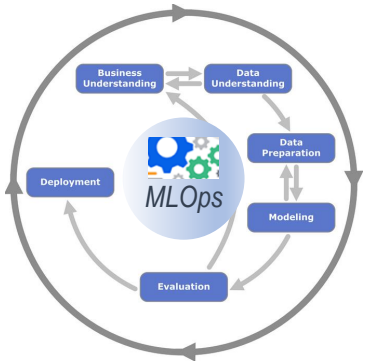
Towards an Enterprise grade Machine Learning pipeline with R

Contributions to whiteboxing machine learning for interoperation with production environments

Thomas.Strehl@at.ibm.com *Thomas.Weinrich@at.ibm.com* *Rudolf.Pailer@at.ibm.com* *@20200120*

REnterprise: Machine Learning with R in the Enterprise

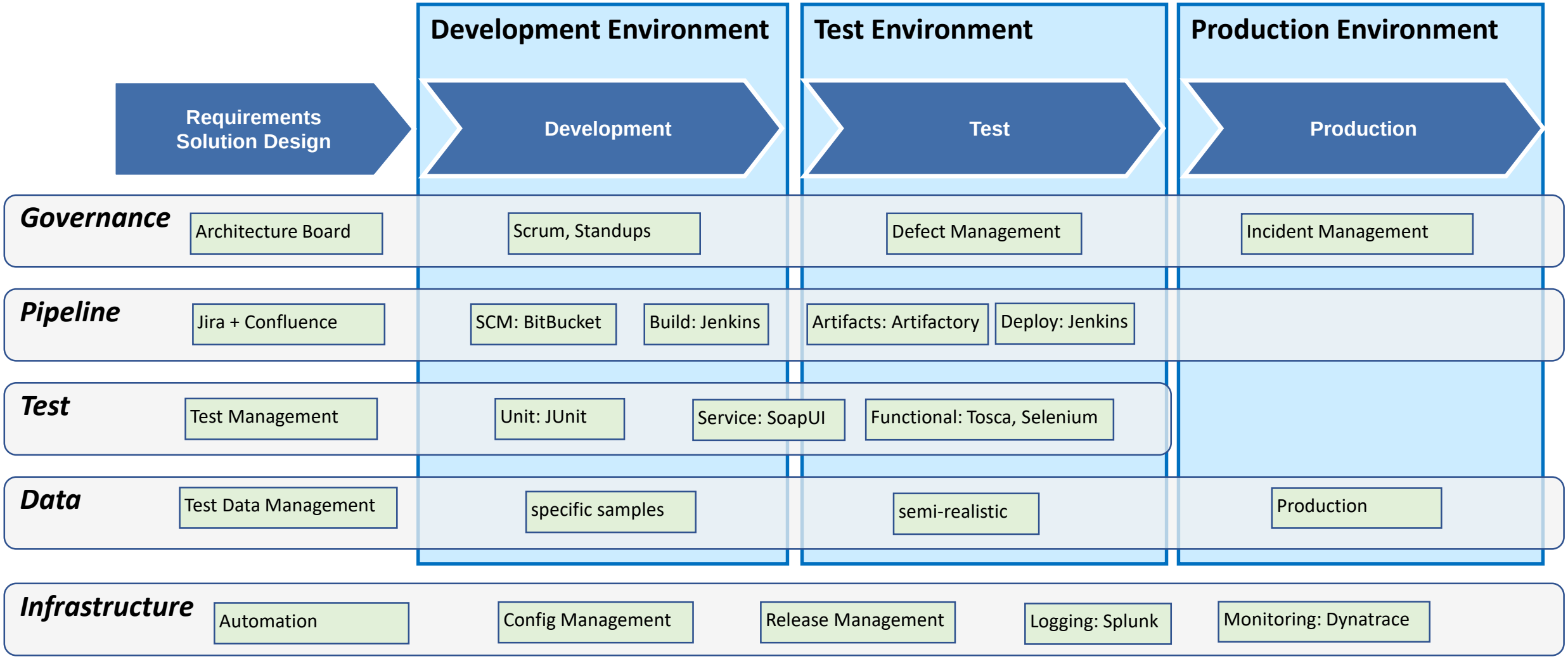


Whiteboxing ML for Interoperation with production environments

- Enterprise Environment
  - Processes, Governance, Architecture, Security
  - Requirements, Develop, Test, Release Management, Rollout
  - Documentation, Incident Management
- CRISP-DM + MLOps -> CrispML
  - Standard process for Data Mining related projects
  - DevOps Automation for Machine Learning
  - -> Service orientated Architecture for data preparation, training and scoring
- Demo: rep-admin + rep-crispml
  - CrispML demo implementation on kubernetes
  - Automated ML pipeline for R

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Classical Software Factory Lineup

