Autonomics at the Edge:
Resource Orchestration for "Edge Native" applications

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

Collaborators in this work:

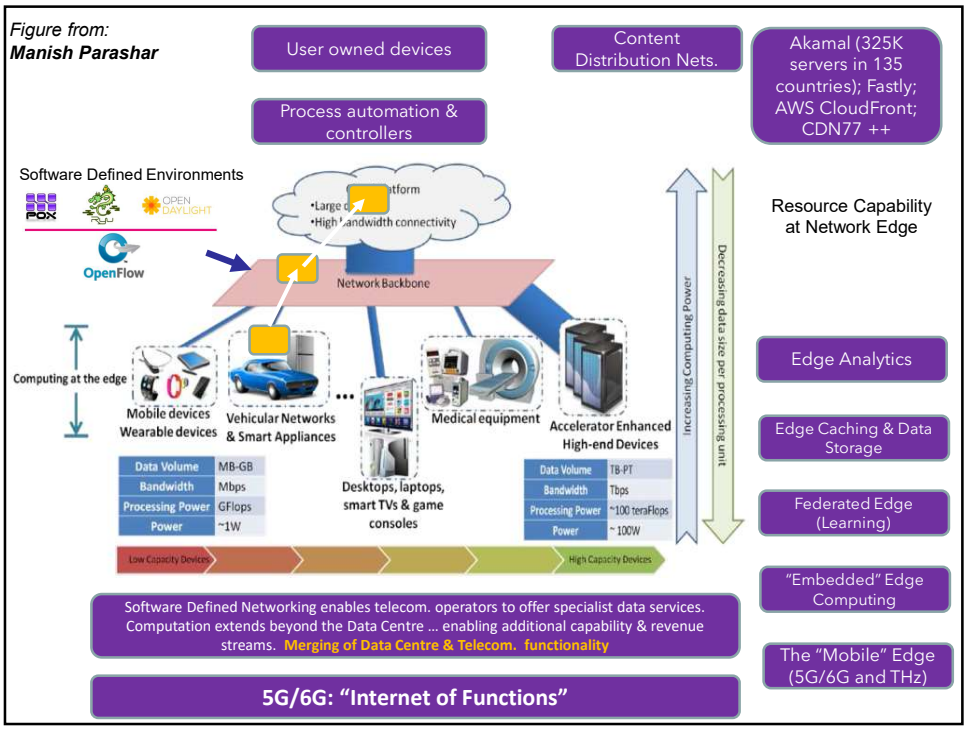
1

I ask more questions than provide answers ….

I hope some of these will be the basis for further collaborative work

Thanks NESTiD for the invitation

2



3

Autonomics & Edge Computing

Adapt execution of applications across the IoT to Cloud

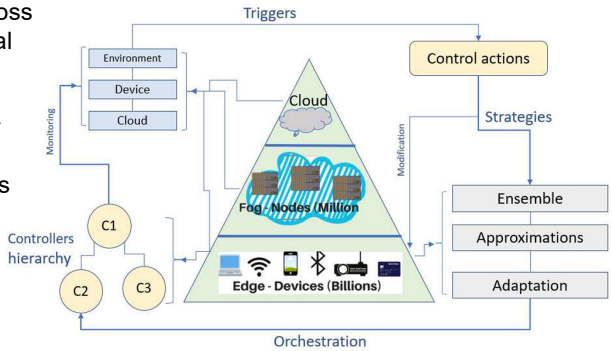
Use the MAPE-architecture from autonomic self-management

Level of access can vary across different layers of the practical Infrastructure

Increasing availability of user owned resources (e.g. home hubs, streaming "box", games machines)

Autonomic computing is rooted in the use of feedback control

Equivalent to adapting interactions and behaviours of multiple controllers



Autonomics at the edge: resource orchestration for edge native applications, *I. Petri, O. Rana, L. Bittencourt, D. Balouek-Thomert and M. Parashar. IEEE Internet Computing 25 (4), pp. 21-29, 2021. DoI: 10.1109/MIC.2020.3039551*

4

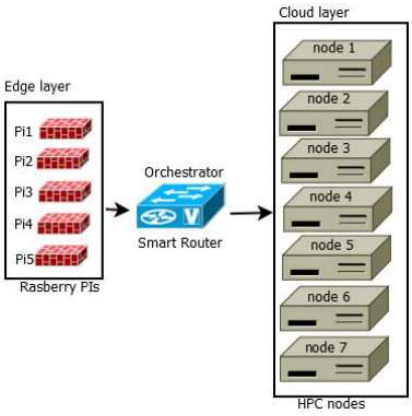
Edge-Cloud Orchestration

Dynamically determines the placement and scheduling of user applications to:

- improve utilisation of edge resources;
- meet overall application execution constraints such as deadline, network latency and security.

Supports:

- schedule tasks on locally available edge resource(s) or forward tasks to a cloud system.
- The forwarding process is based on properties and security credentials of resources
- aggregate tasks prior to forwarding these to a cloud system;
- disaggregate tasks prior to forwarding these to edge resources.



User tasks can be "tagged" – Using tags as a basis for scheduling

A. Singh, N. Auluck, O. Rana, A. Jones and S. Nepal, ***"Scheduling real time security aware tasks in fog networks".*** *IEEE Transactions on Services Computing 14(6), pp.1981-1994, 2021: DoI: 10.1109/TSC.2019.2914649*

5

What is at the "Edge"? (Pisces P...)

