

A photograph of three people (two women and one man) looking at a laptop screen in a dimly lit room. The image is overlaid with a semi-transparent red filter. The text 'Beck et al.' is written in white in the top right corner.

**Beck
et al.**

Beck et al. @RNUG

Minikube - get HCL Connections in the smallest possible setup

Munich, December 2020 – Martin Schmidt

Beck et al.

Current Positioning

A professional, experienced consultancy and Digital Transformation service provider with a continuous start-up mentality.

Enable Digital Work
Human-centric transformation

Employees become corporate digital citizens



work.
together.

Form Data to Results
Data-driven transformation

Big Data becomes business value

Shape a flexible IT
Flexible & agile infrastructure & operations

Save corporate IT departments from becoming obsolete

Delivery models / Revenue streams

Projects based on T&M or fixed service fees
Managed services based on monthly service fees
Subscription based services
Provisions from software license sales & cloud commissions.

Services are delivered from our virtual teams from Germany, Romania & Brazil

Our Business Practices



Enable Digital Work

Creating the conditions for digital work, providing platforms, promoting sustainable use. Change Leadership, accompanying, organizational development.



Form Data to Results

Support data-driven decisions, AI-based analytics in the area of IT data and unstructured content AI driven solutions; social media analytics.

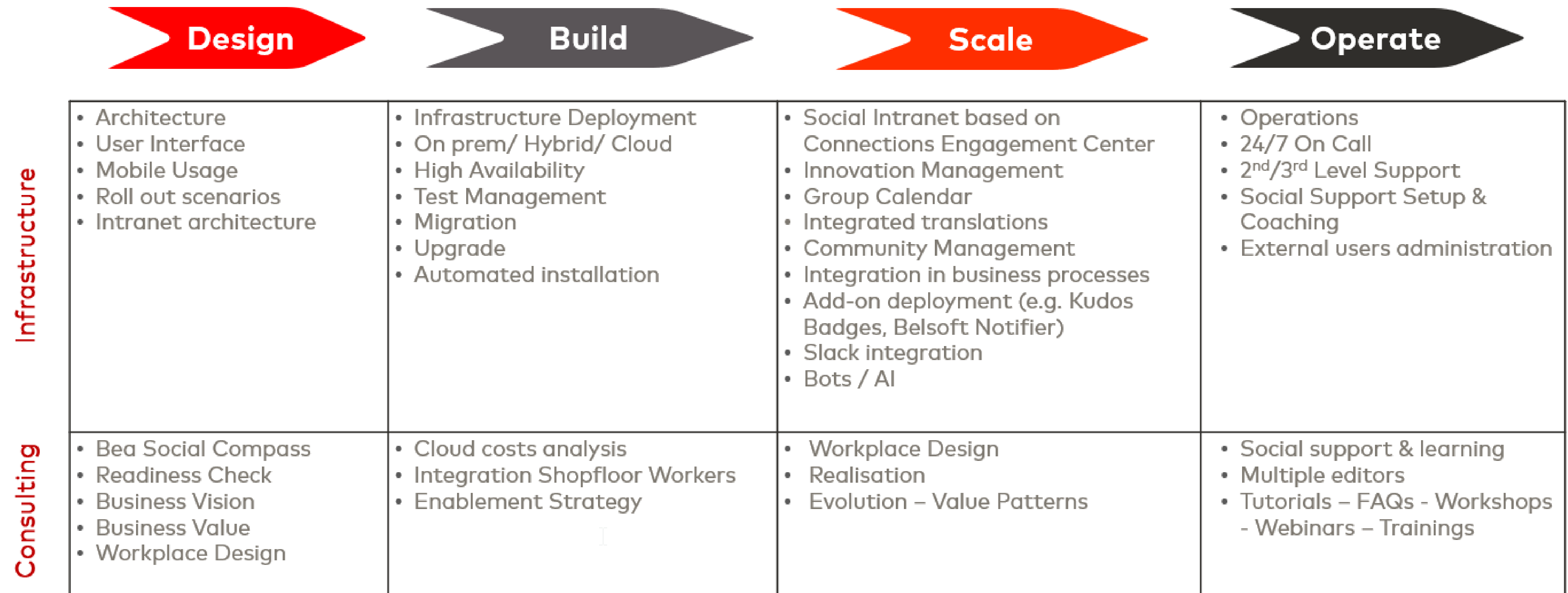


Shape a Flexible IT

We help corporate IT organizations transform their own IT into flexible, adaptive and robust IT services based on modern hybrid cloud environments.

