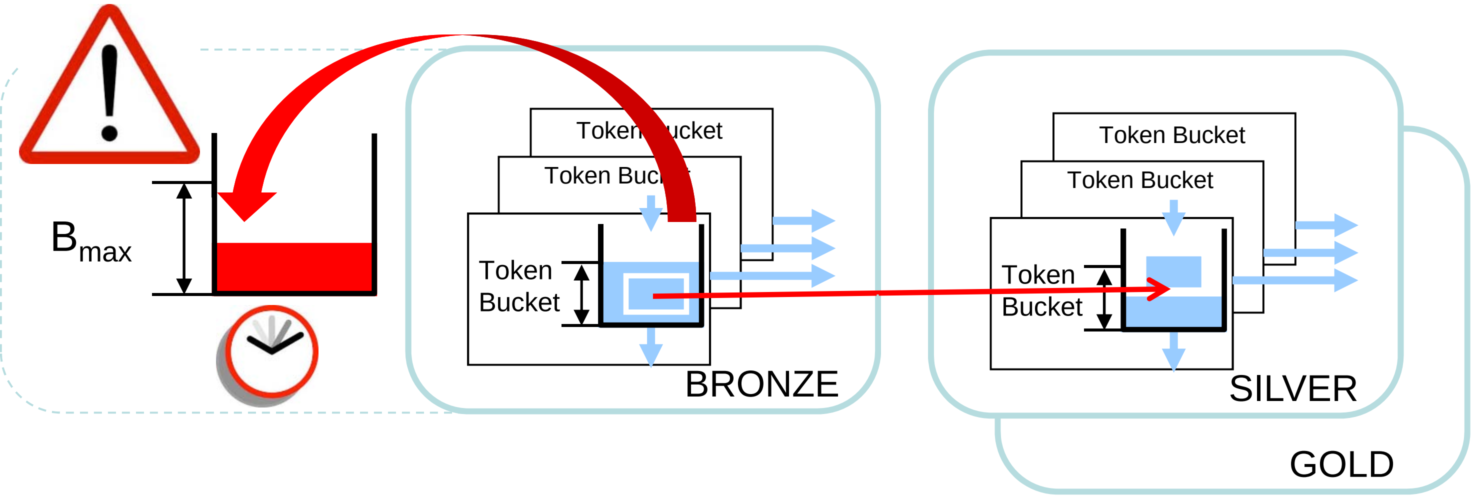
Unified resource mngt with TB
➔Redistribute unused resources



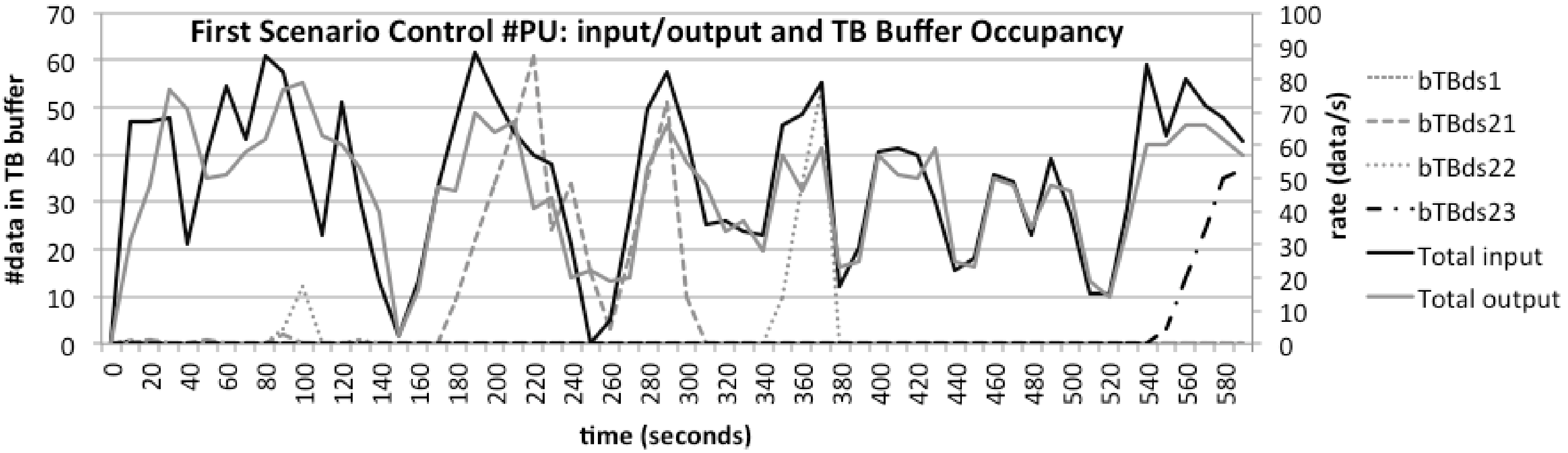
$$B_{max}^{gold} > B_{max}^{silver} > B_{max}^{bronze}$$

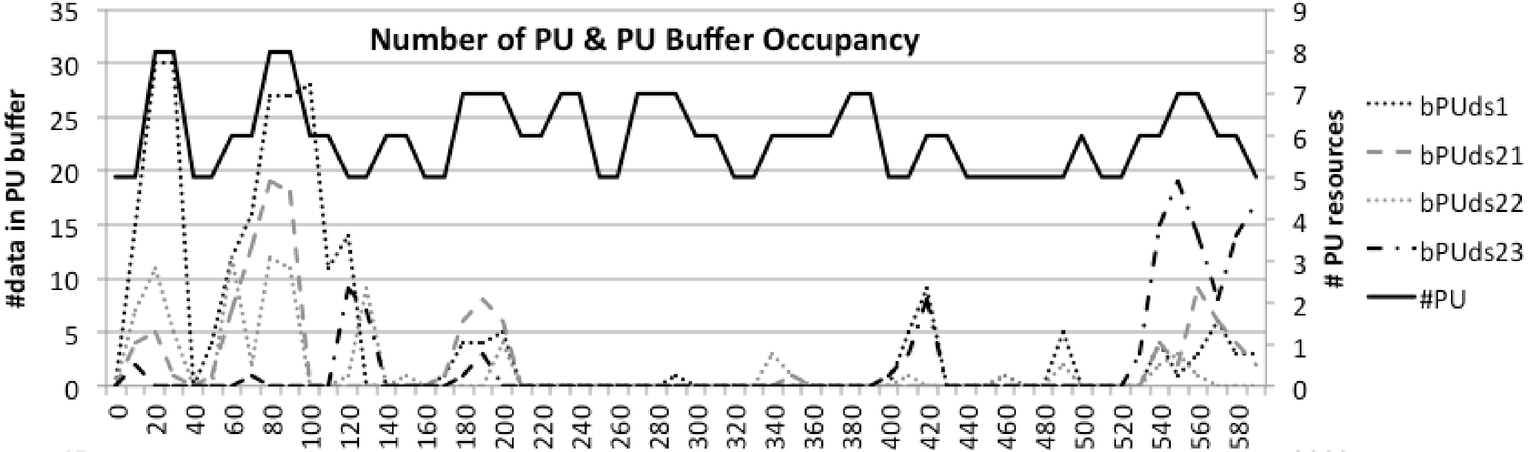
- Under-utilization of resource in a flow will produce tokens in excess
- Within a service class,
  - Tokens in excess of all flows are collected and stored up to B_{max} tokens
  - Token's lifetime is limited to a few control intervals to limit inconsistency

Unified resource mngt with TB
➔Take resources from lower classes

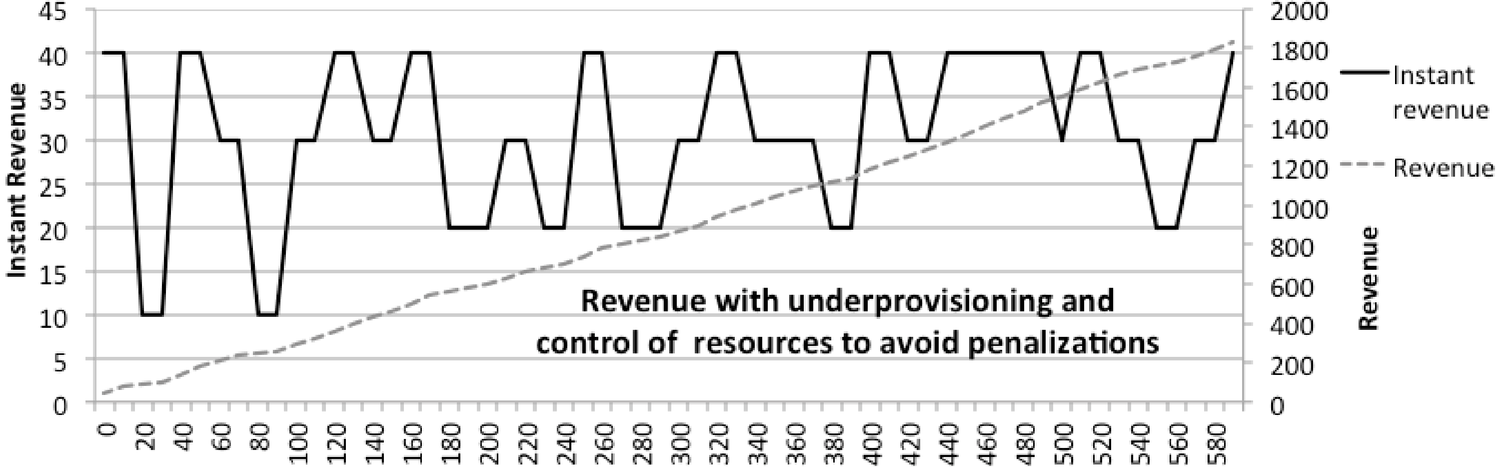


- Silver class has higher revenue and higher penalty than Bronze class for example: shortage of resource in Silver class **is more costly**
- Taking resources from Bronze to Silver is more **revenue-efficient**
  - Tokens are taken directly from a Bronze flow’s token bucket
  - Can put a limit to the number of tokens the system can take
- **Safer** than taking tokens from the Bronze unused token bucket