P

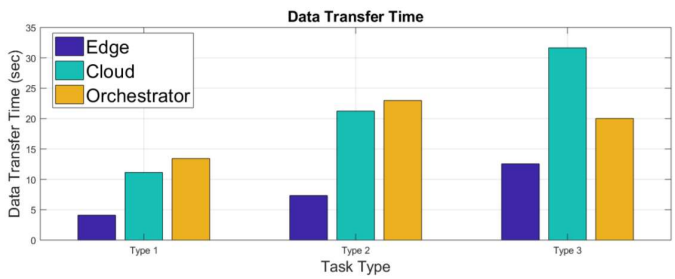


Table II: Job Information.

JobType	Data Size(MB)	Tasks†
JobType1	50	16
JobType2	100	24
JobType3	150	32

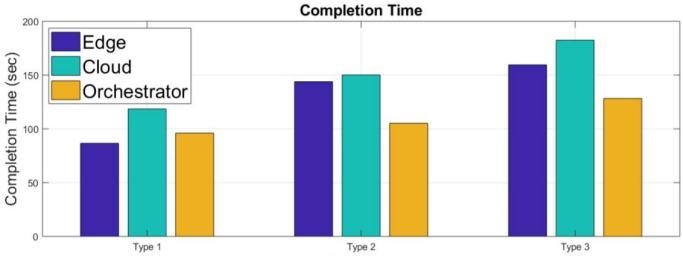
† – A job is composed of a set of tasks

...ystem: 17 double tubes (25 W ...ed at night for a short period

...roduces small pieces of ice for ... cold storage to preserve the fish. ...es about 32 Kw of power.

...ge is under operation all the time. ...d the most power-consuming ...he building. This is because of the low temperature needed (-5 degrees).

Box washing machine



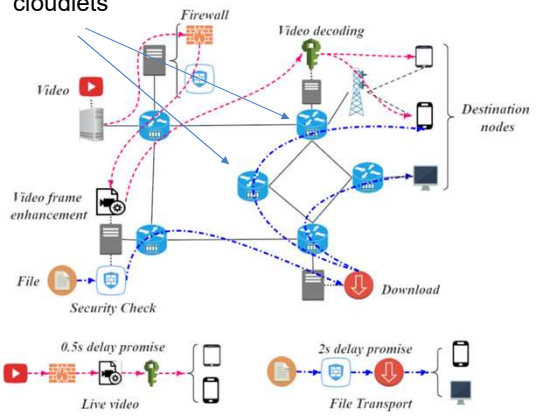
Appliance	Power R				
Box washing machine	50 KV				
Ice Flake machine	30kW				
Cold storage room	30kWh	180	Possible	Twice a day	Between (0:00-23:45)
Lighting system	25 W/per tube	60	Possible	Twice a day	Between (0:00-23:45)

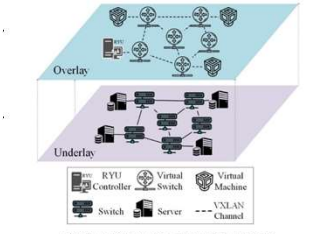
6

Edge & Intransit capability
(AI to optimise infrastructure & placement)

physical underlay consists of five switches: Huawei S5720-32C-HI-24S-AC, H3C S5560-30S-EI, Ruijie RG-5750C-28Gt4XS-H, CISCO 3750X-24T, and Centec aSW1100-48T4X. It also has five servers with i7-8700 CPU and 16GB RAM, & Raspberry PI with 1.2GHz CPU and 1GB RAM. Open vSwitch nodes emulated using Mininet.

"cloudlets"





(a) The underlay and overlay of the test-bed

(b) The physical deployment of the hardware switches

(c) The raspberry pi

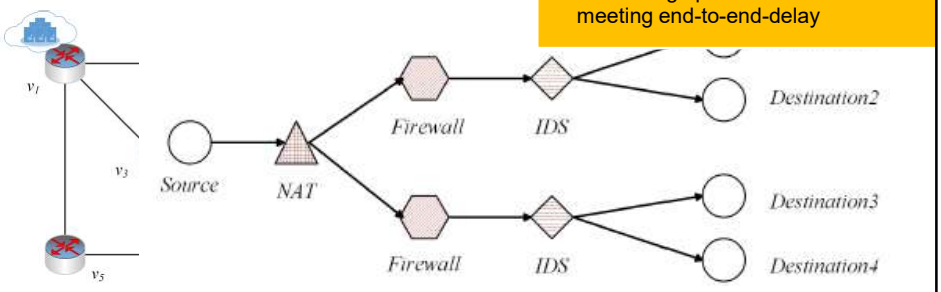
Qin, Yugen, Xia, Qiufen, Xu, Zichuan, Zhou, Pan, Galis, Alex, Rana, Omer, Ren, Jiankang and Wu, Guowei, **"Enabling multicast slices in edge networks".** *IEEE Internet of Things* 7 (9) , 2020, pp. 8485-8501.

7

Enacting Workflows across Cloud + Fog resources

- **Divide workflow into Virtual Network Functions (VNF) – identify placement of VNFs on "Cloudlets" (Edge/In-Network hosted)**
- **Metrics/Optimisation objectives:**
  - **Minimise overall cost of enactment**
  - **Achieve end-to-end delay target**
  - **Meet security constraints associated with**

Route traffic of a particular multi-cast request to each destination D_i:
- By chaining existing or newly instantiated VNF instances (across different cloudlets),
- Minimising operational costs & meeting end-to-end-delay



Z. Xu, Y. Zhang, W. Liang, Q. Xia, O. Rana, A. Galis, G. Wu and P. Zhou, **"NFV-enabled multicasting in mobile edge clouds with resource sharing"**. Proc. *48th International Conference on Parallel Processing (ICPP)*, Kyoto, Japan, 5-8 August 2019. ACM Press.

8

Three themes …

How do we partition machine learning algorithms across Edge-Network-Cloud resources?
What should run where (privacy, capacity, resilience).