Beck et al.'s HCL Connections Offering

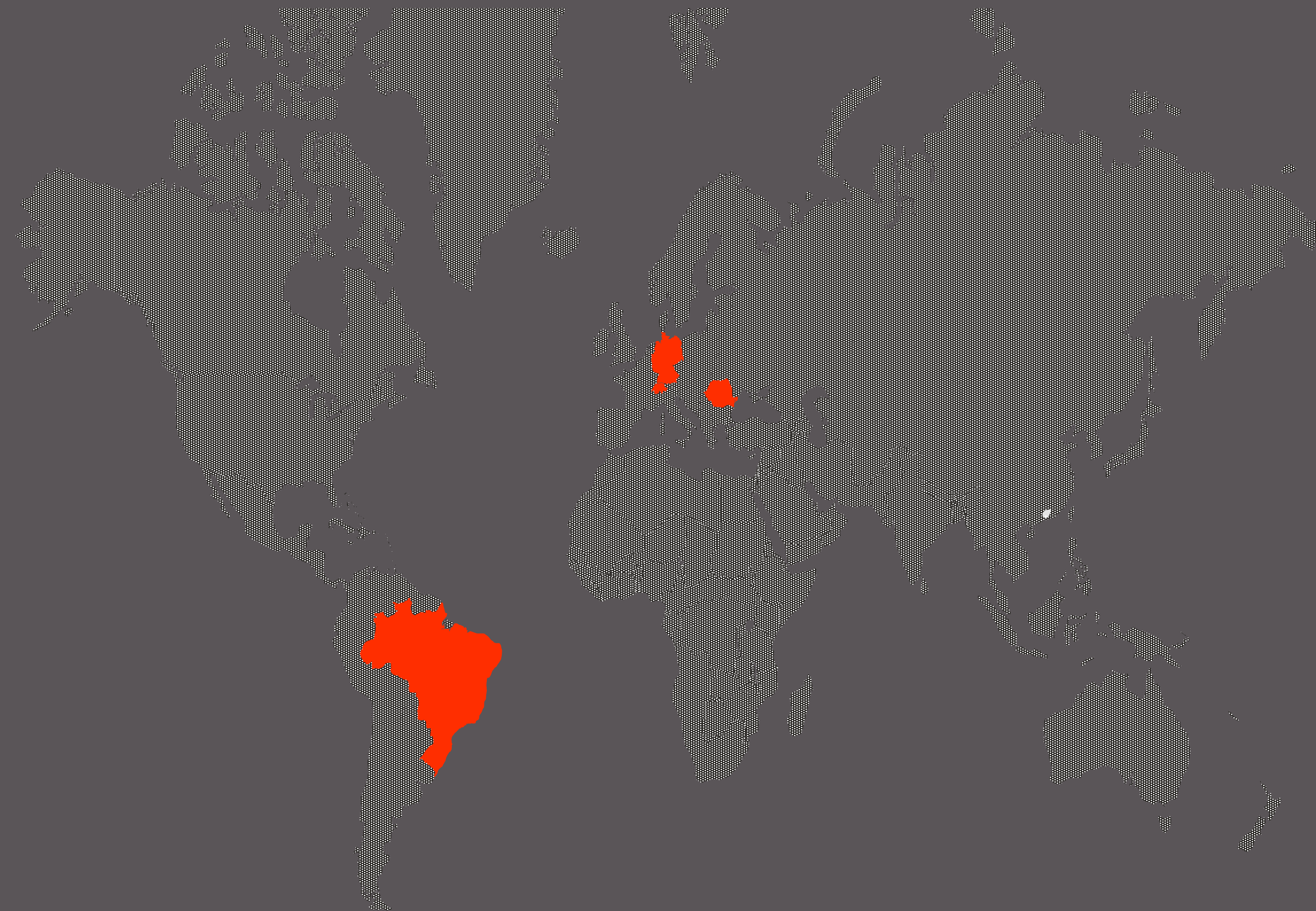
We cover the complete supply chain of HCL Connections



Facts & Figures

| | |
|-----------------------------------|--|
| Locations | Munich (DE), Cluj (RO), Florianópolis (BR) |
| Employees | Total 103; DE 51 - RO 35 - BR 17 (Headcounts) |
| Certified Technicians | 75% of all employees |
| Shareholder | Since spring 2020 Beck et al. together with binary, direktgruppe and infoWAN have been forming a new Digital Champion for D/A/CH. |
| Revenue DE | 2018: € 9 Mio 2019: € 12 Mio |
| Revenue Share | 45% recurring revenue, 45% projects, 10% licenses/ provisions |
| Equity (DE/Group) | € 1.2 Mio/ 1.5 Mio (as of 1.1.2018) |
| Relevant customer segments | Automotive (Dräxlmaier, TRATON, Continental, Magna, Infineon,) Travel & Transport (Lufthansa Group, Amadeus) „Hidden champions“ (KWS SAAT SE, pfm medical, Viridium, ...) Public clients (BMVG, German Forces, EU agencies) |
| Technology partnerships | Microsoft, AWS, HCL, OKTA, Nexthink, CloudHealth, Splunk, Beezy, MangoApps, Coyo, ... |

Beck et al. Subsidiaries



Beck et al. Romania, Cluj-Napoca, Transylvania
„Silicon Valley of Eastern Europe“
Set-up 2006; currently 35 employees (headcount).
Second largest university city in Romania
Very stable team with a strong management team