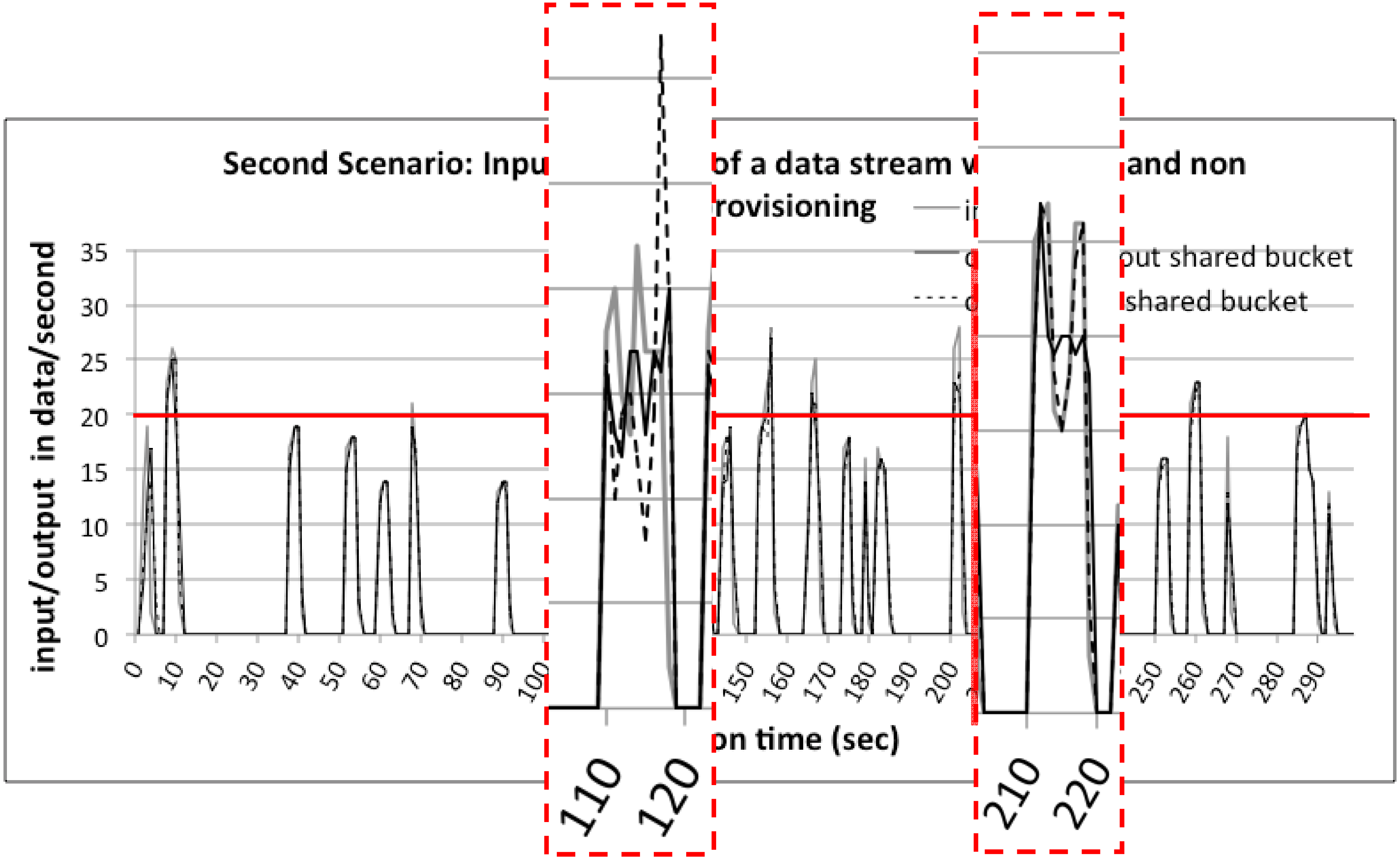




ADDITION
OF
RESOURCES
FOR GOLD
CUSTOMERS

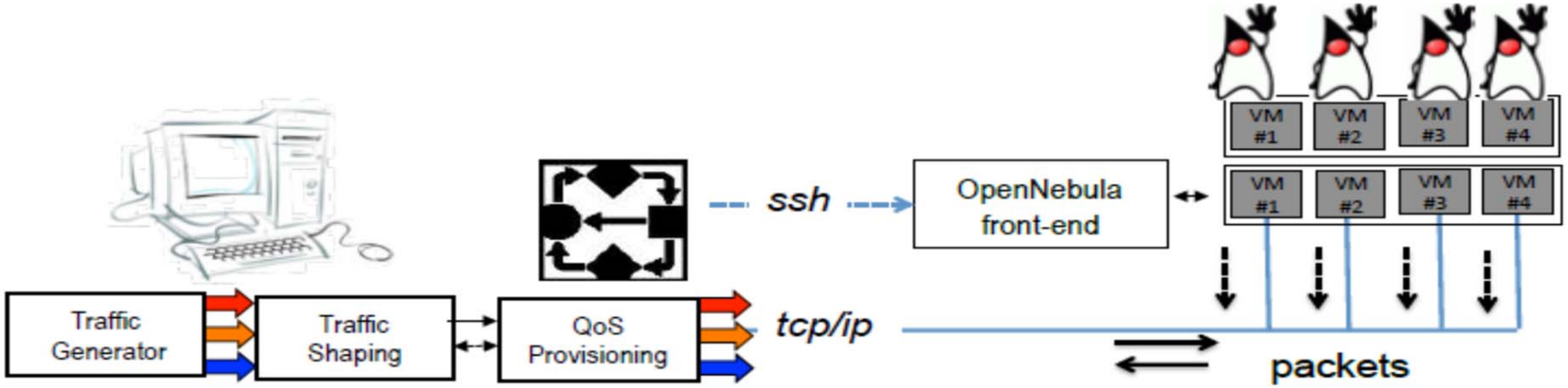


Simulation results – throughput



Integration with OpenNebula

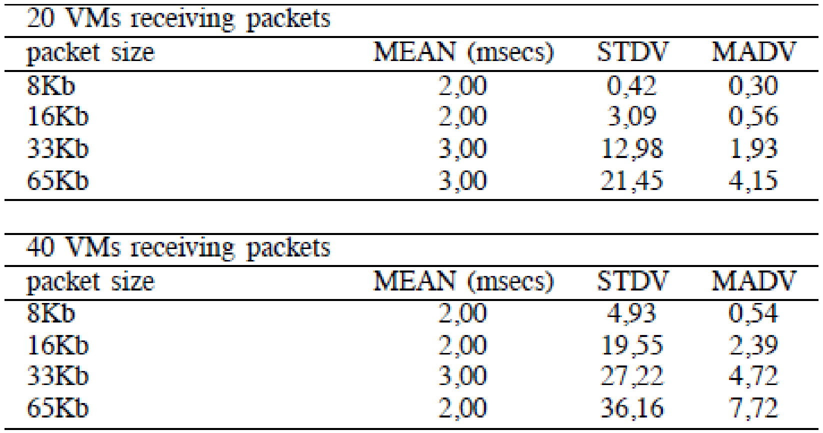
Rafael Tolosana-Calasanz, José Ángel Bañares, Omer Rana, Congduc Pham, Erotokritos Xydas, Charalampos E. Marmaras, Panagiotis Papadopoulos, Liana Cipcigan: "Enforcing Quality of Service on OpenNebula-Based Shared Clouds". IEEE/ACM CCGRID 2014, pp 651-659, Chicago, USA



Traffic shaping achieved through the use of a Token Bucket manager. A token bucket for each data stream

Monitoring number of accumulating packets in an intermediate buffer – triggers creation of new VM instances

OpenNebula 4.4 (32 physical nodes), 32GB/node, 8 cores/node. 4VMs (8VMs) send packets to 20VMs (40VMs) with 5 processes/VM, at 400 packets/s



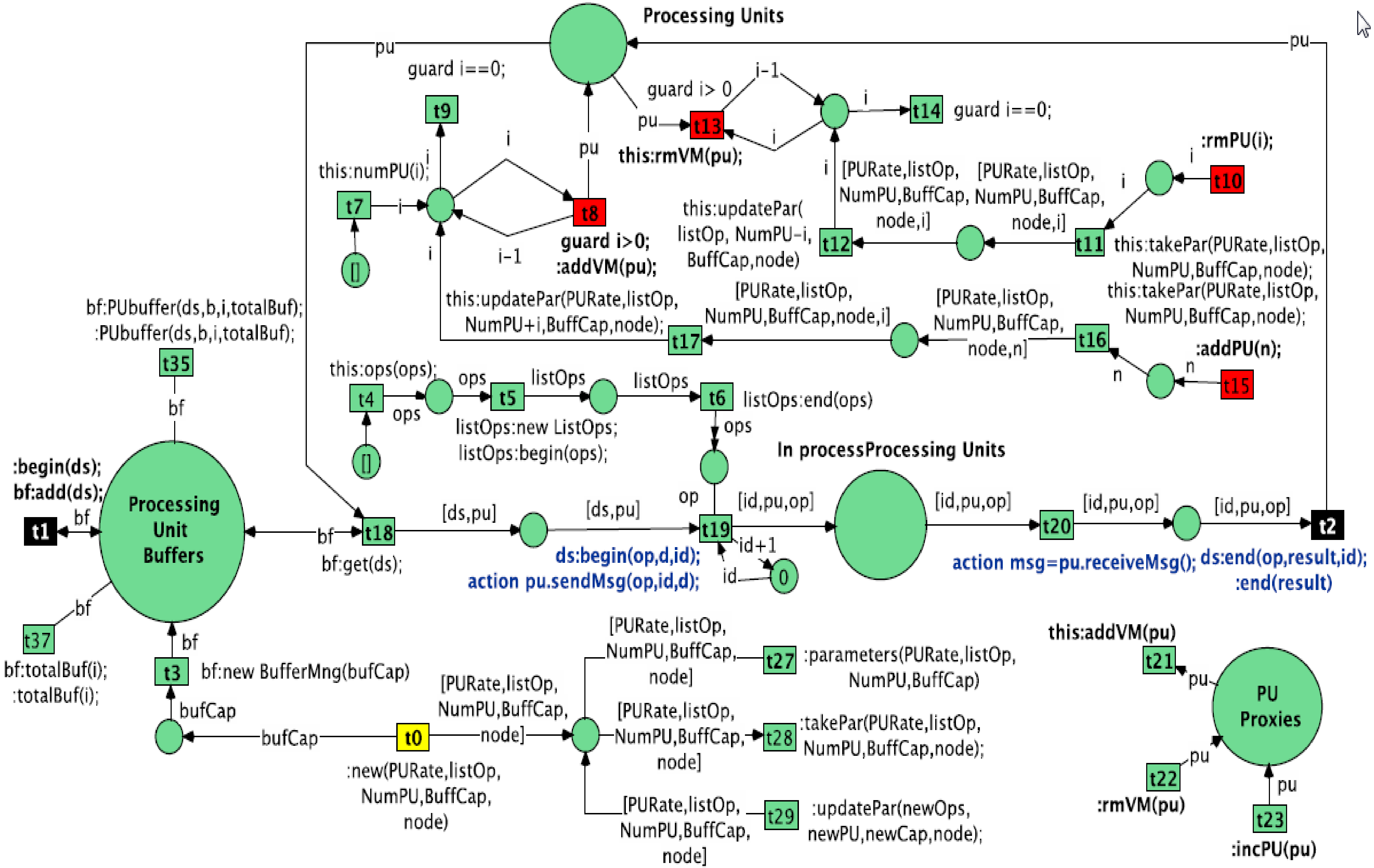
20 VMs receiving packets			
packet size	MEAN (msecs)	STDV	MADV
8Kb	2,00	0,42	0,30
16Kb	2,00	3,09	0,56
33Kb	3,00	12,98	1,93
65Kb	3,00	21,45	4,15

