

# **Interactive Application Deployment Planning for Heterogeneous Computing Continuums**

**Daniel Hass, Josef Spillner**

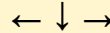
**May 14, 2021 | AINA-M2EC | ~~Toronto~~ Online**

# Continuum Computing



[Waste sorting, source: Zen Robotics]

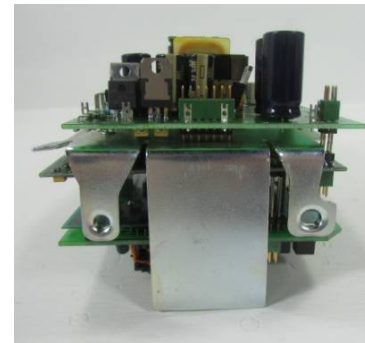
mobile...  
vehicular...  
urban...  
equipment...  
**edge/fog to cloud**



[Generic die bonder, source: Whatech]



[Flow measurement, source: Endress+Hauser]



## Structured literature overview

- <http://continuum.research-output.org/> (67 papers)

# Continuum Computing

## Resource heterogeneity

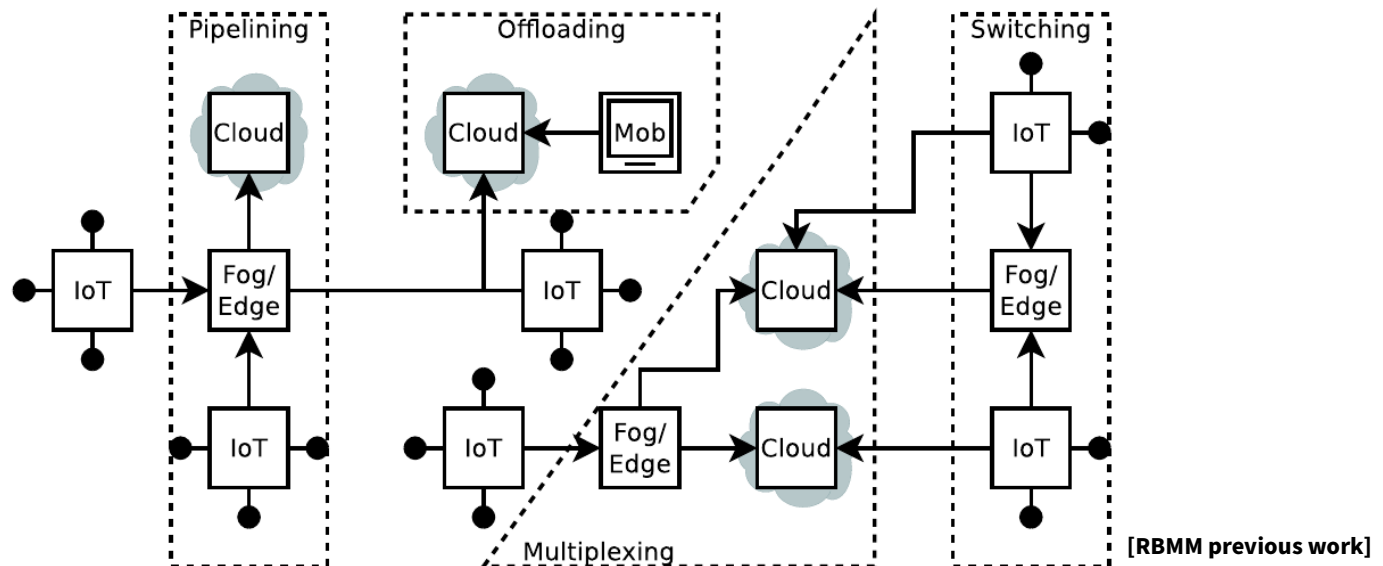
- Hardware architectures
- Hardware capabilities + capacities
- OS/Hypervisor
- Runtimes

CPU, GPU types  
GHz, GB, IOPS  
Nested virt, cgroups  
Container, V8/WASM, ...