Beck et al. Brazil: Florianópolis
”Silicon Island of Latin America”
Set-up 2008; currently 17 employees (headcount).
Young, stable team with strong technical
background

Beck et al. Switzerland: Urdorf, Zürich area
representation only, no operational office

Amazon EC2

Amazon Elastic Container Registry

Amazon Elastic Container Service

AWS Lambda

Amazon Virtual Private Cloud

AWS Elastic Beanstalk

Auto Scaling

Elastic Load Balancing

Amazon S3

Amazon EBS

Amazon Glacier

AWS Storage Gateway

Amazon RDS

Amazon DynamoDB

Amazon ElastiCache

Amazon Redshift

Amazon CloudFront

AWS Direct Connect

AWS CodeDeploy

Amazon CloudWatch

Amazon EC2 Systems Manager

Amazon CloudFormation

AWS CloudTrail

AWS Config

AWS Partnership.



AWS Competency Program

- Microsoft Workloads on AWS
- DevOps Competency (coming soon)

AWS Advanced Consulting Partner

- AWS customer since 2008
- AWS Advanced Partner since 2013
- Founded in 1999
- Offices in Germany, Romania, Switzerland and Brazil
- 100+ employees

Service Delivery Program

- Certified Amazon EC2 for Windows Server Partner
- FSx for Windows File Server



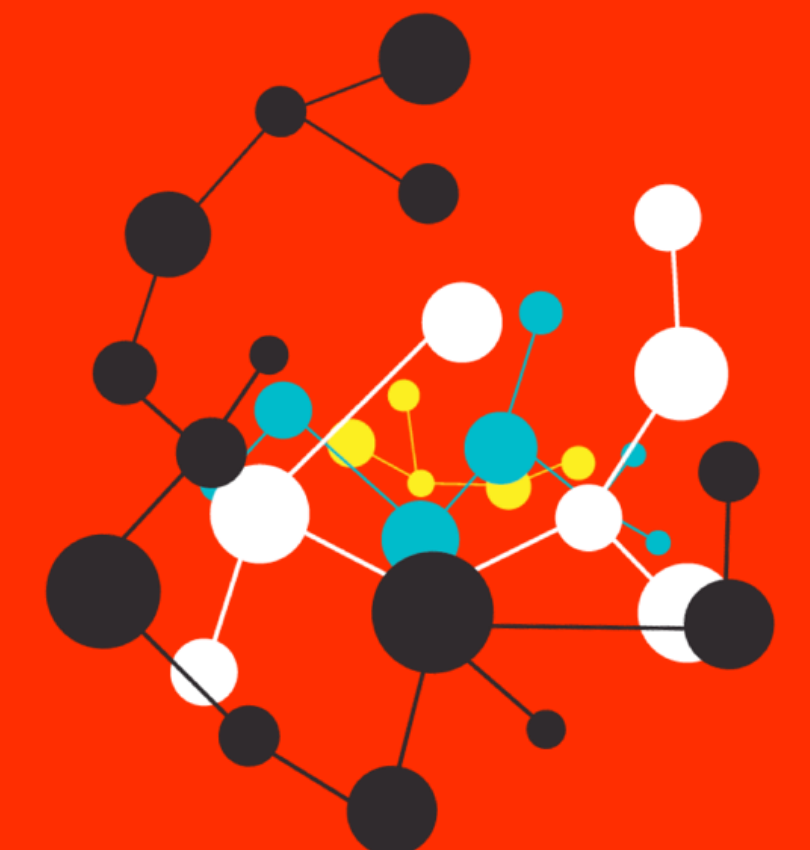
What we do

Beck et al. provides cloud consulting services ranging from strategy, design, build, implementation to managed services.

Our team of experienced certified experts is a trusted partner for all cloud initiatives and scenarios. Our focus is to hand over individual cloud solutions that will best solve our customers actual business challenges.

References

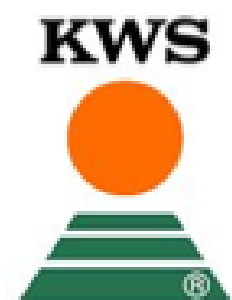
Fritz Dräxlmaier GmbH & Co. KG
 KWS SAAT SE
 European Food Safety Authority (EFSA)
 AXINOM GmbH
 ARDEX GmbH
 Infoscreen GmbH



Some of our technology partners.



Related customers.