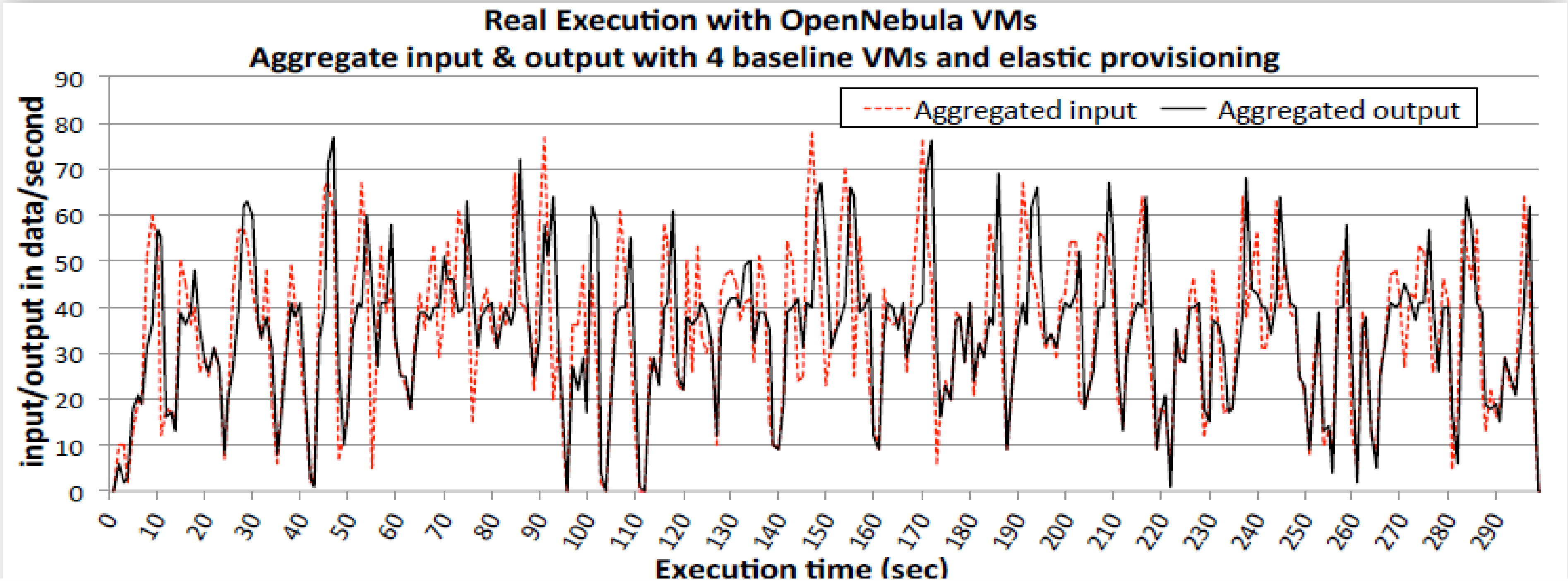
40 VMs receiving packets			
packet size	MEAN (msecs)	STDV	MADV
8Kb	2,00	4,93	0,54
16Kb	2,00	19,55	2,39
33Kb	3,00	27,22	4,72
65Kb	2,00	36,16	7,72

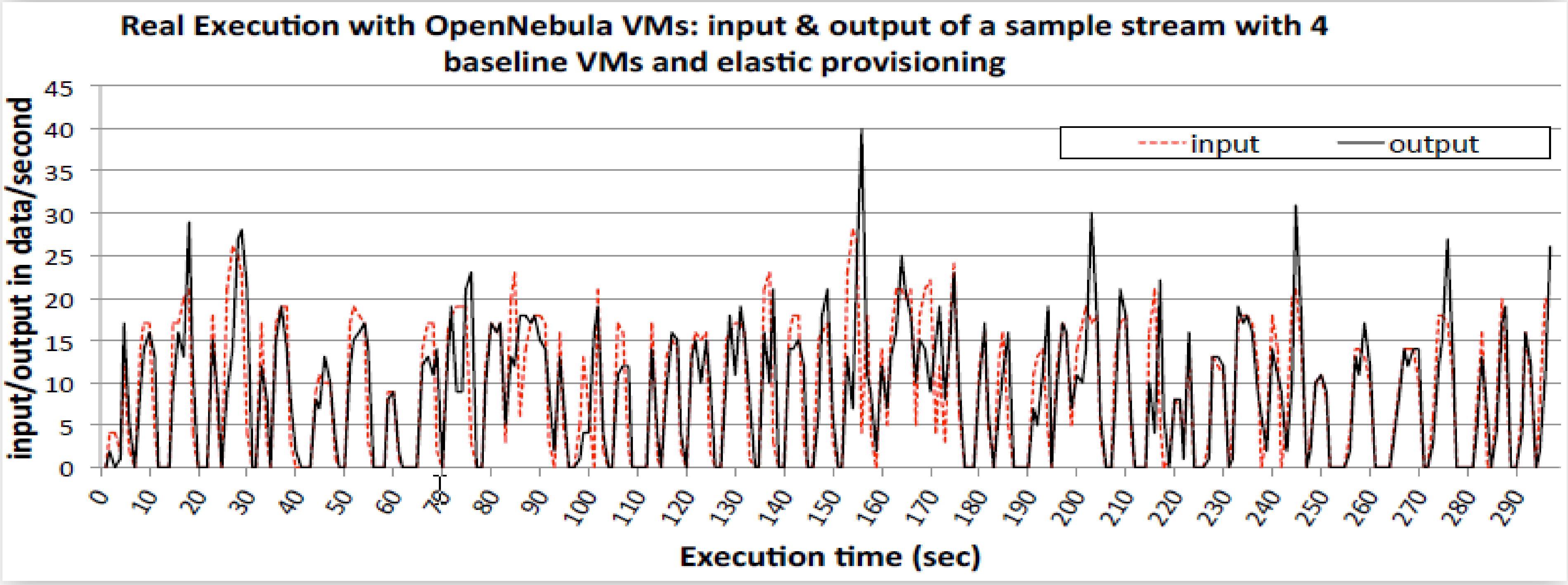
TABLE I
JITTER MEAN, STANDARD DEVIATION AND MEDIAN ABSOLUTE DEVIATION WITH VMS RECEIVING PACKETS AT 400 PACKETS PER SEC

VM cross talk and dynamic VM creation

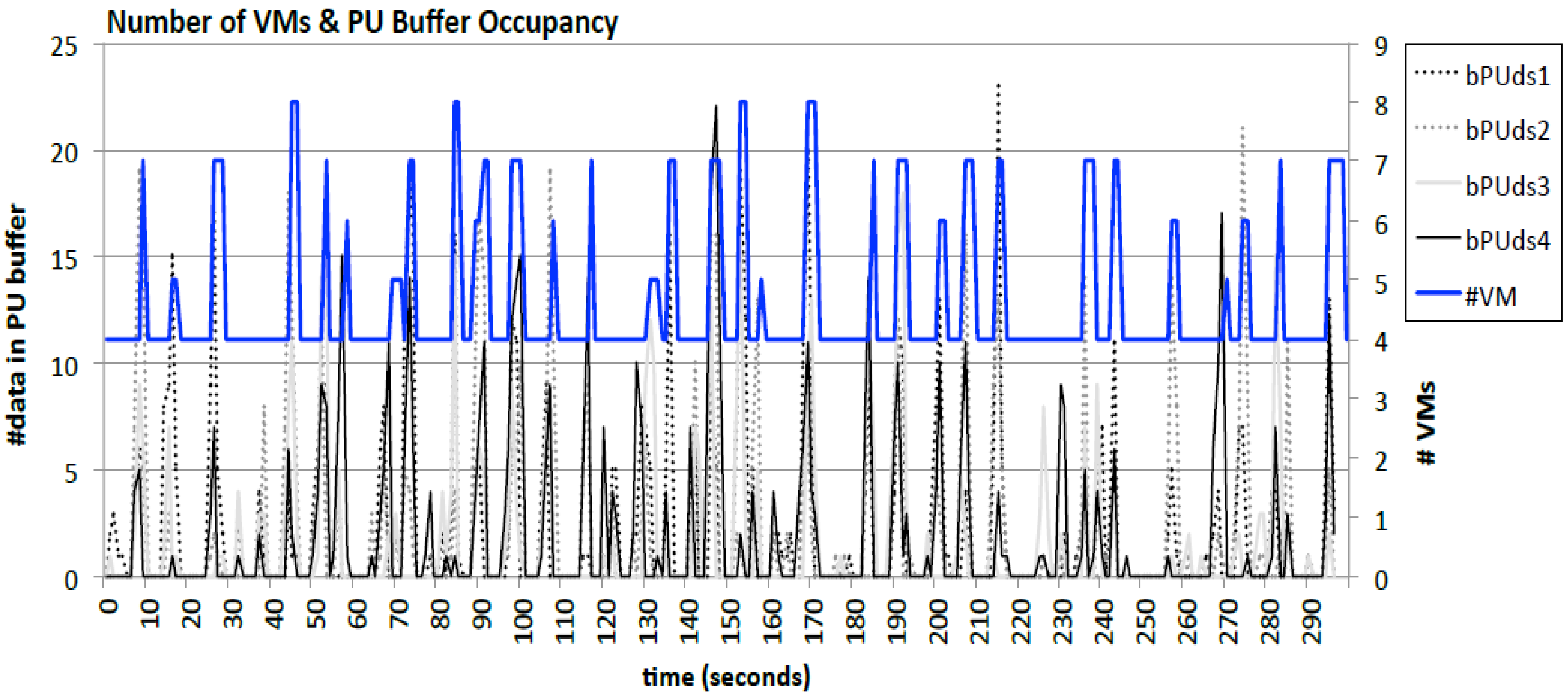
Managing VMs




Real Execution with OpenNebula VMs
Aggregate input & output with 4 baseline VMs and elastic provisioning
Aggregated input
Aggregated output
input/output in data/second
0
10
20
30
40
50
60
70
80
90
Execution time (sec)
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290

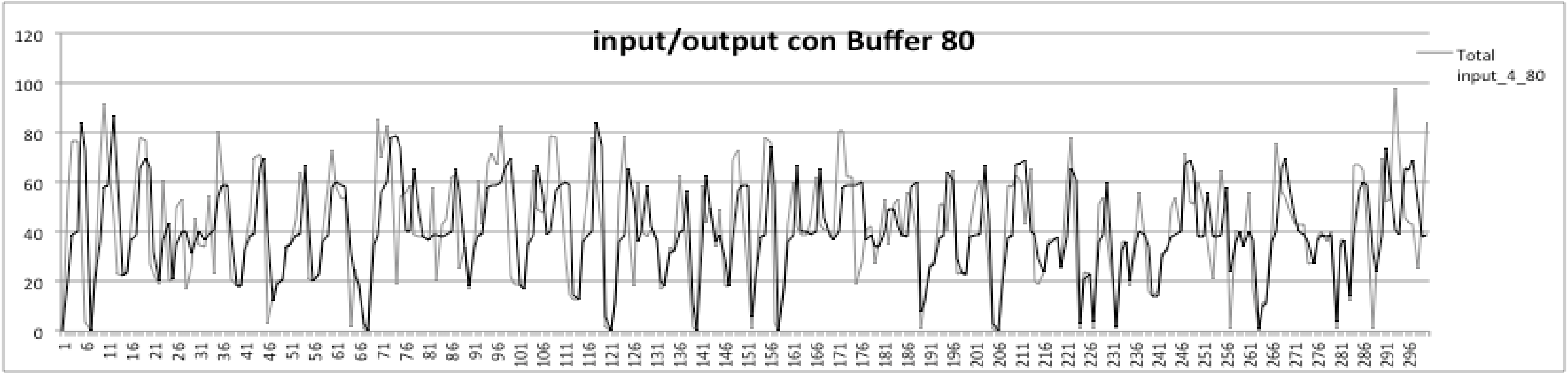

Real Execution with OpenNebula VMs: input & output of a sample stream with 4 baseline VMs and elastic provisioning
input
output
input/output in data/second
0
5
10
15
20
25
30
35
40
45
Execution time (sec)
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150
160
170
180
190
200
210
220
230
240
250
260
270
280
290