X

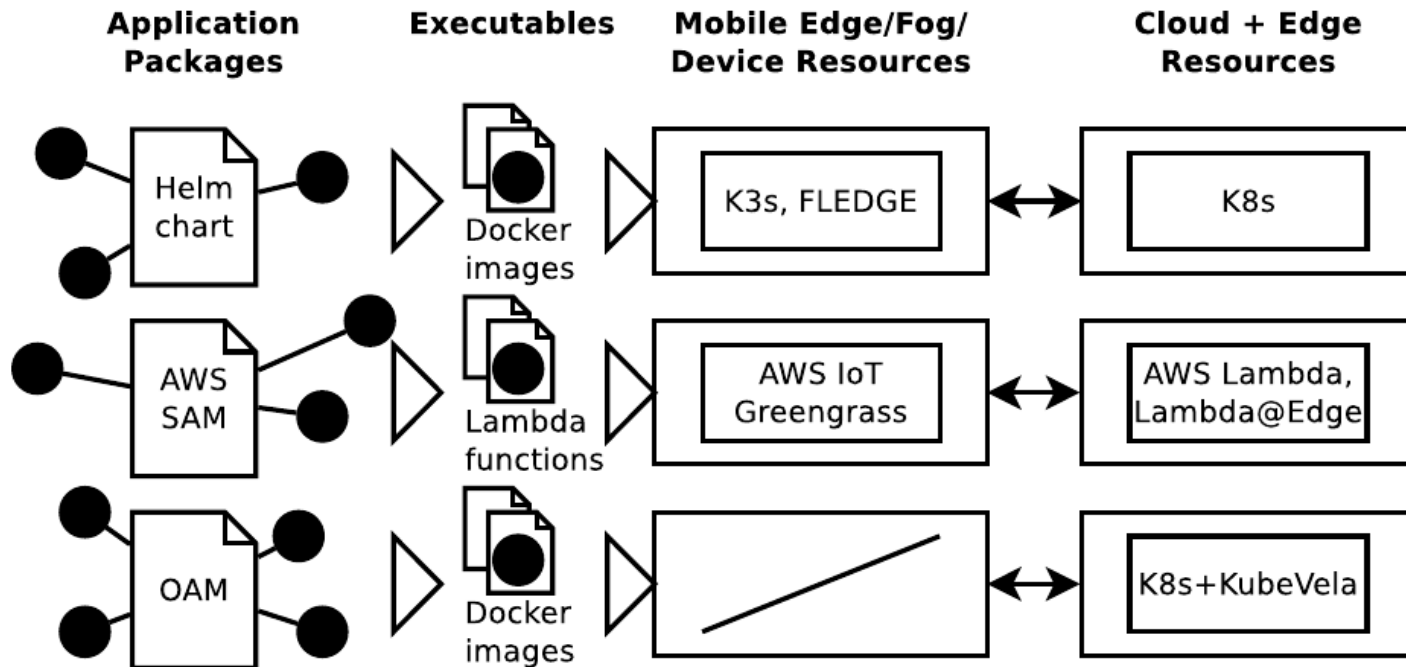
## Application modularity

- micro-/nanoservices composition + (user) requirements



# Methodology

## Industry-adopted continuum technologies



**Step 1:**  
Technology Overview  
& Planning Tool Creation

Helm, SAM, TOSCA, OAM  
*[sorry no CAMEL studied yet]*



**Step 2:**  
Empirical Evaluation  
& Feasibility Analysis

Helm, SAM



**Step 3:**  
Performance & Scalability  
Evaluation

Helm

# Deployer Design

## Workflow state chart

→ ask for choices on demand

- in: application/resource DSL
- out: runtime DSL + resource allocation
- solver

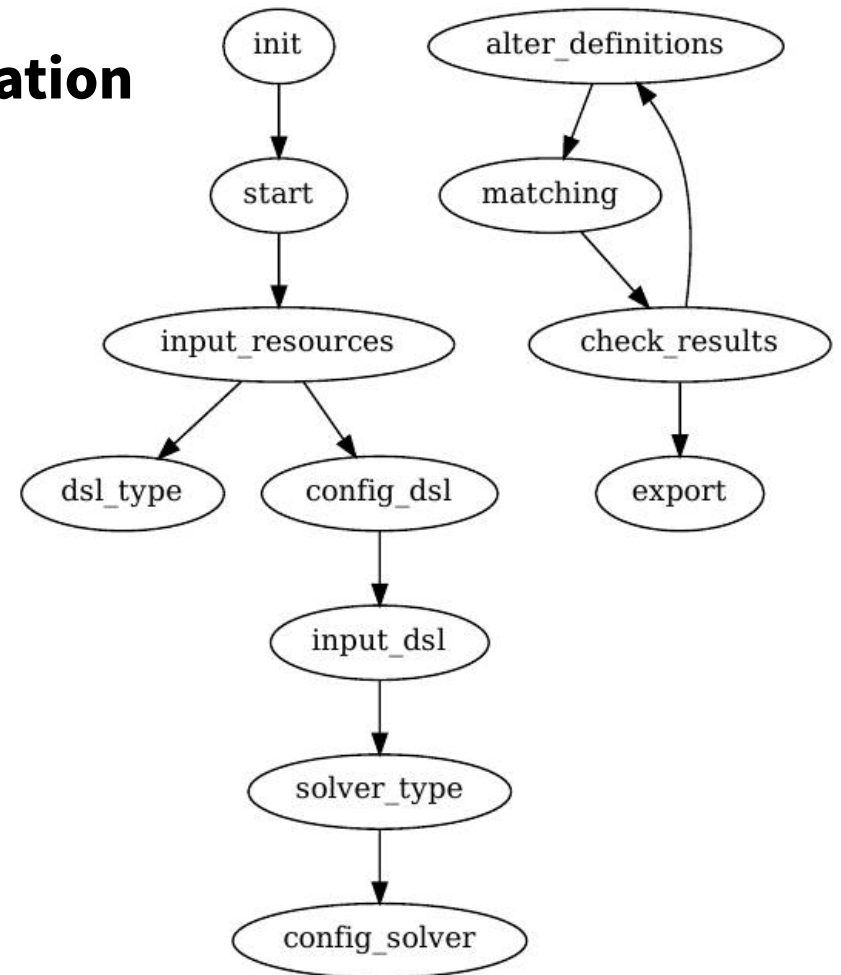
- composition information
  - e.g. helm.tgz, k8s.yaml
- resources.yaml
- user requirements on optimality

## Interactive use cases

- education
- exploration
- troubleshooting
- manual optimisation

## Batch use cases

- DevOps, system automation
- combinatorial optimisation
- systematic regression testing



# Implementation

- interactive mode
- batch mode
- partial automation

```
CPU | | 0%
RAM | | 0%
DEPLOYMENTS:
LABEL:
-----
Name: node-3
CPU: 4   MEMORY: 4096 MB
CPU | | 0%
RAM | | 0%
DEPLOYMENTS:
LABEL: cloud:public
-----

Enter DSL type: helm

Parsing DSL -
[Warning] No resource request provided for module RELEASE-NAME-mariadb. This can result in suboptimal deployment placement.
Parsing DSL \

Parsed workloads:

Name: RELEASE-NAME-wordpress
CPU: 0.3   MEMORY: 536 MB
LABEL:
-----
Name: RELEASE-NAME-mariadb
CPU: 0   MEMORY: 0 MB
LABEL:
-----

Choose a solver for the workload placement:
[0] Greedy Solver (sorts workloads and fills targets in a greedy fashion)
[1] SAT Solver (offers various options for mathematical optimal placements)

Enter Solver type: |
```