Minikube - get HCL Connections in the smallest possible setup

10.12.2020 on RNUG online forum 2020



Martin Schmidt



Martin Schmidt

Senior IT Architect

IBM / HCL Connections since 2007

Experience in

- Deployment, Migration, Operation and Customization
- Kubernetes EKS and AWS, Container

Focusing on

- enhancing with own containers
- make the most out of customizer

More and more

- DevOps, Automation

+49 89 538863 878

Martin.Schmidt@becketal.com

linkedin.com/in/
martin-schmidt-526267153

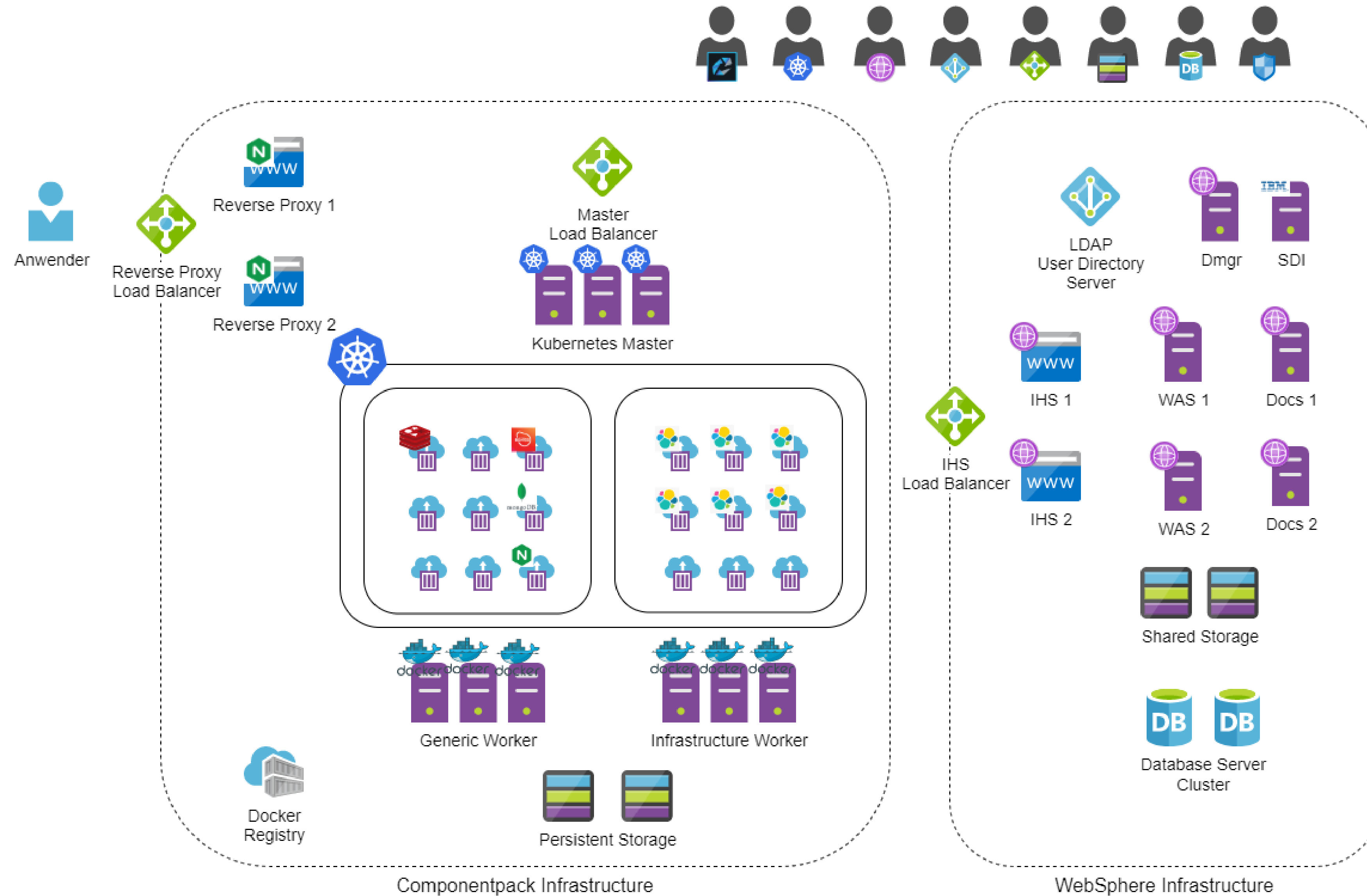
@MSSputnic

Minikube - get HCL Connections in the smallest possible setup

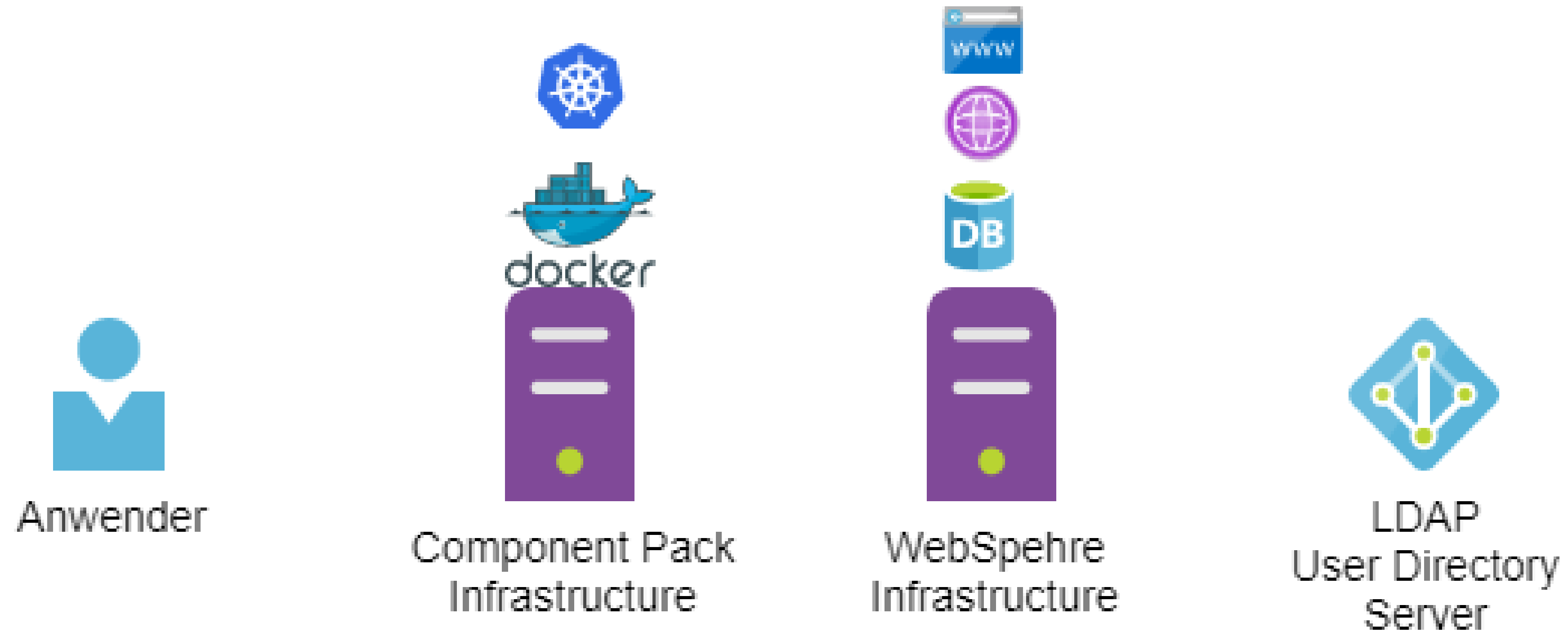
We will look at

- Infrastructure
- Sizing calculation
- Minikube
- Installation & Configuration
- Elasticsearch

Full Redundant Connections Infrastructure



Proof of Concept Infrastructure – Minimum requirements



Sizing your machine

Source: https://help.hcltechsw.com/connections/v65/admin/install/cp_install_sizing_for_production.html

| Container | Count | Limits CPU | Limits RAM | Required CPU | Required RAM | Requests CPU | Requests RAM | Min CPU | Min RAM |
|------------------------|-------|------------|-------------------|--------------|--------------|--------------|---------------------|-------------|-------------|
| analysisservice | 1 | 0,5 | 1024 | 0,5 | 1024 | 0,05 | 100 | 0,05 | 100 |
| appregistry-client | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| appregistry-service | 1 | 0,5 | 500 | 0,5 | 500 | 0,1 | 150 | 0,1 | 150 |
| cnx-ingress-controller | 1 | 0,5 | 512 | 0,5 | 512 | 0,02 | 64 | 0,02 | 64 |
| community-suggestions | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| haproxy | 1 | 0,5 | 200 | 0,5 | 200 | 0,05 | 50 | 0,05 | 50 |