Triggering VM launch with data buffer occupancy

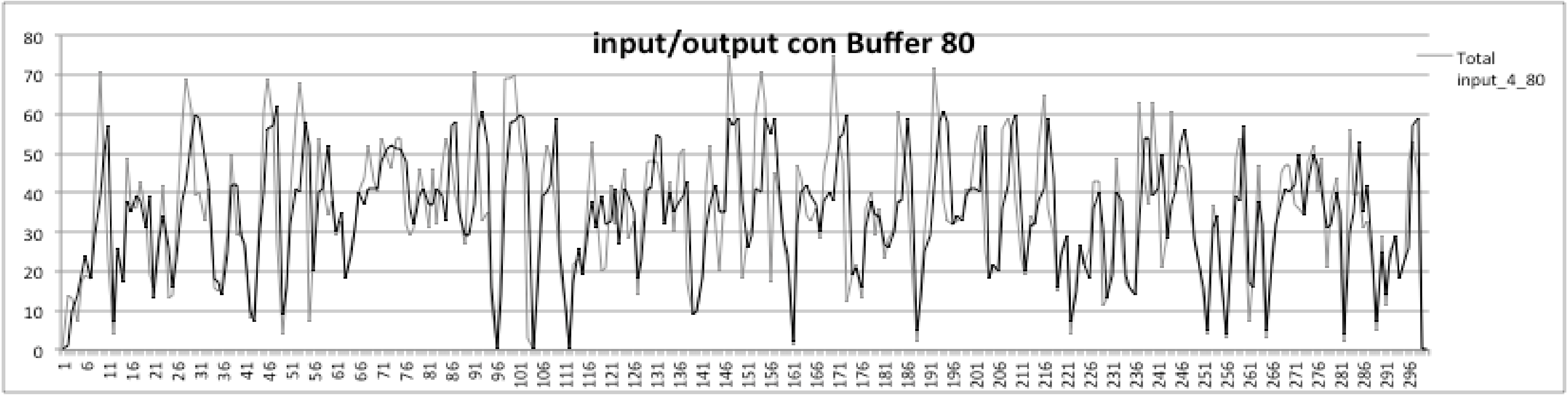


Comparative Simulation/Real VMs

Simulated: Aggregated Traffic



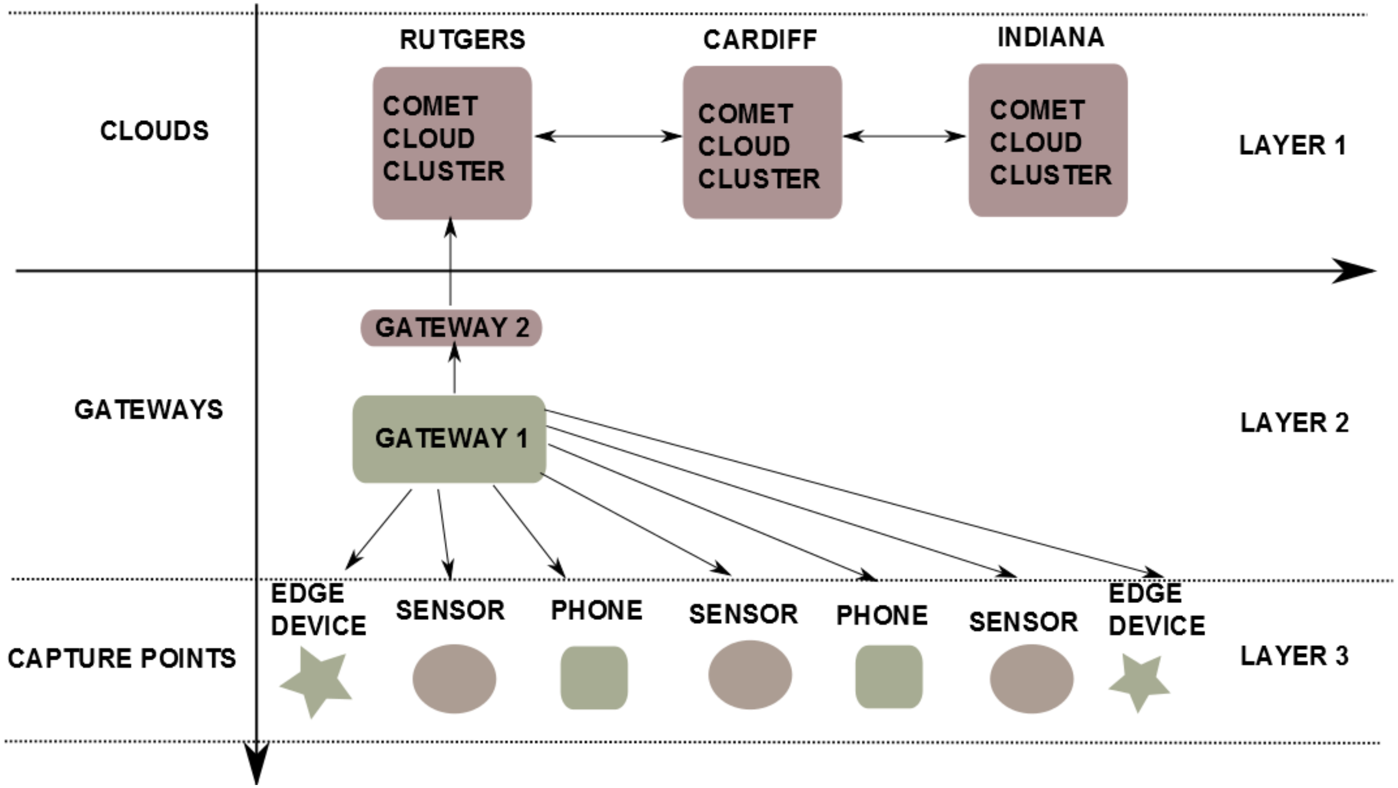
OpenNebula: Aggregated Traffic



Building Energy Simulation – Concluding Scenario

MULTI-LAYERED FEDERATED CLOUDS

CometCloud-based Multi-Layered Federated Clouds

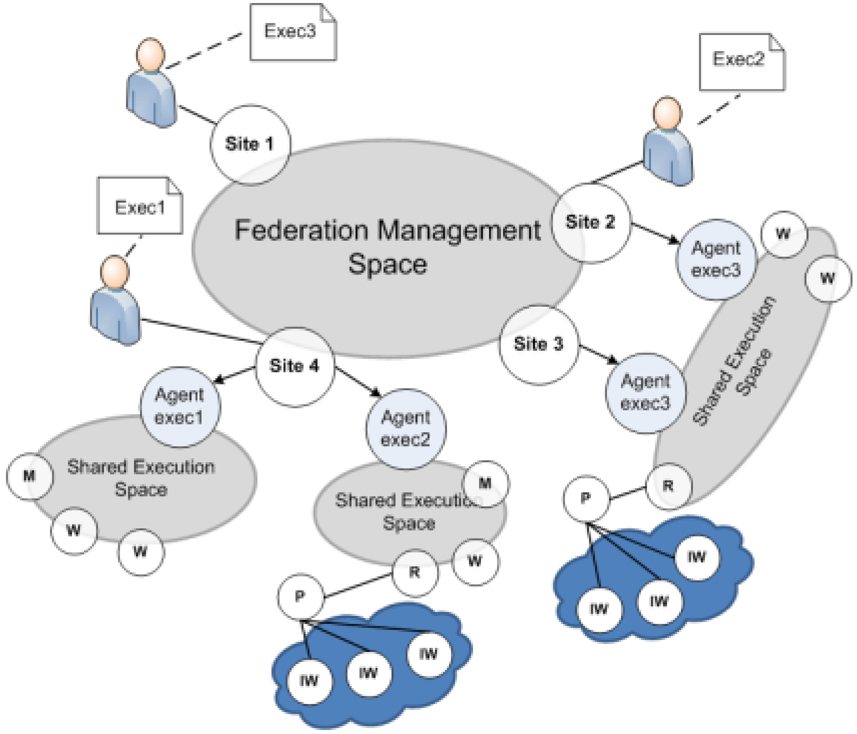


Multiple data access and processing layers
Deciding what to do where – creation of a “decision function”
Different objectives: L3: power, range; L2: stream aggregation; L1: throughput (use of “Software Define Networks” at L2)
No need to migrate “raw” data to Cloud systems

On-Demand Federation using CometCloud

- Cross-layer federation management using user and provider policies
- Federation is coordinated using Comet spaces at two levels

- Management space
  - Orchestrate resources in the federation
  - Interchange operational messages
- Shared execution spaces
  - Created on demand by agents
  - Provision local resources and connect to public clouds or external HPC systems



I. Petri, T. Beach, M. Zou, J. Diaz-Montes, O. Rana and M. Parashar, "Exploring Models and Mechanisms for Exchanging Resources in a Federated Cloud", IEEE international conference on Cloud Engineering (IC2E 2014), Boston, Massachusetts, March 2014.

Overview of the CometCloud Space

- Virtual shared space abstraction
  - Based on application properties
  - Mapped onto a set of peer nodes
- The space is accessible by all system nodes.
  - Access is independent of the physical locations of data tuples or hosts
- Coordination/interaction through the shared spaces
  - Runtime management, push/pull scheduling and load-balancing
- Dynamically constructed transient spaces enable application to exploit context locality

Implementation

- Requirements for a site to join the federation:
  - Java support
  - Valid credentials (authorized SSH keys)
  - Configure some parameters (i.e. address, ports, number of workers)
- Resources

Resources	Cardiff	Rutgers
Machines	12	32
Core per Machine	12	8
Memory	12 GB	6 GB
Network	1 GbE	Infiniband

- Indiana site
  - Uses FutureGrid (OpenStack, Infiniband interconnect, 2 cores/machine with 4GB memory) – also supports Cloudmesh Teefaa and Rain

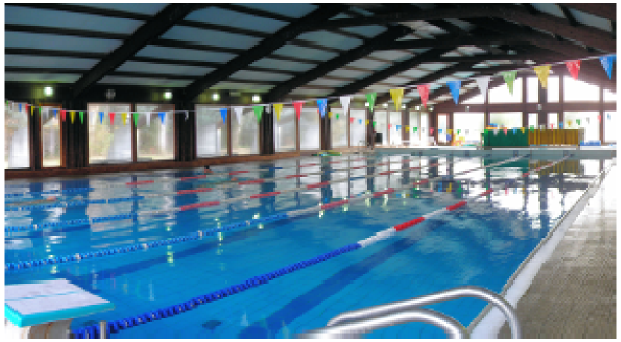
EnergyPlus and Building Optimisation

- Real time optimisation of building energy use
  - sensors provide readings within an interval of 15-30 minutes,
  - Optimisation run over this interval
- The efficiency of the optimisation process depends of the capacity of the computing infrastructure
  - deploying multiple EnergyPlus simulations
- Closed loop optimisation
  - Set control set points
  - Monitor/acquire sensor data + perform analysis with EnergyPlus
  - Update HVAC and actuators in physical infrastructure