Can machine learning algorithms be adapted based on the **characteristics of devices on which they are hosted?** What does this mean for **stability/convergence vs. performance?**

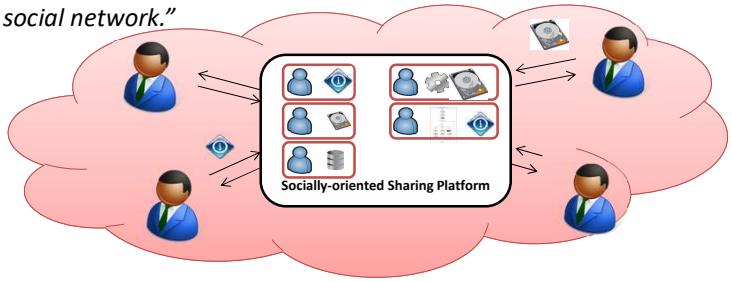
How can user **owned resources be more effectively used?** How are **resource characteristics taken into consideration?**

9

The vision of a social cloud

- Definition: (based on Chard et al. 2011)

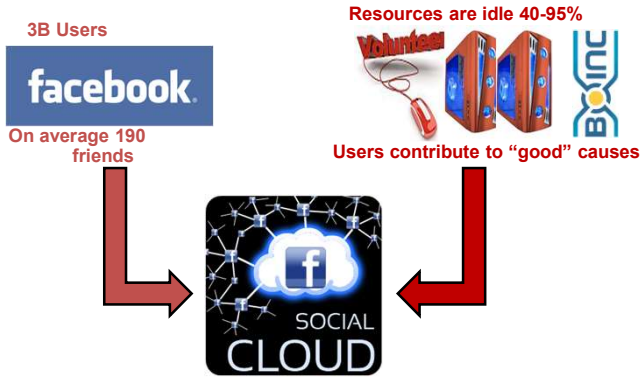
  "Social Clouds are a scalable, dynamic and user-centric resource sharing framework in which computational resources, services and information are shared amongst members on the premise of the relationships encoded in a social network."



Social Clouds enable the sharing of (heterogeneous) resources in a framework where the social structures infer an implicit level of trust

10

Social Cloud:
building on existing Social Network Platforms



Ubiquitous: Facebook ~ 3B users (Q4, 2021)
Some represent "trusted" relationships
Have notions of pre-existent trust fabric inherently interwoven into the network structure
Many applications now use social networks as a platform for:
- Authentication e.g. Facebook Connect
- Online Presence e.g. fb.com/your_page, Google Places (API)
- Application Portals e.g. progress thru processors, ASPEN and PolarGrid project

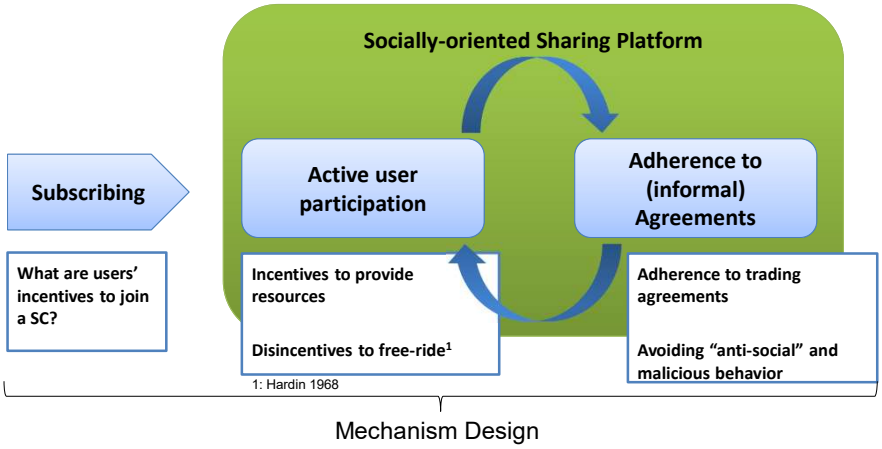
11

Kyle Chard, Simon Caton, Omer F. Rana, Kris Bubendorfer:
"**Social Clouds: A Retrospective",** IEEE Cloud Computing 2(6): 30-40 (2015)

Name	Resource type	Use of social network	Social network integration	Allocation
Social Compute Cloud*	Compute	Authentication and social-graph extraction	Facebook API integrated with the clearinghouse through the Django Social Auth plugin	Preference matching
Social Storage Cloud*	Storage	Authentication, application interface, and social-graph extraction	Integrated Facebook application	Economic (posted price/auction)
Social Content Delivery Network (S-CDN)*	Storage	Authentication and social-graph extraction	Facebook API and coauthorship network	Social network analysis
F2Box (FriendBox)	Storage	Authentication, application interface, and social-graph extraction	Integrated Facebook application	Equal allocation across friends' resources
Subdivision Social Cloud	Compute	Authentication and social-graph extraction	Integrated Facebook application	Economic (bartering)
SocialCloud	None	Social-graph extraction	None	Scheduling-based model
Cycle Sharing in Social Networks (CSSN)	Compute	Social-graph extraction, social network constructs to transport messages	Facebook API	Social postings
Multi-community-cloud collaboration (MC^3)	Compute	Collaboration across community clouds with social networks	None	Social network analysis
Community clouds and Community Networks Testbed for the Future Internet (CONFINE) project	Compute	Ad hoc extension to existing network topologies	None	Scheduling-based techniques and community economies