| indexingservice | 1 | 0,5 | 1024 | 0,5 | 1024 | 0,2 | 100 | 0,2 | 100 |
| itm-services | 1 | 1 | 500 | 1 | 500 | 0,1 | 75 | 0,1 | 75 |
| mail-service | 0 | 0,5 | 500 | 0 | 0 | 0,05 | 75 | 0 | 0 |
| middleware-graphql | 1 | 1 | 500 | 1 | 500 | 0,1 | 75 | 0,1 | 75 |
| mw-proxy | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| orient-web-client | 1 | 1 | 1024 | 1 | 1024 | 0,1 | 75 | 0,1 | 75 |
| people-idmapping | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| people-migrate | 1 | 1 | 1024 | 1 | 1024 | 0,1 | 75 | 0,1 | 75 |
| people-relation | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| people-scoring | 1 | 0,5 | 1500 | 0,5 | 1500 | 0,05 | 75 | 0,05 | 75 |
| retrieval-service | 1 | 0,5 | 1024 | 0,5 | 1024 | 0,2 | 100 | 0,2 | 100 |
| userprefs-service | 1 | 0,5 | 400 | 0,5 | 400 | 0,05 | 75 | 0,05 | 75 |
| es-client | 1 | 2 | 2048 | 2 | 2048 | 0,1 | 1536 | 0,1 | 1536 |
| es-data | 1 | 2 | 4096 | 2 | 4096 | 0,5 | 3072 | 0,5 | 3072 |
| es-master | 1 | 1 | 1024 | 1 | 1024 | 0,1 | 768 | 0,1 | 768 |
| filebeat | 0 | 2 | 2048 | 0 | 0 | 0,5 | 512 | 0 | 0 |
| kibana | 0 | 3 | 4096 | 0 | 0 | 1 | 1024 | 0 | 0 |
| logstash | 0 | 3 | 8192 | 0 | 0 | 0,5 | 400 | 0 | 0 |
| mongo | 1 | 2 | 3096 | 2 | 3096 | 0,1 | 100 | 0,1 | 100 |
| redis-sentinel | 1 | 0,5 | 100 | 0,5 | 100 | 0,01 | 50 | 0,01 | 50 |
| redis-server | 1 | 1 | 1024 | 1 | 1024 | 0,05 | 75 | 0,05 | 75 |
| sanity | 1 | 0,1 | 512 | 0,1 | 512 | 0,1 | 128 | 0,1 | 128 |
| sanity-watcher | 1 | 0,5 | 100 | 0,5 | 100 | 0,01 | 50 | 0,01 | 50 |
| solr | 0 | 2 | 4096 | 0 | 0 | 0,02 | 600 | 0 | 0 |
| zookeeper | 0 | 0,5 | 400 | 0 | 0 | 0,01 | 300 | 0 | 0 |
| | | | Limits Sum | 19,6 | 23232 | | Requests Sum | 2,34 | 7168 |