EnergyPlus is a whole building energy simulation program that engineers, architects, and researchers use to model energy and water use in buildings. Modelling the performance of a building with EnergyPlus enables building professionals to optimize building design to reduce energy usage – http://apps1.eere.energy.gov/buildings/energyplus/

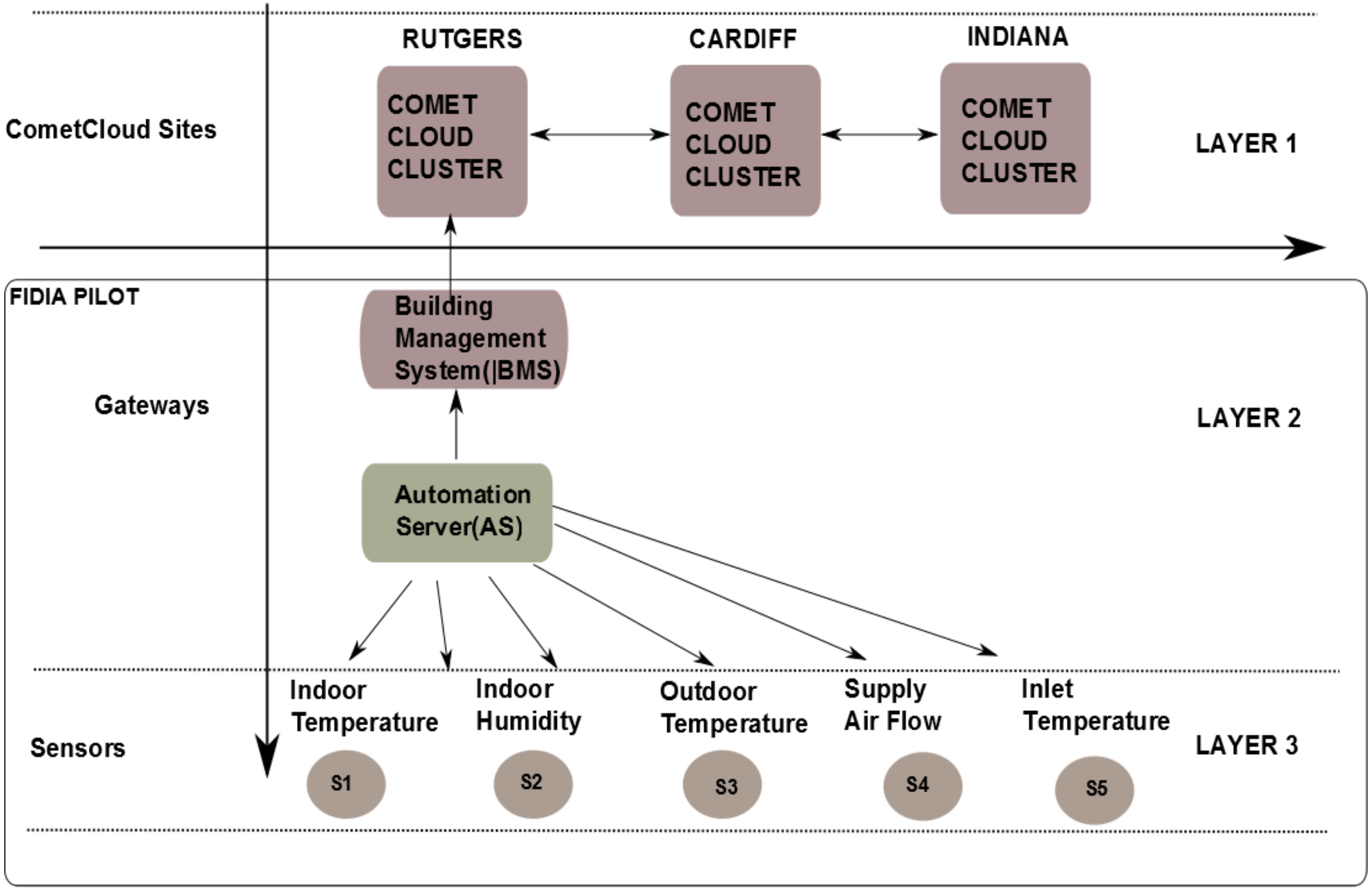
Instrumented Facility

CENTRO SPORTIVO FIDIA ROMA (http://www.asfidia.it/)



Pool (indoor) – size: 25m x 16m, depth: 1,60m to 2,10m, Capacity: 760 m³
Learning Pool (indoor) – size: 16m x 4 m, depth: 1m, Capacity: 64 m³
1 Gym (indoor) provided of electric equipment (electric bicycles, etc…)
1 Fitness room (indoor) size: 18m x 9m x 3m, Volume: 486m³
1 Volleyball court (indoor) – size: 40m x 28m x 8m, Volume: 8960 m³
2 Tennis/Five-a-side courts (outdoor, with changing rooms) – size: 30m x 20m

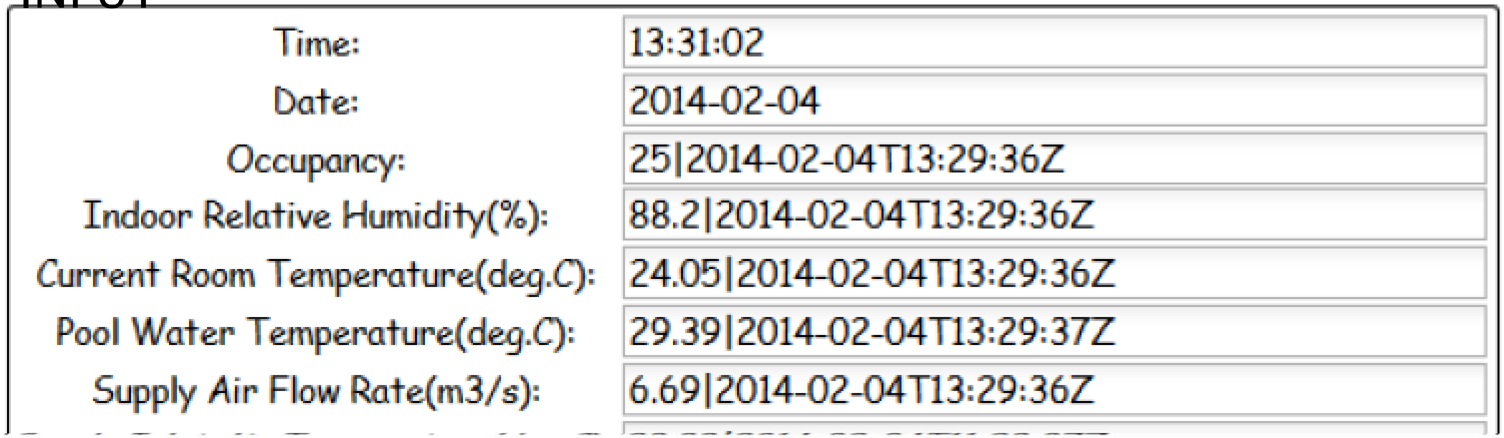
Federated Clouds in Building Optimisation



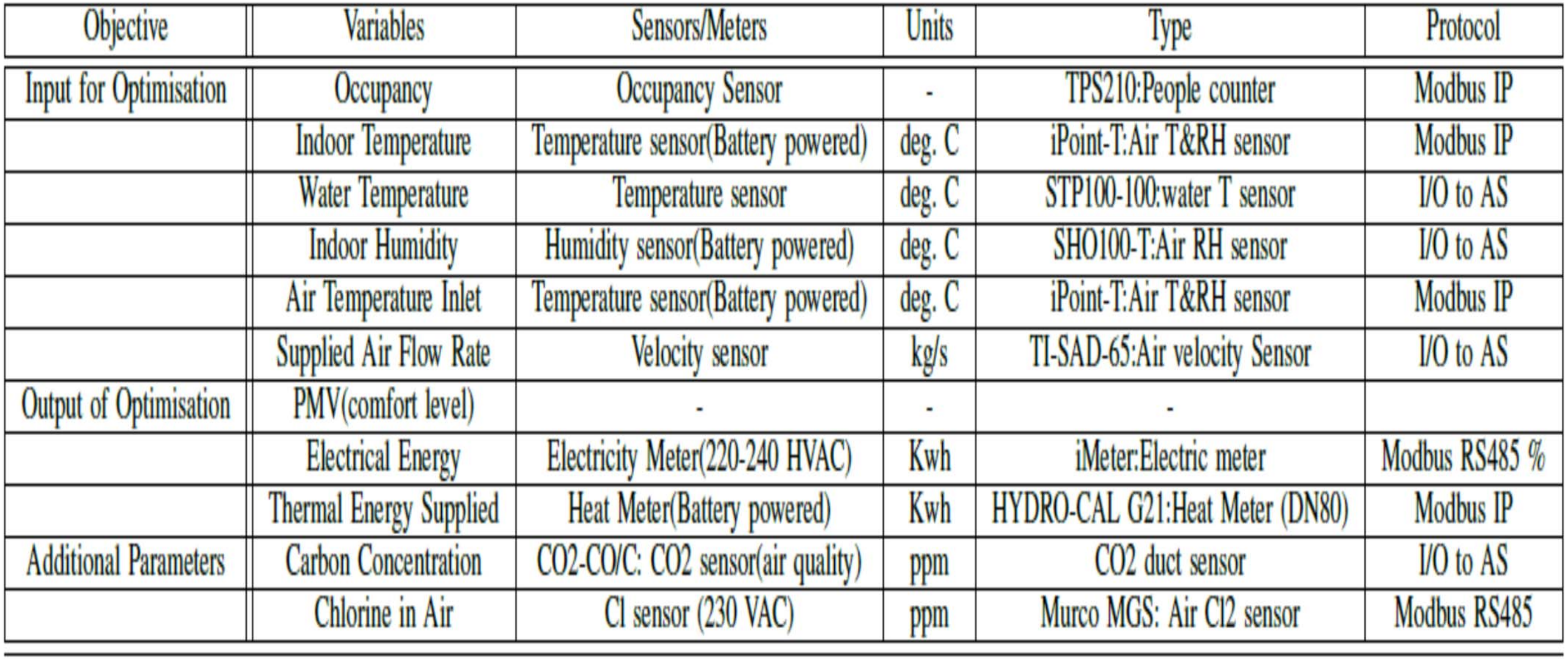
*I. Petri, O. Rana, J. Diaz-Montes, M. Zou, M. Parashar, T. Beach, Y. Rezqui, and H. Li, "**In-transit Data Analysis and Distribution in a Multi-Cloud Environment using CometCloud**," The International Workshop on Energy Management for Sustainable Internet-of-Things and Cloud Computing. Co-located with International Conference on Future Internet of Things and Cloud (FiCloud 2014), Barcelona, Spain, August 2014.*