12

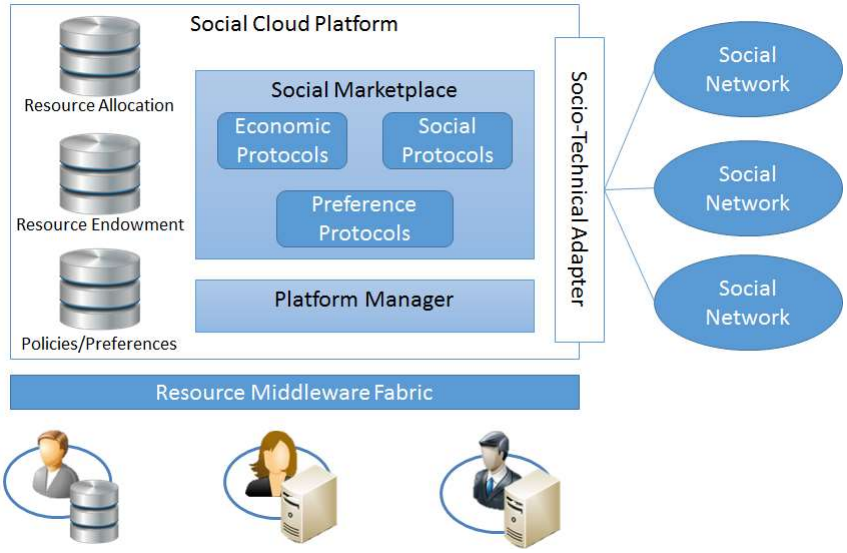
Incentives play a crucial role …



Mechanisms are able to provide incentives:
(i) **Monetary**: cost/service or cost/unit time;
(ii) **Non-Monetary**: badges, Top 10 list, reciprocal.

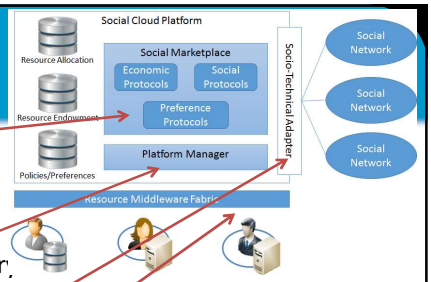
13

Platform Architecture



14

Architecture Components

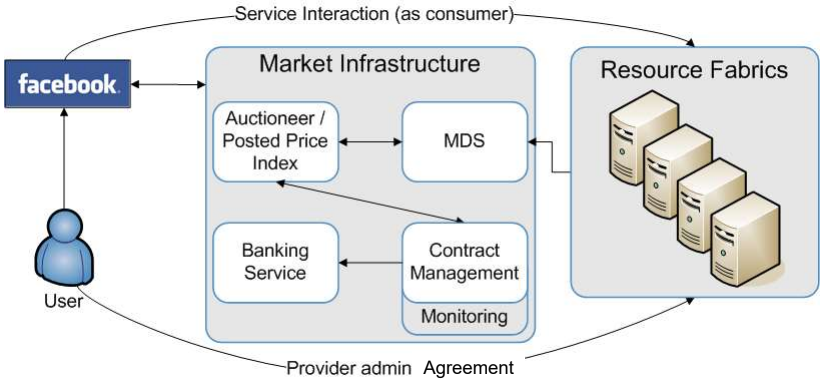


- Social Marketplace
  - Matching supply with demand
  - Protocols for resource allocation, rules of exchange, information store/registr
- Platform Manager
  - Overall system coordination
- Socio-Technical Adapter
  - Identify verification (e.g. authorization token)
  - Connect to various "types" of social networks (DBLP, Facebook etc)
- Resource Fabrics
  - Virtualisation & Sandboxing mechanism
  - Integration with Globus end points, Docker/OpenWhisk, Seattle VM, etc

15

Social Cloud: Platform for Data Storage (Demonstrator)

- Simple Storage Service Implemented as a Facebook application
- Use Case: a back up facility



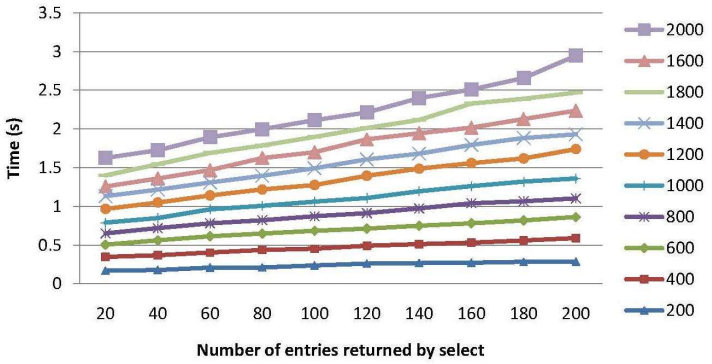
Kyle Chard, Kris Bubendorfer, Simon Caton, Omer F. Rana, ***"Social Cloud Computing: A Vision for Socially Motivated Resource Sharing".*** *IEEE Transactions on Services Computing 5(4): 551-563 (2012)*

16

17

Posted Price Scalability

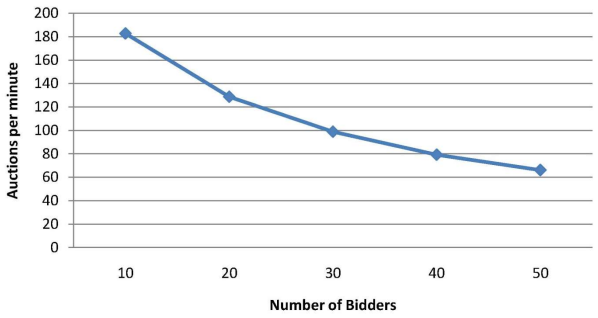
- Varying the size of the MDS (directory service) and number of matches
- With a size of 2000 sites/nodes, 100 matches can be discovered in ~ 2 seconds



18

Auction Scalability

- 500 Auctions and the worst case scenario:
  - all auctions run concurrently
- 50 bidders can complete 65 auctions per minute
- Under our assumptions this is already enough for a large social network