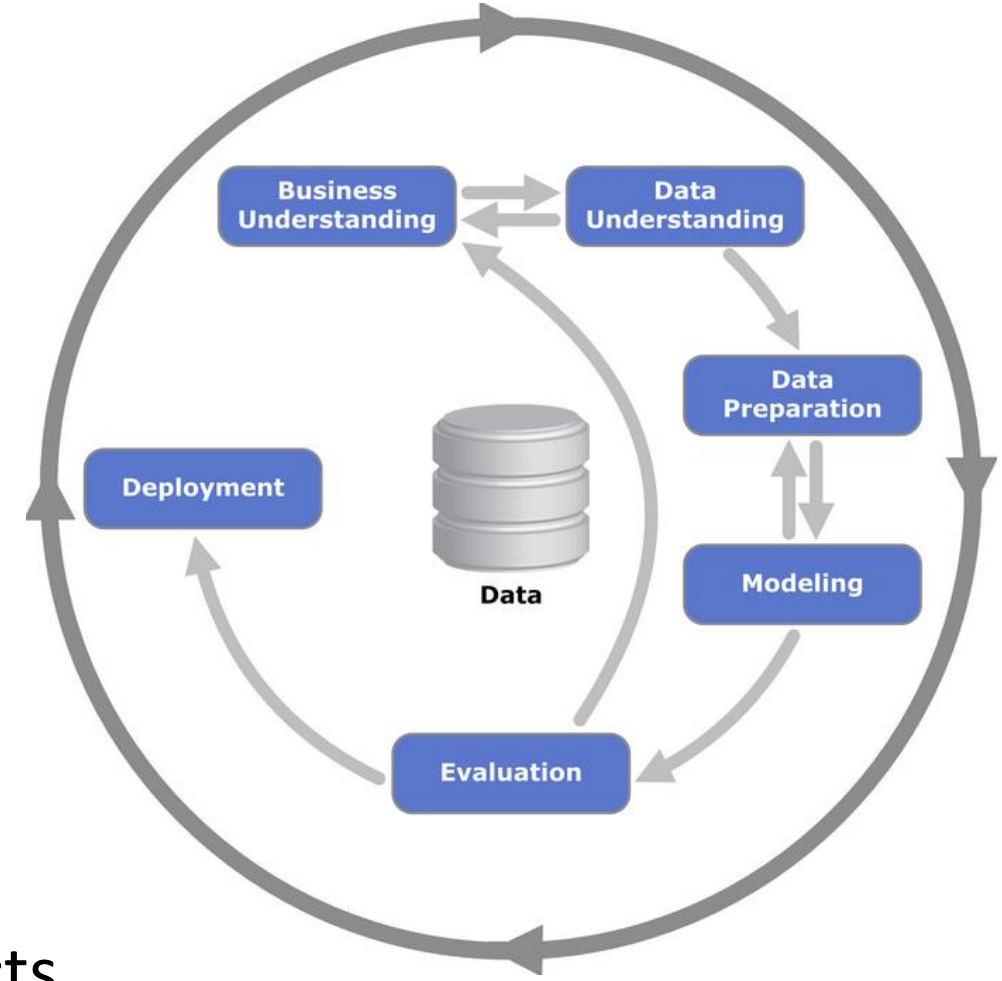


Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Classical Machine Learning Pipeline

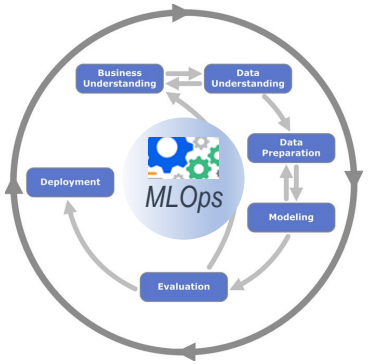
CRISP-DM (Cross InduStry Process for Data Mining)

- Business understanding
- Data understanding
- Data preparation
- Modeling
- Evaluation
- Deployment



- Devised in late 1990
- Used by around 45% of data projects

REnterprise: ML interaction with the Enterprise

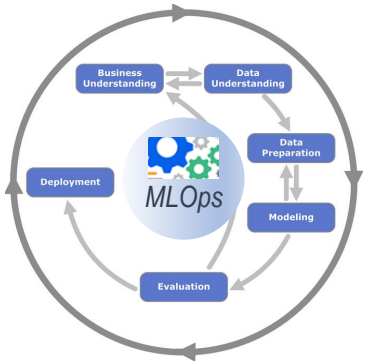


Interaction of Enterprise services with ML services

- Data Preperation + Training
  - Mass Data: FileSystem, DataBase, DWH, DataLake, Data Platform, …
- Scoring
  - Batch Scoring: e.g. Rscript, REST
  - Record Scoring: e.g REST
- Governance
  - Reporting, Statistics, Performance
  - Documentation, Changes, Defects, Incidents

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: CRISP-DM generic Interfaces