INPUT

FIDIA Scenario 1

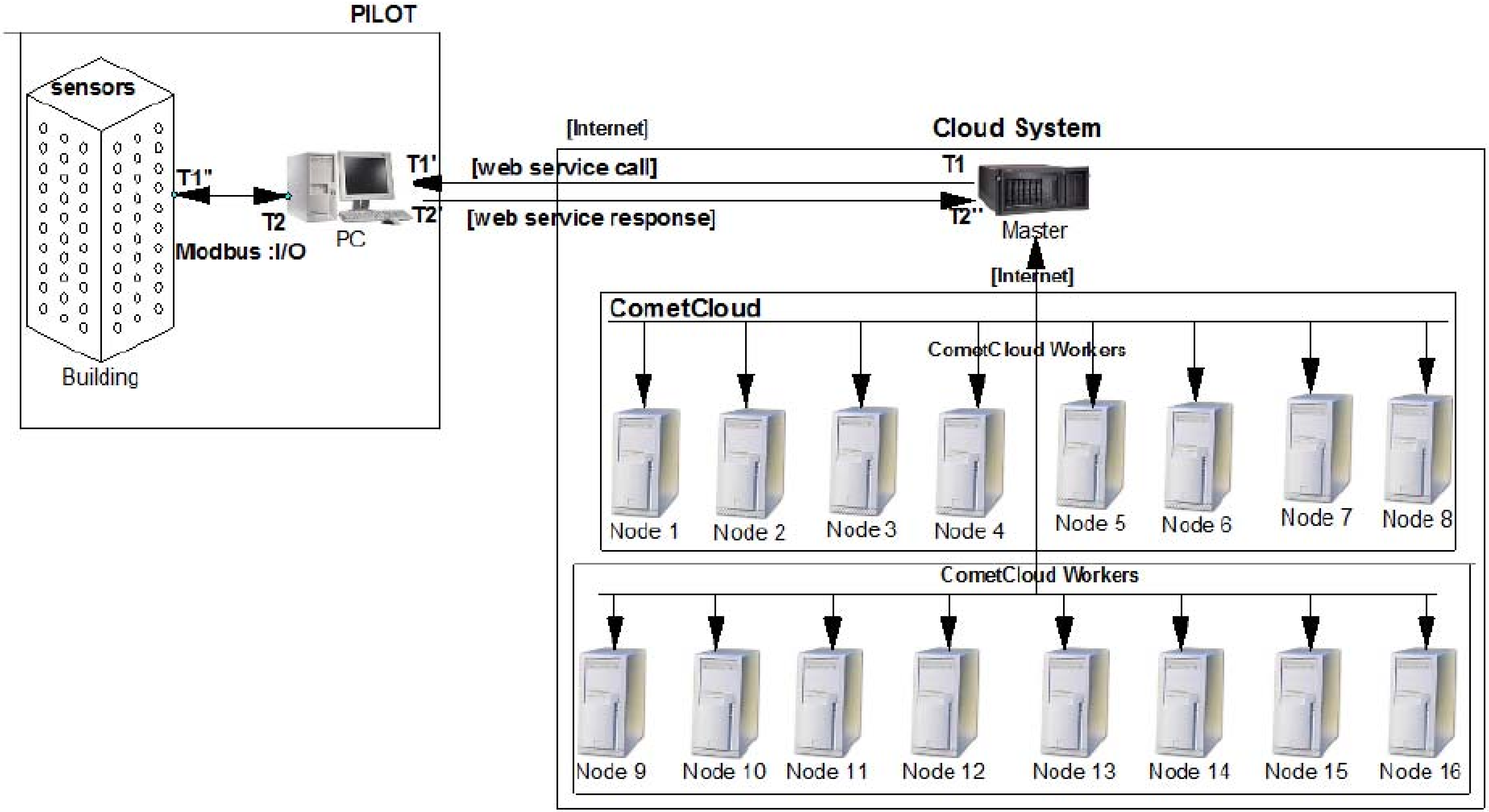


Time:	13:31:02
Date:	2014-02-04
Occupancy:	25|2014-02-04T13:29:36Z
Indoor Relative Humidity(%):	88.2|2014-02-04T13:29:36Z
Current Room Temperature(deg.C):	24.05|2014-02-04T13:29:36Z
Pool Water Temperature(deg.C):	29.39|2014-02-04T13:29:37Z
Supply Air Flow Rate(m3/s):	6.69|2014-02-04T13:29:36Z



Objective	Variables	Sensors/Meters	Units	Type	Protocol
Input for Optimisation	Occupancy	Occupancy Sensor	-	TPS210:People counter	Modbus IP
	Indoor Temperature	Temperature sensor(Battery powered)	deg. C	iPoint-T:Air T&RH sensor	Modbus IP
	Water Temperature	Temperature sensor	deg. C	STP100-100:water T sensor	I/O to AS
	Indoor Humidity	Humidity sensor(Battery powered)	deg. C	SHO100-T:Air RH sensor	I/O to AS
	Air Temperature Inlet	Temperature sensor(Battery powered)	deg. C	iPoint-T:Air T&RH sensor	Modbus IP
	Supplied Air Flow Rate	Velocity sensor	kg/s	TI-SAD-65:Air velocity Sensor	I/O to AS
Output of Optimisation	PMV(comfort level)	-	-	-	
	Electrical Energy	Electricity Meter(220-240 HVAC)	Kwh	iMeter:Electric meter	Modbus RS485 %
	Thermal Energy Supplied	Heat Meter(Battery powered)	Kwh	HYDRO-CAL G21:Heat Meter (DN80)	Modbus IP
Additional Parameters	Carbon Concentration	CO2-CO/C: CO2 sensor(air quality)	ppm	CO2 duct sensor	I/O to AS
	Chlorine in Air	Cl sensor (230 VAC)	ppm	Murco MGS: Air Cl2 sensor	Modbus RS485

EnergyPlus and Building Optimisation



Ioan Petri, Omer Rana, Yacine Rezgui, Haijiang Li, Tom Beach, Mengsong Zou, Javier Diaz Montes, Manish Parashar: "Cloud Supported Building Data Analytics". DPMSS workshop alongside CCGRID 2014: pp 641-650, Chicago, USA. IEEE Computer Society Press.

Federation constraints

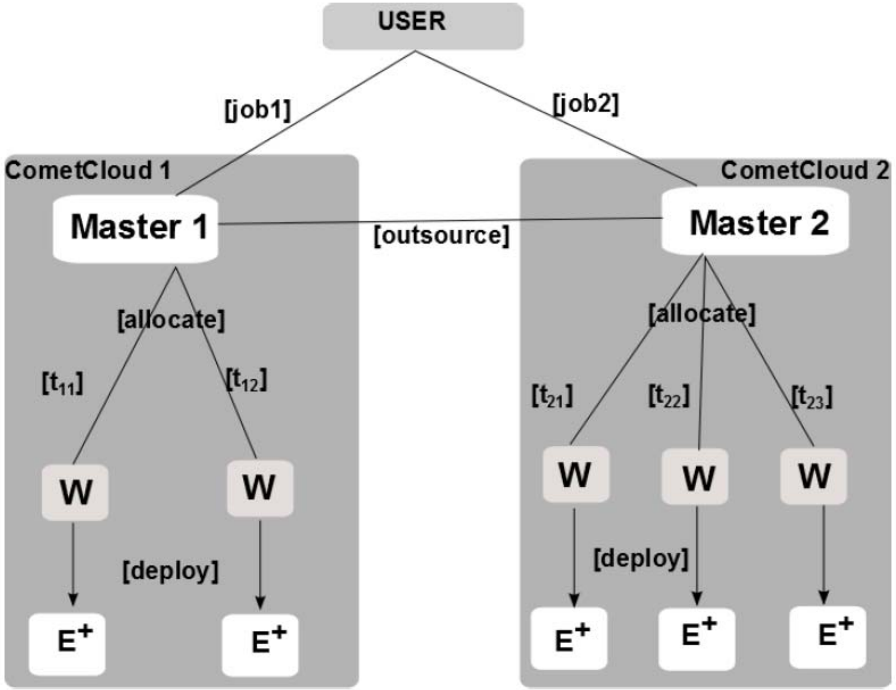
Two metrics:

- Time to complete
- Results quality

Trading quality of results vs. overall simulation time

cost function: f (X) : C -> R where C is a set of constraints(cost, deadline) and R is a set of decisions based on the existing constraints C.

- Each Master decides how to compute the received job :
  - (i) where to compute the tasks: (a) Single CometCloud or (b) federated CometCloud;
  - (ii) how many combinations to run giving the deadline received from the user.



Evaluation

- In our experiments we use two different configurations
  - (a) *single cloud context* where all the tasks have to be processed locally
  - (b) *federation cloud context* where the sites have the option of outsourcing tasks to remote sites.
- We use as inputs for our calculation
  - (i) *CPU time of remote site* as the amount of time spent by each worker to computer the tasks and
  - (*ii)storage time on remote site* as the amount of time needed to store data remotely.
- All the costs have been calculated in £ derived from Amazon EC2 cost.

Experiment 1: Job completed

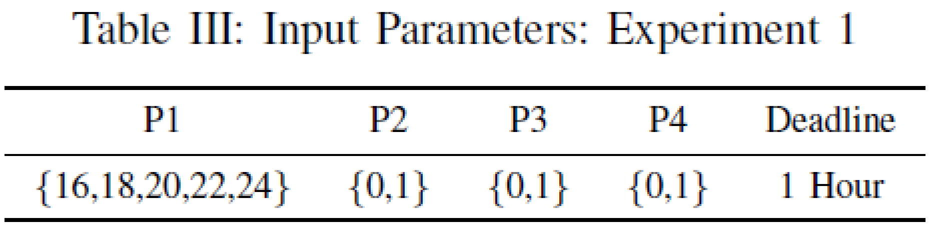


Table III: Input Parameters: Experiment 1

P1	P2	P3	P4	Deadline
{16,18,20,22,24}	{0,1}	{0,1}	{0,1}	1 Hour

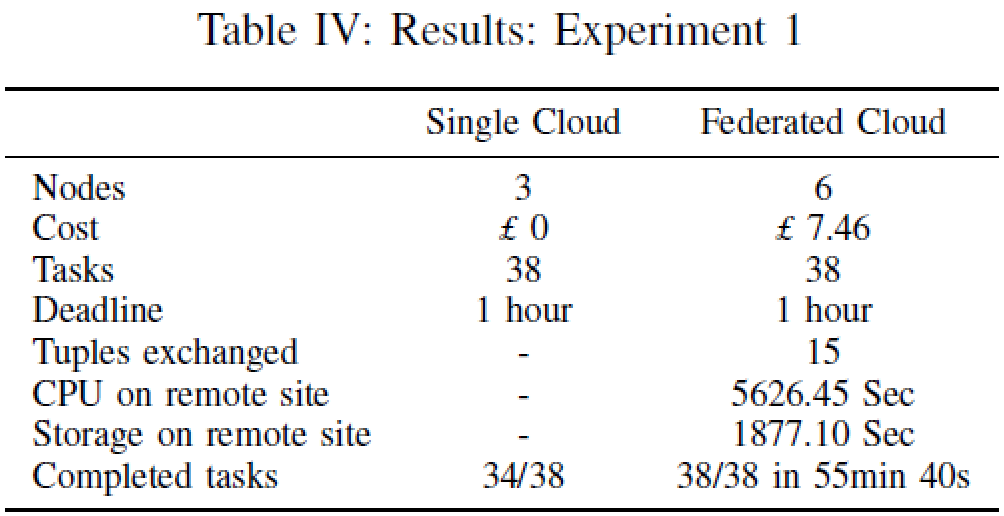


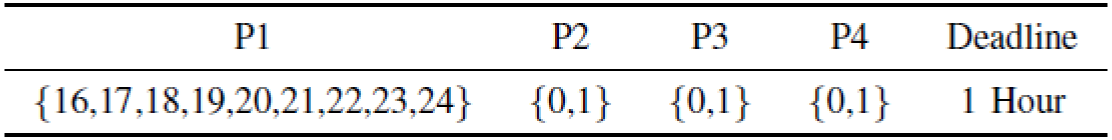
Table IV: Results: Experiment 1

	Single Cloud	Federated Cloud
Nodes	3	6
Cost	£ 0	£ 7.46
Tasks	38	38
Deadline	1 hour	1 hour
Tuples exchanged	-	15
CPU on remote site	-	5626.45 Sec
Storage on remote site	-	1877.10 Sec
Completed tasks	34/38	38/38 in 55min 40s