19

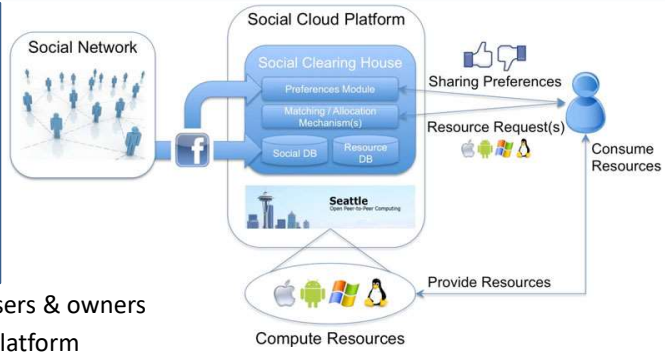
Social Clouds: types of data (e.g. podcasts, blogs, photos, music, documents)

- User generated content may be:
  - ***intermittent content*** often required at different time intervals but not continuously;
  - ***temporary content*** required for a short period of time (e.g. processing memory for running an experiment) and often only once;
  - ***backup content*** required with the highest security implications and privacy;
  - ***working content*** that can be accessed in real time and continuously.

20

Processor Sharing – via extended Seattle VMs

*Seattle is an open research and educational testbed that utilizes **computational resources provided by end users on their existing devices**. Unlike most other platforms, **resources are not dedicated to the platform which allows a greater degree of network diversity and realism at the cost of programmability**. Seattle is designed to preserve user security*



- Matching between users & owners
- Seattle -- Open P2P platform
  - Seattle "Clearing house" mechanism. 10 "vessels" (VMs) for each new install
  - Node Manager: gatekeeper for resources deployed on every contributed resource (credential checking for VM interaction)
  - Host machine location (in a lookup service) + Public/Private keys generated
  - Repy (Reduced Python for sandboxed environments)

Experience with Seattle: A Community Platform for Research and Education
https://ssl.engineering.nyu.edu/papers/zhuang_seattle_gree_13.pdf

21

Processor Sharing – via extended Seattle VMs

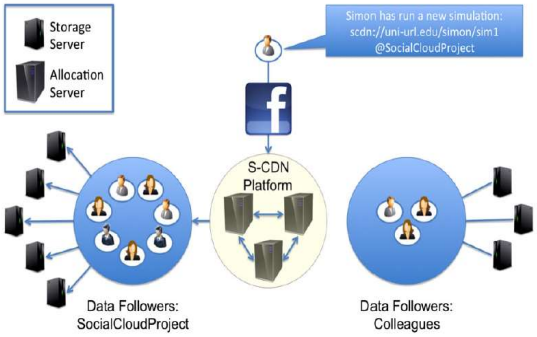
- Identify list of donation nodes
- Filter list based on "friends list" for a particular user
- Match mechanism
  - Select consumer preferences for each friend
  - Select preferences for each friend for requesting user
- Extends Seattle's implementation of (pseudo) random allocation to reduce user/donation permutations

*Simon Caton, Christian Haas, Kyle Chard, Kris Bubendorfer, Omer F. Rana, **A Social Compute Cloud: Allocating and Sharing Infrastructure Resources via Social Networks.** IEEE Transactions on Services Computing 7(3): 359-372 (2014)*

22

Data "Followers" & "Survivability" dynamics (The "Data Wildfire")

- Register interest in a data set
  - Equivalent to "Like" (Facebook) and "Favourite' (Twitter)
  - Event generated on subsequent update on a data set
- Enable popular data set to be propagated
  - Equivalent to a "Share" (Facebook) and "Re-Tweet" (Twitter)
  - Enables data sets with community interest to become popular over time
- Can be useful as a basis to support resource allocation

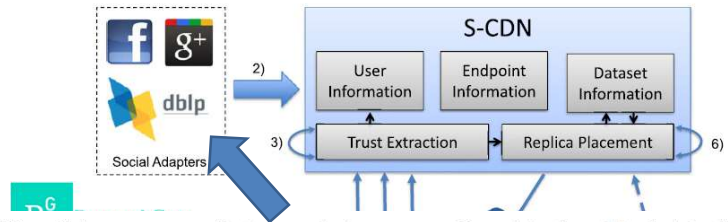


Resource Server: temporary storage at edge resources (user owned)

Allocation Server: overlay server to support data storage – for replicas, cache & repository

23

Data "Followers" & "Survivability" dynamics (The "Data Wildfire")

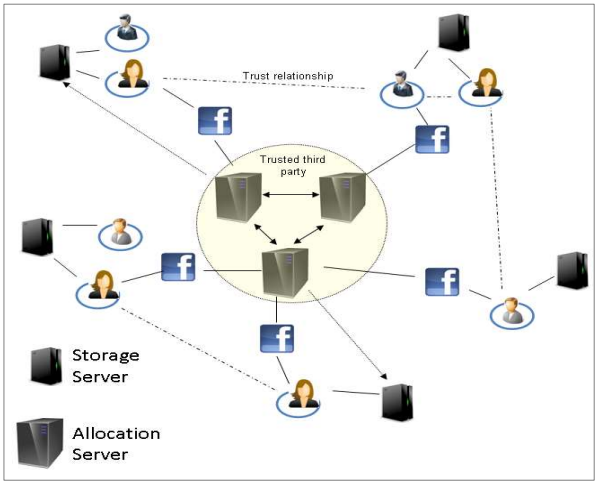


User data: username, first name, last name, email, registration data, last login.
SocialAuth: social network type, social network id, oauth token.
Endpoint data: id, owner, endpoint name.
Datasets: id, owner, dataset name, endpoint, relative path.
Dataset versioning: owner, date, type <"endpoint", "registration", "update", "download">
Dataset replicas: dataset id, owner, endpoint, relative path.
Download metrics: dataset id, downloader, download time.
Relationship metrics: owner, follower, score.