| | CPU | RAM (MB) |
|---------|-----|----------|
| large | 2 | 8096 |
| xlarge | 4 | 16384 |
| 2xlarge | 8 | 32768 |
| 4xlarge | 16 | 65536 |

Minikube

Source: <https://minikube.sigs.k8s.io/docs/start/>

minikube start

minikube is local Kubernetes, focusing on making it easy to learn and develop for Kubernetes.

All you need is Docker (or similarly compatible) container or a Virtual Machine environment, and Kubernetes is a single command away: `minikube start`

What you'll need (Minimum Requirement)

- 2 CPUs or more
- 2GB of free memory
- 20GB of free disk space
- Internet connection
- Container or virtual machine manager, such as: Docker, Hyperkit, Hyper-V, KVM, Parallels, Podman, VirtualBox, or VMWare

=> For HCL Connections Component Pack we will use: t3a.xlarge with 50GB HDD

Run minikube for Connections on CentOS7

Source: <https://becketalservices.github.io/beas-cnx-cloud/minikube/chapter1.html>

```
# prepare CentOS 7
sudo yum -y install epel-release
sudo yum -y update
sudo yum -y install socat vim nano zip unzip bind-utils git
git clone https://github.com/becketalservices/beas-cnx-cloud.git
# helm
curl -L -O "https://get.helm.sh/helm-v2.17.0-linux-amd64.tar.gz"
tar -zxvf helm*
sudo mv $HOME/linux-amd64/helm /usr/bin/helm
helm version --client
# docker
sudo bash $HOME/beas-cnx-cloud/Azure/scripts/install_docker.sh
sudo usermod -a -G docker $USER # User must logoff / login to see effect!
sudo docker version
# minikube
curl -Lo minikube https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
chmod +x minikube
sudo mv minikube /usr/bin/
sudo minikube start --vm-driver=none --kubernetes-version v1.17.6
sudo systemctl enable kubelet.service
sudo minikube addons enable dashboard
# make minikube available for current user
sudo cp -r /root/.kube $HOME
sudo cp -r /root/.minikube $HOME
sudo chown -R $USER $HOME/.kube $HOME/.minikube
sed -i "s@/root@$HOME@" .kube/config
sed -i "s@/root@$HOME@" .minikube/machines/minikube/config.json
alias kubect1="minikube kubectl --"
echo 'alias kubect1="minikube kubectl --"' >> ~/.bashrc
minikube status
kubectl get nodes
```

Initialize helm and create docker registry

Source: <https://becketalservices.github.io/beas-cnx-cloud/minikube/chapter2.html>

```
## Configure Helm
# Create rbac configuration for helm
minikube kubectl -- apply -f beas-cnx-cloud/Azure/helm/rbac-config.yaml

# Initialize helm and deploy server side tiller component
helm init --service-account tiller

## Create your docker registry
# 1. Create a self signed certificate
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /tmp/tls.key -out /tmp/tls.crt -subj "/CN=control-plane.minikube.internal/O=DockerRegistry/C=XX"

# 2. Trust your new certificate
sudo cp /tmp/tls.crt /etc/pki/ca-trust/source/anchors/
sudo update-ca-trust

# 3. Restart Docker daemon to pick up this trust
sudo systemctl restart docker
sleep 60 # just to wait until docker and kubernetes is available again

# 4. Store the certificate inside Kubenetes
minikube kubectl -- create secret tls dr-secret --key /tmp/tls.key --cert /tmp/tls.crt

# 5. Deploy Docker Registry
helm install stable/docker-registry --set tlsSecretName=dr-secret --set service.type=NodePort \
  --set service.nodePort=31456 --set persistence.enabled=true \
  --set persistence.size=10G --set persistence.storageClass=standard

# 6. Check that you can access your Registry via curl
# !! curl should trust the certificate. No -k option necessary.
curl -v https://control-plane.minikube.internal:31456
```