# Solvers

## On resources + preference labels

### Greedy

- single-target (CPU, mem)
- may get stuck

### SAT (Google OR tools)

- multi-target
- optimal solution

### User-defined extensibility

- plugin interface

```
Do you want to start matching? (y/n) y
status: LIKELY-FEASIBLE
CpSolverResponse:
status: OPTIMAL
objective: 13800
best_bound: 13800
booleans: 3
conflicts: 0
branches: 8
propagations: 7
integer_propagations: 17
walltime: 0.000558093
usertime: 0.000558032
deterministic_time: 3.883e-06
primal_integral: 0
```

#### Matching results:

##### Name: node-1

CPU: 2 MEMORY: 8192 MB

CPU		0%
RAM		0%

##### DEPLOYMENTS:

RELEASE-NAME-mariadb, cpu=0, memory=0, label=[]

LABEL:

##### Name: node-2

CPU: 3 MEMORY: 2048 MB

CPU	###		10%
RAM	#####		26%

##### DEPLOYMENTS:

RELEASE-NAME-wordpress, cpu=0.3, memory=536, label=[]

LABEL:

##### Name: node-3

CPU: 4 MEMORY: 4096 MB

CPU			0%
RAM			0%

##### DEPLOYMENTS:

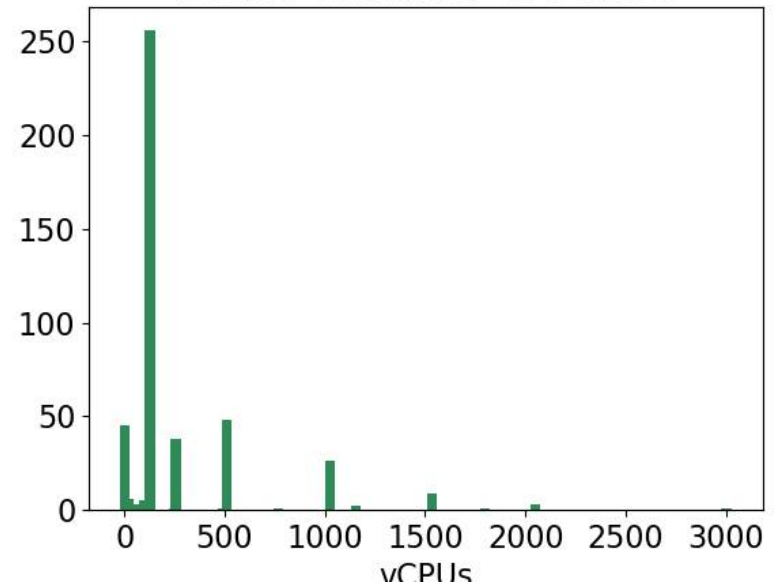
LABEL: cloud:public

# Applicability

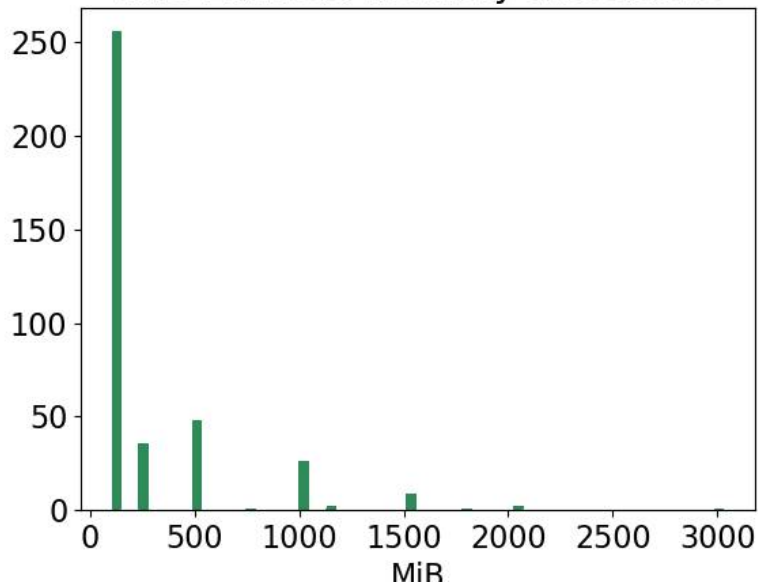
**Are cloud/continuum applications today suitable candidates?**