Social Adapters to identify "community" dynamics
Data set placement and resource allocation based on these properties (use of Globus End points (https://www.globus.org/)– with Argonne National Lab.)

24

A Social Content Delivery Network for Scientific Cooperation



Chard, Kyle, Caton, Simon, Kugler, Kai, Rana, Omer and Katz, Daniel S. 2017. ***A social content delivery network for eScience****. Concurrency and Computation: Practice and Experience 29 (4), e3854. 10.1002/cpe.3854*

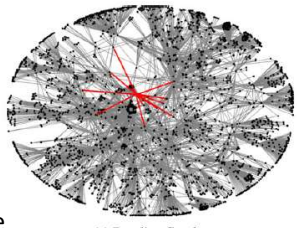
Replica Placement:

- Random
- Node Degree: highest no. of edges
- Community Node Degree (highest degree within a community, i.e. no adjacent placement)
- Clustering Coefficient (similar to highest betweenness scores)

25

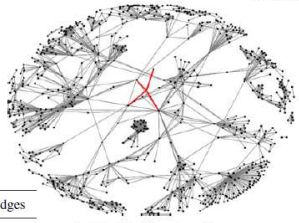
Scenario and Community Representation
(Trust "Boot strapping" problem)

- Baseline Graph: DBLP publications graph (Kyle): 3 degrees
  - Nodes: authors, Edges: co-authorship of 1 or more papers
- Double co-authorship: at least 2 publications
- No. of Authors: ≤ 6 authors on the paper
- Trust: captured through prior collaborative work

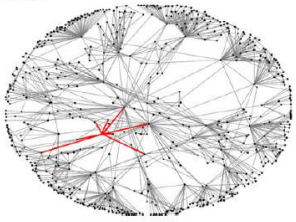


(a) Baseline Graph

Would you trust your co-authors to host your data?



(b) Double Co-authorship



(c) Number of Authors.

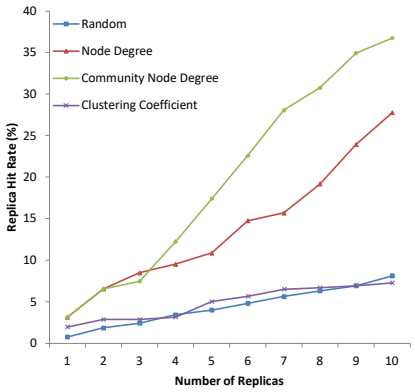
Graph	Nodes	Publications	Edges
Baseline	2335	1163	17,973
Double-author	811	881	5123
Number of authors	604	435	1988

26

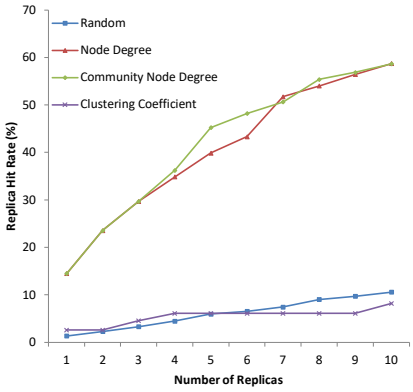
26

Results (60 repetitions) – miss/hit rate in caching

Double Coauthorship



No. of Coauthors



Kyle Chard, Simon Caton, Omer Rana and Dan Katz, ***A Social Content Delivery Network for Scientific Cooperation: Vision, Design, and Architecture****, 3rd Int. Workshop on Data Intensive Computation in the Cloud (DataCloud), ACM/IEEE "SuperComputing" conference, Salt Lake City, Utah, November 11, 2012*

27

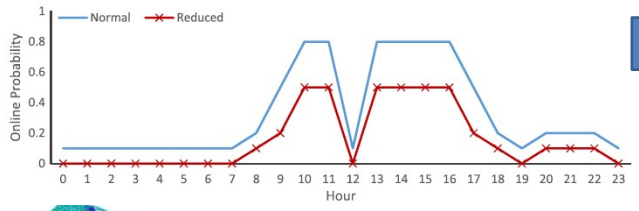
27

Data "Followers" & "Survivability" dynamics (The "Data Wildfire")

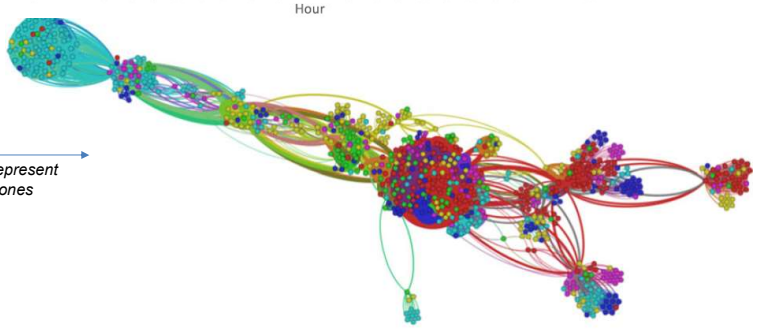
- Register interest in a data set
  - Equivalent to "Like" (Facebook) and "Favourite" (Twitter)
  - Event generated on subsequent update on a data set
- Enable "interesting" data set to be propagated
  - Equivalent to a "Share" (Facebook) and "Re-Tweet" (Twitter)
  - Enables data sets with community interest to become popular over time
- Can be useful as a basis to support resource allocation

28

Device Availability Profile: User Owned vs. Overlay Servers



Availability of user owned devices (Begole et al.)