Install Component Pack services

Prepare configuration file installsettings.sh

Source: <https://becketalservices.github.io/beas-cnx-cloud/minikube/chapter3.html>

```
# Write our environment settings
cat > ~/installsettings.sh <<EOF
# used connections version
installversion=65
installsubversion=10

CNXNS=connections
CNXSize=small # small -> run only 1 replica per pod

# Storage settings (minikube uses 'standard' by default)
storageclass=standard

# ES settings
useStandaloneES=0
useSolr=0

# Docker Registry
ECRRegistry=control-plane.minikube.internal:31456

# Component Pack
GlobalDomainName="$(hostname -d)"
ic_admin_user="admin_user"
ic_admin_password='admin_password'
ic_internal="ic_internal"
ic_front_door="ic_front_door"
master_ip="$HOSTNAME"
# for test environments with just one node or no taint nodes, set to false.
nodeAffinityRequired=false

EOF
```

Install Component Pack services

Create basic configuration

Source: <https://becketalservices.github.io/beas-cnx-cloud/minikube/chapter3.html>

```
## Download and extract CP
unzip ComponentPack*.zip

## Load overall configuration
. ~/installsettings.sh

## Write Config Files
bash beas-cnx-cloud/common/scripts/write_cp_config.sh

## Create namespace
kubectl create namespace $CNXNS

## To create all volumes on efs, you can use the generated
# install_cp.yaml configuration file:

helm upgrade connections-volumes \
  ~/beas-cnx-cloud/Azure/helm/connections-persistent-storage-nfs \
  -i -f ~/install_cp.yaml --namespace $CNXNS

## Upload Docker images to registry
pushd microservices_connections/hybridcloud/support
./setupImages.sh -dr ${ECRRegistry} -u dummy -p dummy
popd

# remove local docker images
docker rmi $(docker images -q)

# to force the deletion of the sanity images run
docker rmi -f $(docker images ${ECRRegistry}/connections/sanity -q)
```

Deploy Component Pack to Cluster

Source: <https://becketalservices.github.io/beas-cnx-cloud/minikube/chapter4.html>

1. Bootstrapping the Kubernetes cluster

```
helmchart=$(ls microservices_connections/hybridcloud/helmbuilds/bootstrap*)  
helm upgrade bootstrap $helmchart -i -f ~/install_cp.yaml --namespace $CNXNS
```

2. Installing the Component Pack's connections-env

```
helmchart=$(ls microservices_connections/hybridcloud/helmbuilds/connections-env*)  
helm upgrade connections-env $helmchart -i -f ~/install_cp.yaml --namespace $CNXNS
```

3. Installing the Component Pack infrastructure

```
helmchart=$(ls microservices_connections/hybridcloud/helmbuilds/infrastructure*)  
helm upgrade infrastructure $helmchart -i -f ~/install_cp.yaml --namespace $CNXNS
```

4. Installing Elasticsearch

```
helmchart=$(ls microservices_connections/hybridcloud/helmbuilds/elasticsearch*)  
helm upgrade elasticsearch $helmchart -i -f ~/install_cp.yaml --namespace $CNXNS
```

5. Installing Orient Me

```
helmchart=$(ls microservices_connections/hybridcloud/helmbuilds/orientme*)  
helm upgrade orientme $helmchart -i -f ~/install_cp.yaml --namespace $CNXNS
```

6. ... (Customizer, Ingress Controller, [Activities Plus])

=> You will see that due to the setting of „replicaCount: 1“ in the install_cp.yaml configuration file, each pod is started only once to save resources.

Reverse Proxy for Customizer

- Minikube has no LoadBalancer functionality
- Services on Minikube can not listen on port http (80) and port https (443) without opening configuration
- Simplest solution: Install nginx on minikube server

```
sudo yum -y install nginx
sudo systemctl start nginx
sudo systemctl enable nginx
```

```
# SELinux allow forwarding
sudo setsebool -P httpd_can_network_connect 1
```

```
sudo mkdir -p /etc/pki/nginx/private
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/pki/nginx/private/server.key -out /etc/pki/nginx/server.crt
```

```
sudo vim /etc/nginx/nginx.conf
```

➤ enable https section

➤ add here the required configuration according to

https://help.hcltechsw.com/connections/v65/admin/install/cp_config_customizer_setup_nginx.html

➤ Use <http://control-plane.minikube.internal:30301> as proxy_pass url

Integration with Connections on WebSphere

- No difference to the normal integration
see: Configuring the Component Pack
https://help.hcltechsw.com/connections/v65/admin/install/cp_config_intro.html

Elasticsearch management from command line

To see and manage the Elasticsearch indices from command line easily, some command line tools are provided in the git repository.