**→ Study of samples: AWS SAR [63% of Lambdas RCed], KubeApps Hub (now Artifacthub) [25% of Docker Images RCed]**

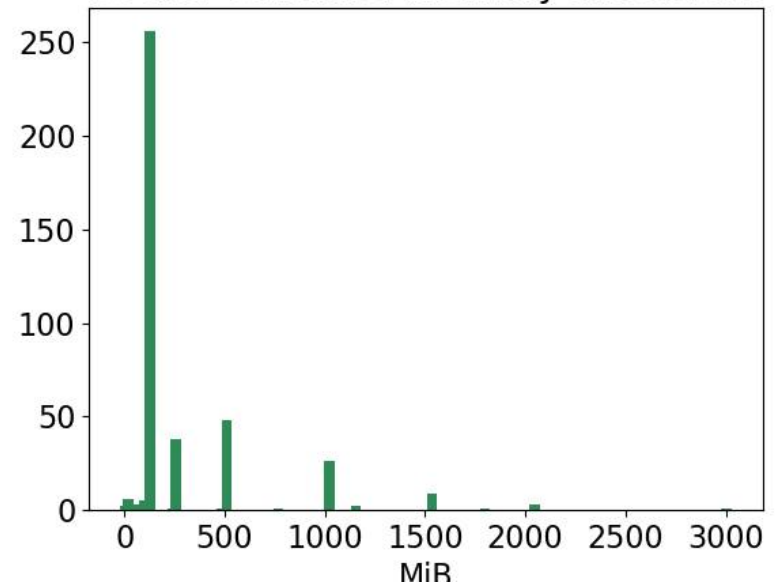
Docker container CPU slices



AWS Lambda memory allocations



Docker container memory allocations



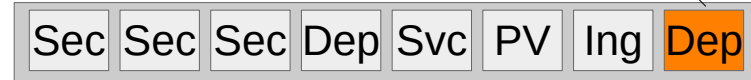


# Solver Performance

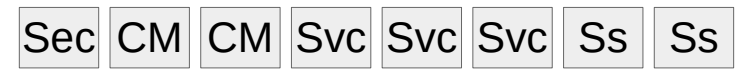
Is the performance suitable for industrial needs?