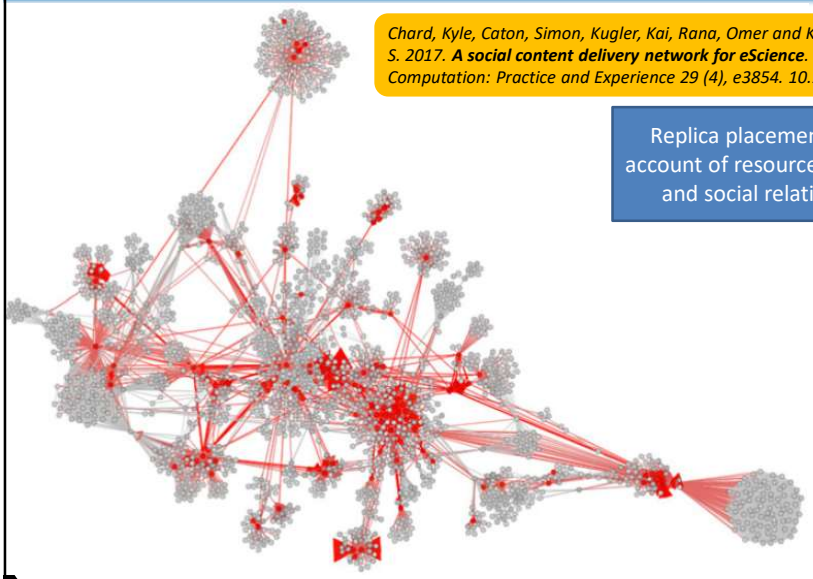


Colour codes represent Different time zones

Begole JB, Tang JC, Smith RB, Yankelovich N. Work rhythms: analyzing visualizations of awareness histories of distributed groups. In: *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, ACM, New Orleans, LA, 2002; 334–343

29

Replica distribution: Red endpoints contain > 5 replicas

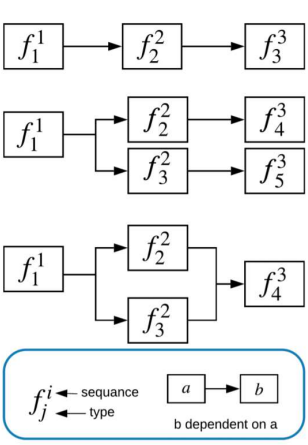


Chard, Kyle, Caton, Simon, Kugler, Kai, Rana, Omer and Katz, Daniel S. 2017. ***A social content delivery network for eScience****. Concurrency and Computation: Practice and Experience 29 (4), e3854. 10.1002/cpe.3854*

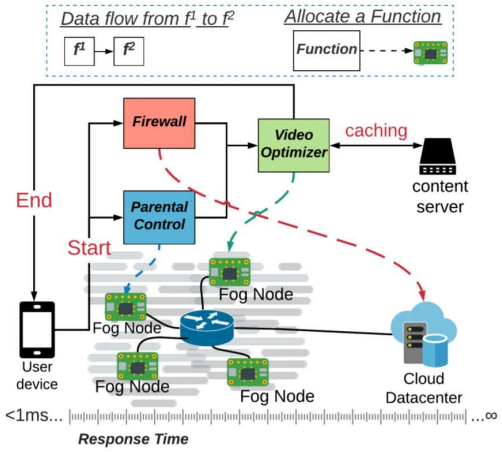
Replica placement, taking account of resource properties and social relationships

30

Function Execution … SFC (w. Osama AlMurshed)



(a) Examples of input-output data dependencies



(b) SFC allocation processes in the Fog and Cloud

Objectives:
Minimise Completion time & Risk – given a number of available "locations"
Subject to a maximum number of replicas. Identify also "similarity" score for choosing locations

31

Parsl: Interactive parallel programming in Python (from Dan Katz) -- http://parsl-project.org

Apps define opportunities for parallelism
- Python apps call Python functions
- Bash apps call external applications

Apps return "futures": a proxy for a result that might not yet be available

Apps run concurrently respecting data dependencies. Natural parallel programming!

```
pip install parsl
```

```
@python_app
def hello ():
    return 'Hello World!'
print(hello().result())
```

python

```
Hello World!
```

```
@bash_app
def echo_hello(stdout='echo-hello.stdout'):
    return 'echo "Hello World!"'
echo_hello().result()
with open('echo-hello.stdout', 'r') as f:
    print(f.read())
```

BASH

```
Hello World!
```

Parsl Executor & Parsl apps

Replacing "Seattle" with Parsl/FuncX endpoints

ILLINOIS NCSA | THE UNIVERSITY OF CHICAGO | Argonne NATIONAL LABORATORY

32

32

funcX: used alongside Parsl (from Dan Katz)

Turn **any** machine into a function serving endpoint

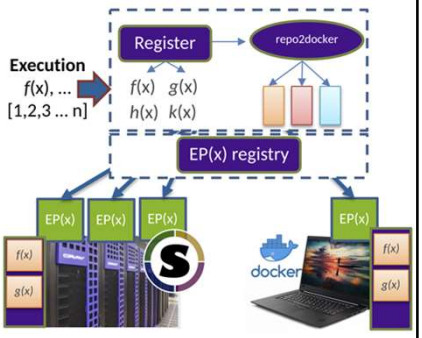
Overcome heterogeneity in distributed infrastructure

Functions:

- Register once & can associate a container for encapsulation
- Authn/z (via Globus Auth) for user execution
- Add Globus group to a function to share it

Endpoints:

- Lightweight agent that can be deployed by users
- Abstracts underlying resource and elastically scales to demand