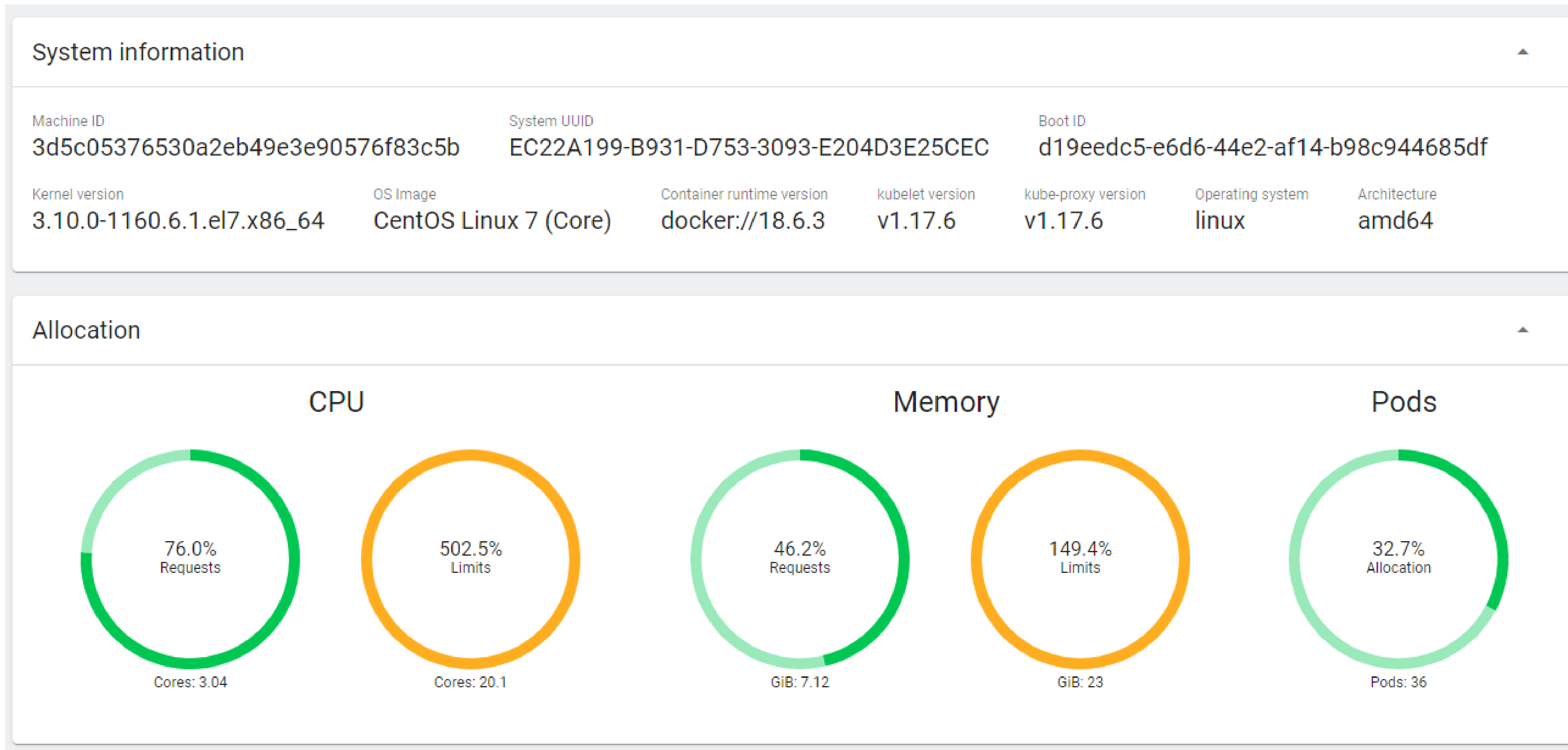
The tools can be found in `beas-cnx-cloud/elasticsearch/`

- Tool to extract the elastic search certificates from the component pack configuration
`./getcerts.sh`
- **List All Indexes**
`./esget.sh "_cat/indices?v"`
- **Remove replica from index to make it “green” on a 1 node cluster:**
`./esput.sh "orient-me-collection/_settings?pretty" rem_replica.json`
- **See** `commands.txt` for more

Kubernetes Dashboard

minikube kubectl -- proxy

<http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/http:kubernetes-dashboard:/proxy>



Sources and Resources

- Documentation HCL Component Pack on managed Kubernetes:
<https://becketalservices.github.io/beas-cnx-cloud/>
- Configuring additional HCL Connections applications
https://help.hcltechsw.com/connections/v65/admin/install/t_inst_config_addons.html

The way you get to the future is the future you get.

**Beck
et al.**



@beck_et_al



Martin.Schmidt@becketal.com