- the federation site has two options: (i) run tasks on the local infrastructure (single cloud case) or (ii) outsource some tasks to a remote site (federation cloud case)
- A corresponding deadline of 1 hour, only 34 out of 38 can be completed.
- In the federation in 55 minutes by outsourcing 15 to the remote site.
- The process of outsourcing has an associated cost of 7.46 £

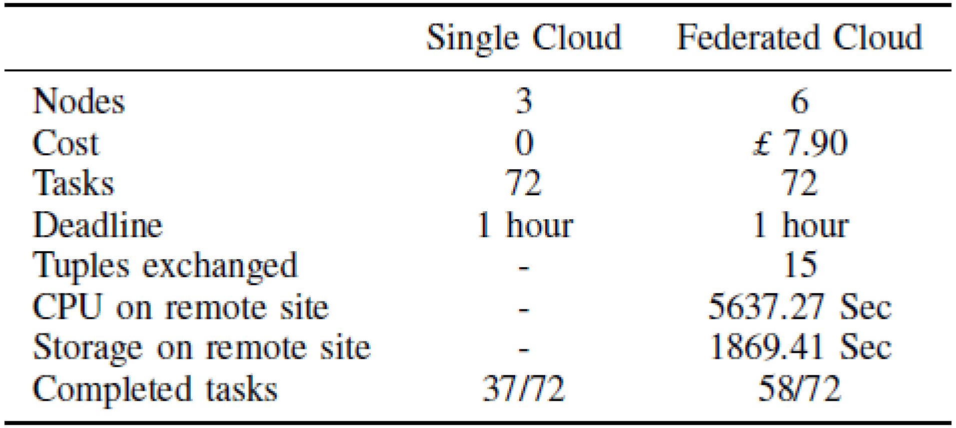
Experiment 2: Job uncompleted:

Table V: Input Parameter: Experiment 2



P1	P2	P3	P4	Deadline
{16,17,18,19,20,21,22,23,24}	{0,1}	{0,1}	{0,1}	1 Hour

Table VI: Results: Experiment 2

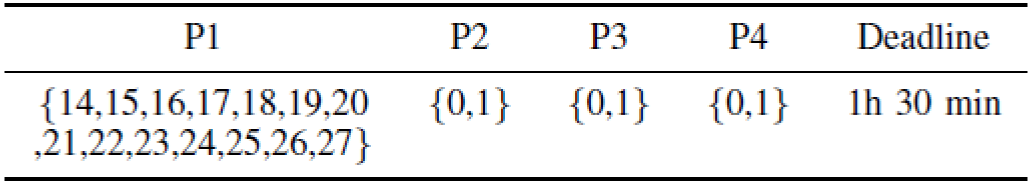


	Single Cloud	Federated Cloud
Nodes	3	6
Cost	0	£ 7.90
Tasks	72	72
Deadline	1 hour	1 hour
Tuples exchanged	-	15
CPU on remote site	-	5637.27 Sec
Storage on remote site	-	1869.41 Sec
Completed tasks	37/72	58/72

- In the context of single cloud federation (3 workers) only 37 out of 72 tasks are completed within the deadline of 1 hour.
- Exchanging 15 tuples between the two federation sites, with increased cost for execution and storage.

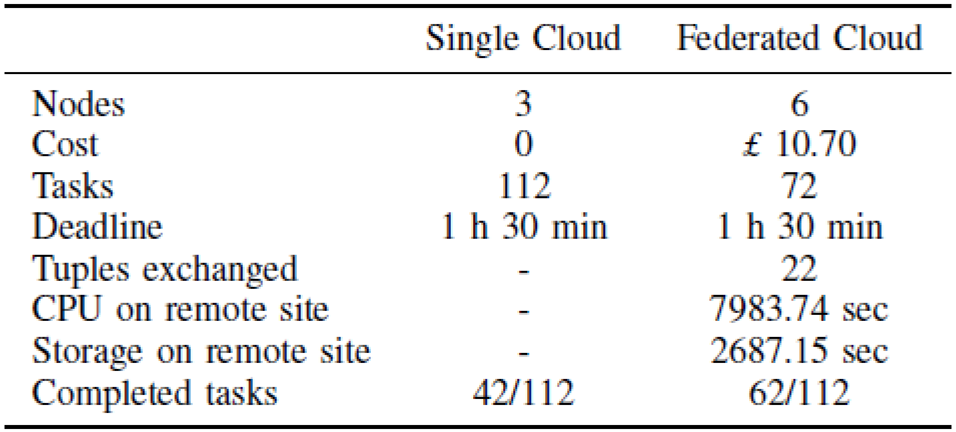
Experiment 3: Job uncompleted–parameters ranges extended:

Table VII: Input Parameters: Experiment 3



P1	P2	P3	P4	Deadline
{14,15,16,17,18,19,20 ,21,22,23,24,25,26,27}	{0,1}	{0,1}	{0,1}	1h 30 min

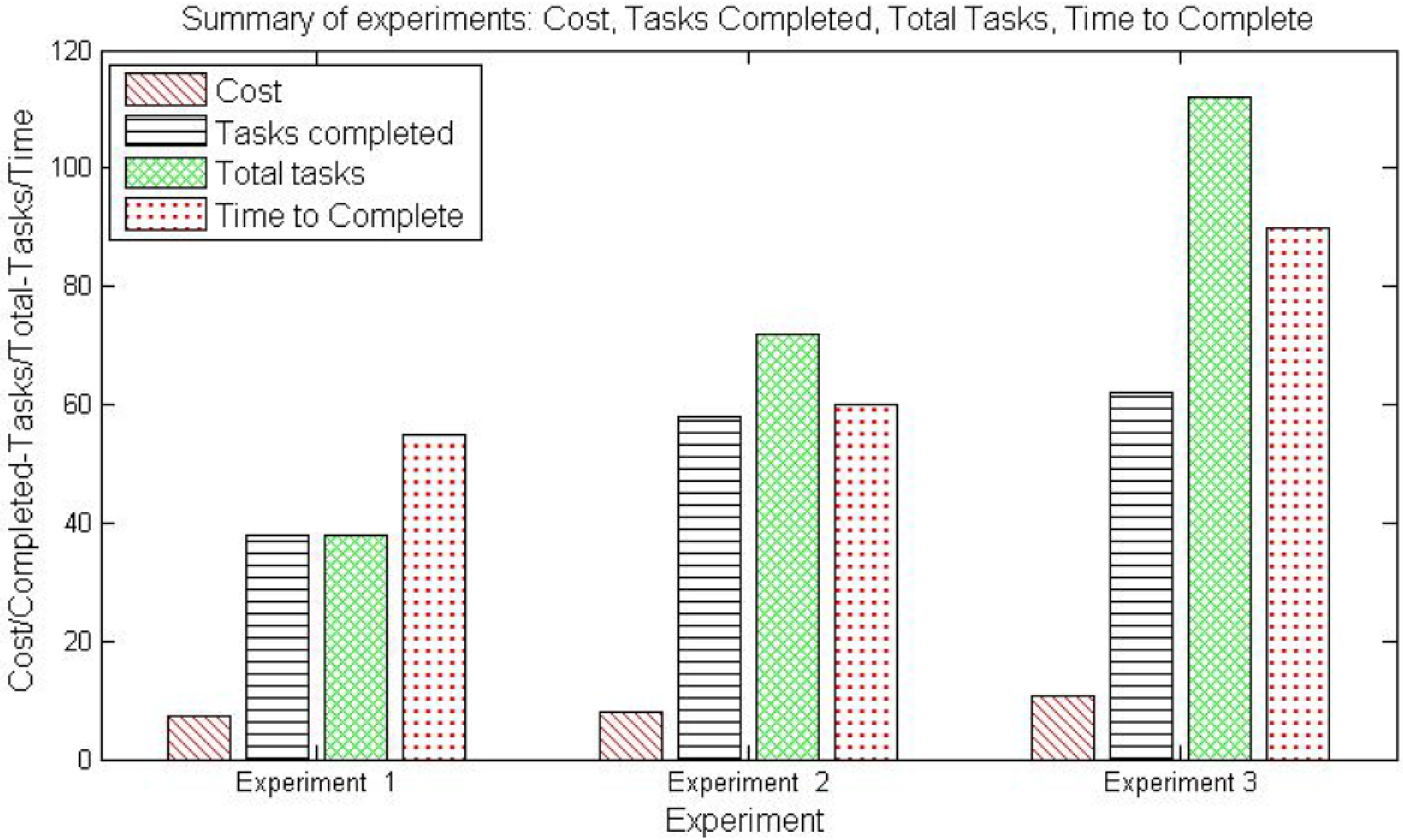
Table VIII: Results: Experiment 3



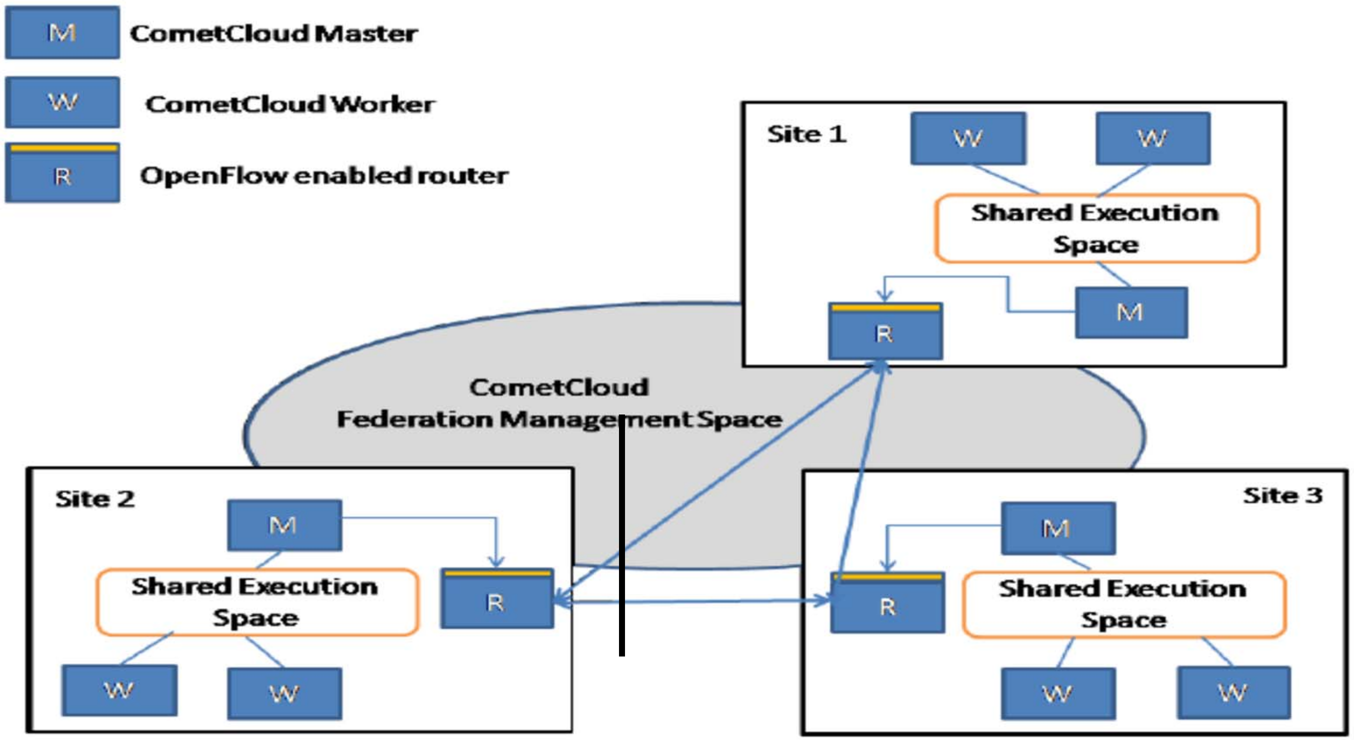
	Single Cloud	Federated Cloud
Nodes	3	6
Cost	0	£ 10.70
Tasks	112	72
Deadline	1 h 30 min	1 h 30 min
Tuples exchanged	-	22
CPU on remote site	-	7983.74 sec
Storage on remote site	-	2687.15 sec
Completed tasks	42/112	62/112