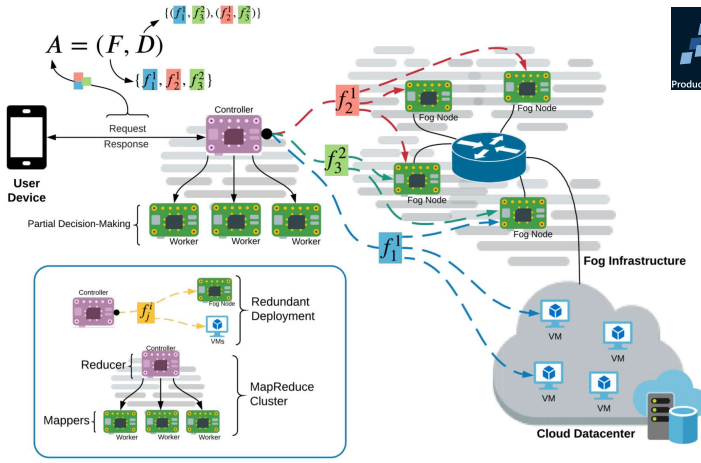
Function - A snippet of Python code that performs an activity

Endpoint - A logical entity that represents a compute resource and can execute a function

funcX service - A cloud-hosted service to register functions, invoke functions, and retrieve results

ILLINOIS NCSA | THE UNIVERSITY OF CHICAGO | Argonne NATIONAL LABORATORY

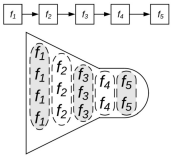
Function Execution ... SFC



Parsl
Productive parallel programming in Python
+ Jupyter integration

Understanding the replication "tunnel" to support failure at the edge

Close alignment with function replication across Edge and Cloud resources



Function Execution ... SFC

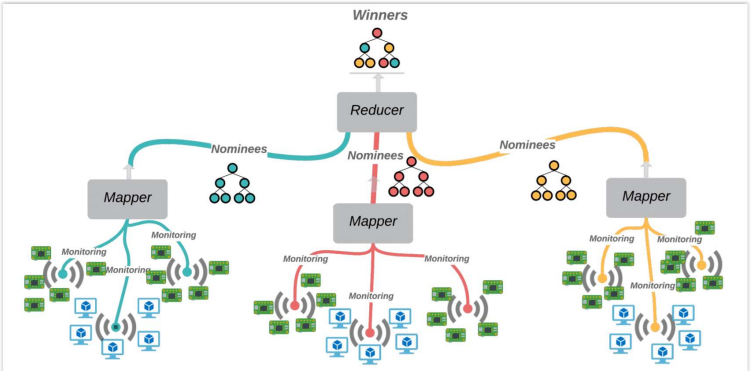


FIGURE 5 Each Mapper has a group of locations to monitor, each group has its color (green, red and yellow). The final Max-heap has a variety of node's color, due to them coming from different Mappers

The GNH uses **MapReduce to identify potential locations**

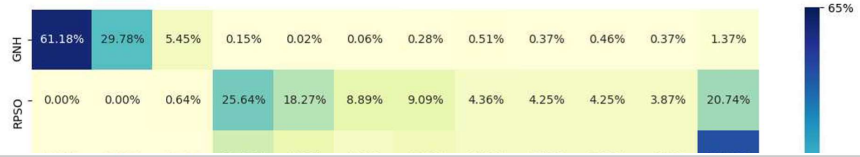
search space is divided between workers, i.e., Mappers.

Mapper: Identify potential locations within their group of nodes -- keeps a local record of the locations they monitor, and the records are results of monitoring the computing resources and the network connections linking these locations

Reducer: receives the result from every Mapper, then concatenates them, then applies a Max-heap push-pop function to each location in the Mappers' result. Decides *optimal* locations for redundant deployment.

35

Function Execution ... SFC



Version	CPU	Core(s)	Memory			Storage			Network Interface Speed	
RPi 3 Model A+	1.4 GHz	4	256 MB	512 MB		8 GB	16 GB	32 GB	300 Mbps	
RPi 1 Model B	700 MHz	1	256 MB	512 MB		8 GB	16 GB	32 GB	100 Mbps	
RPi 1 Model B+	700 MHz	1	256 MB	512 MB		8 GB	16 GB	32 GB	100 Mbps	
RPi 2 Model B	900 MHz	4		1 GB		8 GB	16 GB	32 GB	100 Mbps	
RPi 3 Model B	1.2 GHz	4		1 GB		8 GB	16 GB	32 GB	100 Mbps	300 Mbps
RPi 3 Model B+	1.4 GHz	4		1 GB		8 GB	16 GB	32 GB	300 Mbps	1000 Mbps
RPi 4 Model B	1.5 GHz	4	1 GB	2 GB	4 GB	8 GB	16 GB	32 GB	300 Mbps	1000 Mbps
RPi Zero W	1 GHz	1		512 MB		8 GB	16 GB	32 GB	300 Mbps	

10 20 30 50 60 701 1101 2001 3001 4001

Variable	Number/Rangs
Application requests	10,000,000
SFC length	(1-20)
Location	100
FNs	80
VMs	20
FN's Latency	(21 - 50 ms)
VM's Latency	(50 - 300 ms)
FN MTTF	(10 - 30 ms)
FN MTTR	(5 - 15 ms)
VM MTTF	(30 - 300 ms)
VM MTTR	(2 - 10 ms)

"Greedy Nominator Heuristic (GNH): Virtual Function Placement on Fog Resources" Osama Almurshed, Omer Rana and Kyle Chard, Concurrency & Computation: Practice & Experience, 2022

36

In Conclusion

- Social Clouds provide an important user-driven alternative to data-centre based Clouds
  - Wuala networks, AmazingStore (China), etc
- Issues of Trust, Reputation and Economic incentives is key
  - Include other factors: availability, reliability, uptime, power usage, etc
  - Traditionally captured through Service Level Agreements
- Current focus: Broker "emergence" in Social Clouds
  - Identify dominating sets in a social graph
  - Implementation using CometCloud

37