Deploying Helm charts

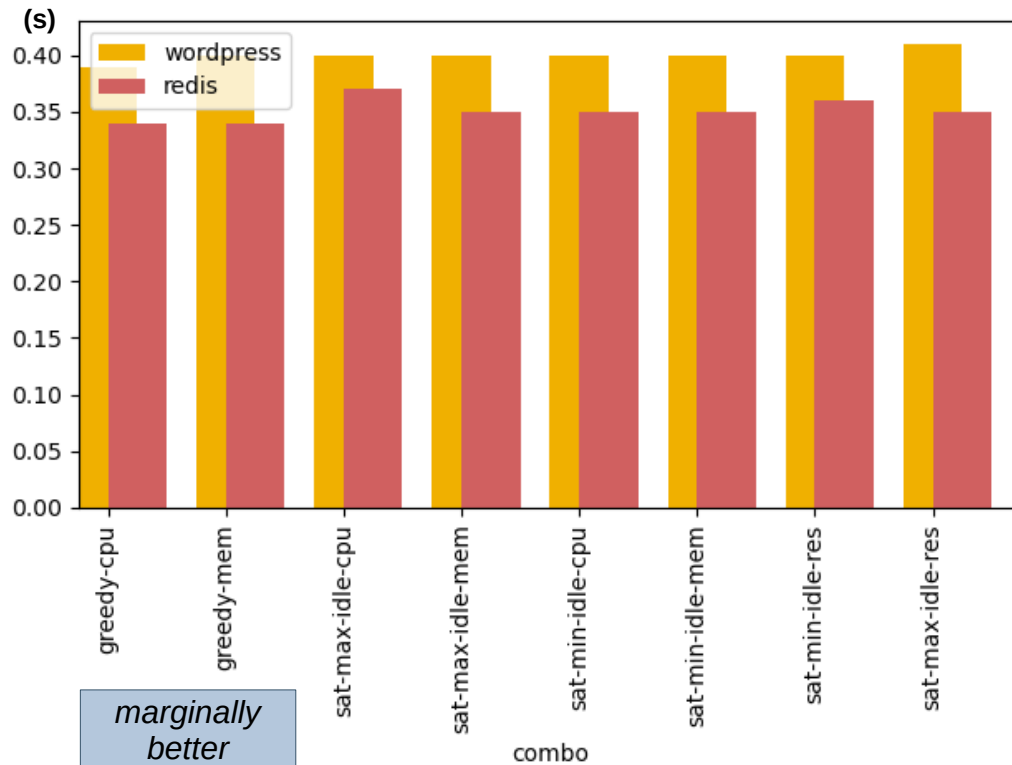
Wordpress



Redis



MariaDB

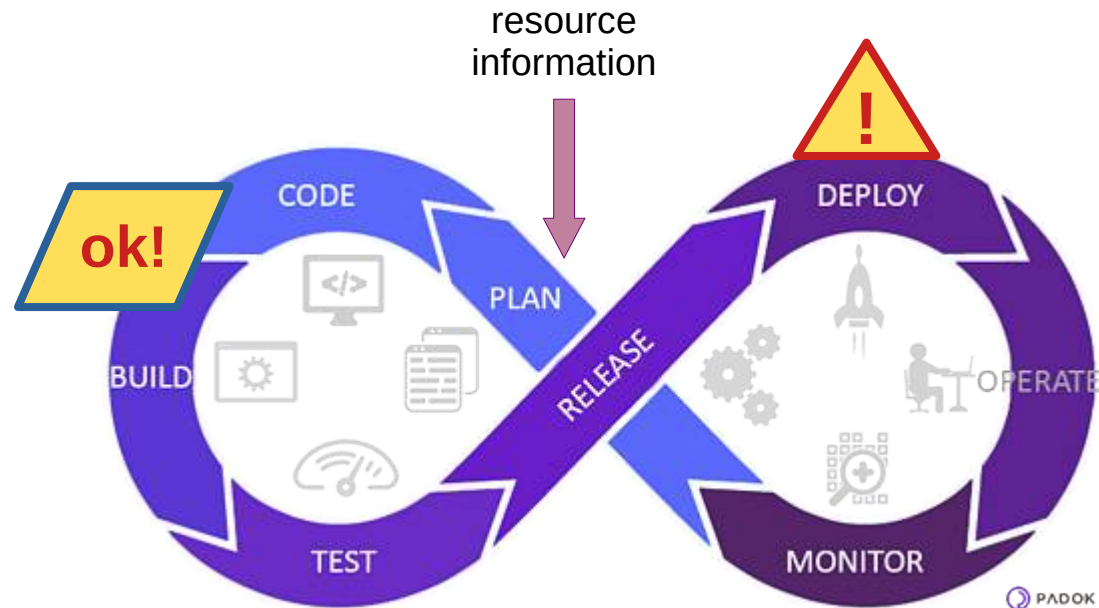


consistently  
< 1/2s  
  
= CI/CD  
capable

marginally better

# Conclusion

## Time-saving software engineering + DevOps for continuums



### Open source tool

- <https://doi.org/10.5281/zenodo.4584220>