- we extend the deadline associated to 1 hour and 30 minutes
- when using the federation to outsource a percentage of tasks we observe that the number of tasks completed increases to 62

Summary of results



Integration with Software Defined Networks



SDN emulation using Mininet

Use of three sites (Rutgers (New Jersey, US), Cardiff (UK) and FutureGrid (Indiana, US) to simulate use of multi-hop interaction).

Different cost of execution per site + cost of network data transfer

Integration with Software Defined Networks: Experiments

- We consider that building data is available at each site, with data being generated at different rates
- The amount of input data to be transferred can be 10MB, 20MB, or 30MB
- We assume SDN capabilities are available across all sites.
- We allocate five SDN channels between each pair of sites with a guaranteed bandwidth of 1 Mbps
- We also have a network channel without QoS guarantees, called shared channel, that has a bandwidth of up to 0.2 Mbps

Ioan Petri, Mengsong Zou, Ali Reza Zamani, Javier Diaz-Montes, Omer F. Rana and Manish Parashar, "Integrating Software Defined Networks within a Cloud Federation", Cluster, Cloud & Grid Computing (CCGrid), Shenzen, China (May 2015). IEEE Computer Society Press.

Integration with Software Defined Networks: Configuration

- The policy used in our experiments is selecting the site that can *complete the workload with the minimum Time to Completion (TTC) subject to Cost < Budget*.

- If a job cannot be completed at any site within the given deadline and budget constraint, this job is declined.

- The TTC of a job is DataTransfer + ComputationTime.

- The Cost is DataTransferCost + ComputationCost.

- The shared network is free, while the cost of the SDN network varies based on utilization. The default cost of each SDN network channel is $0.05/second. This cost increases when utilization exceeds 50%.
  - The cost of the network is calculated as follows:

$$Cost = BaseCost * (1 + (\frac{ChannelsInUse}{TotalChannels} - 0.5) * 2) \tag{1}$$

Results

Table II: Total amount of time spent transfe
shared network and SDN network.

	Shared Network		
Transfer Time	Estimated	Real	Estin
Rutgers	560	4652	24
Cardiff	180	1588	4
Futuregrid	264	1489	6

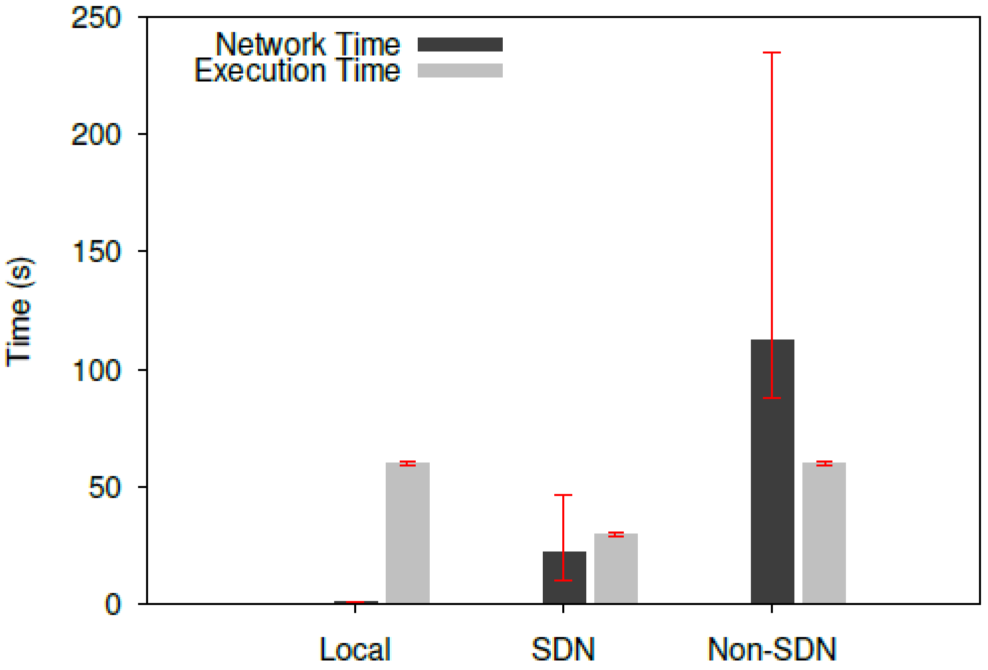
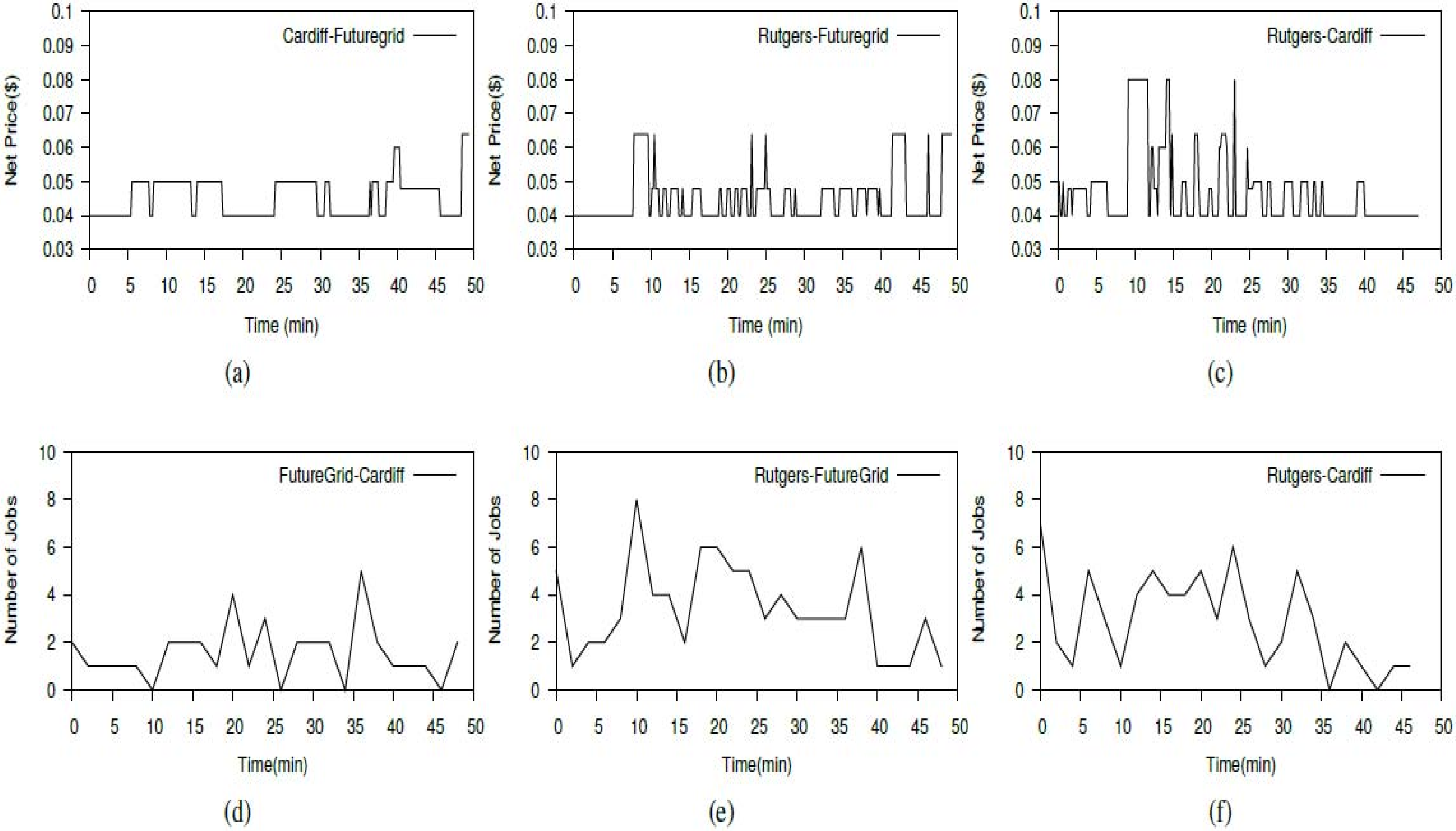


Figure 4: Average Execution and Network transfer time for local site and outsourcing to a remote site with and without SDN.

Table III: Number of jobs outsourced
computed locally.

job Source	Outsourced	Local
Rutgers	49	74
Cardiff	96	40
Futuregrid	100	30

Integration with Software Defined Networks: Overview



(a) (b) (c)

(d) (e) (f)

Figure 3: Summary of experimental results. At the top we have the price of reserving SDN over time and at the bottom we have the number of jobs outsourced using SDN over time.

Conclusion …

- Emergence of data-driven + data intensive applications
- Use of Cloud/data centres and edge nodes collectively
- Pipeline-based enactment a common theme
  - Various characteristics – buffer management and data coordination
  - Model development that can be integrated into a workflow environment
- Automating application adaptation
  - … as infrastructure changes
  - … as application characteristics change

Collaborators …

- *COSMOS:* *Jeffrey Morgan, Peter Burnap, William Housley, Matthew Williams, Adam Edwards (Cardiff) + Rob Procter (Warwick)*
- *Ioan Petri (Cardiff, CS)*
- *Yacine Rezgui, Haijiang Li. Tom Beach (Cardiff ENGIN)*
- *Manish Parashar, Javier Diaz-Montes, Ivan Rodero, Mengsong Zou (Rutgers University, US)*
- *Rafael Tolosana, Ricardo Rodriegez and Jose Banares (University of Zaragoza, Spain)*
- *Congduc Pham (University of Pau, France)*