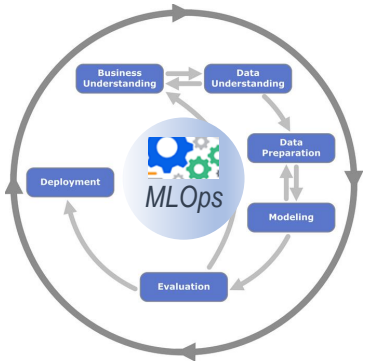


CrispML: Implementation of 9 methods for Training and Scoring

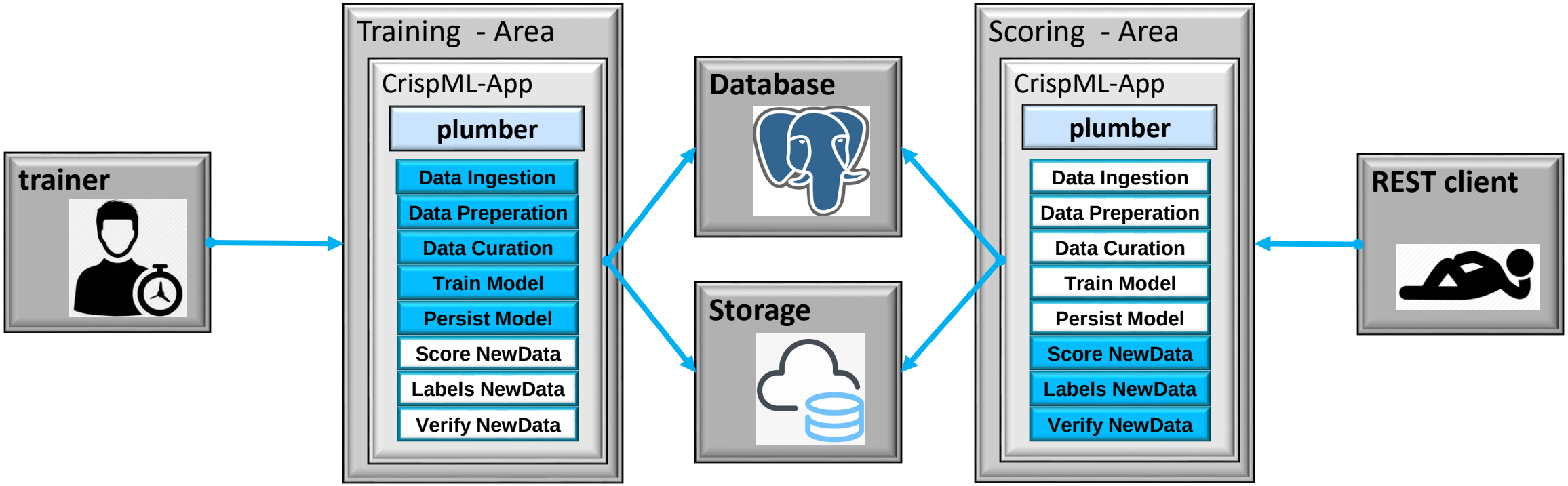
- Training and Scoring
  - DataIngestion: Raw Data
  - DataPreparation: Algorithm independent
  - DataCuration: Algorithm specific
- Training
  - ModelTraining: Algorithm, Hyperparameters
  - ModelReport: ML KPI's
  - ModelPersist: Model Registry
- Scoring
  - NewDataScore: persist each score with reference to metadata
  - NewDataLabel: import new ground truth
  - NewDataReport: verify new ground truth against persisted score

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: CrispML Components

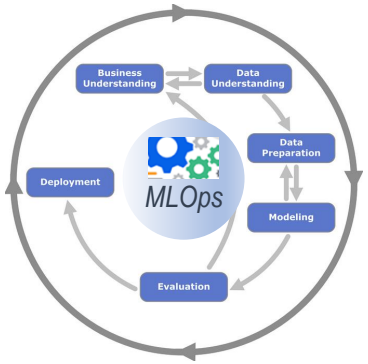


CrispML: Training and Scorings Servers and Clients



Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: CrispML Big Picture



Orchestration: *Run ML cycle including training and verification in place in each enviroment*

Deployment: *Stage ML functionality across environmnets*

All ML functionality in R package

Development Environment

- Data Ingestion
- Verify NewData
- Data Preperation
- Labels NewData
- Data Curation
- Score NewData
- Train Model
- Persist Model

GitLab

R-Package

- Data Ingestion
- Data Preperation
- Data Curation
- Train Model
- Persist Model
- Score NewData
- Labels NewData
- Verify NewData

Complete pipeline subject to QA

QA Environment

Training Environment

- Data Ingestion
- Data Preperation
- Data Curation
- Train Model
- Persist Model

Scoring Environment

- Score NewData
- Labels NewData
- Verify NewData

Fallback: Use model trained in QA

Production Environment

Training Environment

- Data Ingestion
- Data Preperation
- Data Curation
- Train Model
- Persist Model

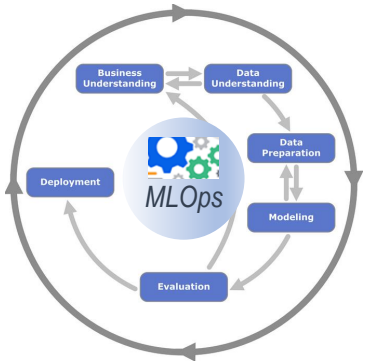
Scoring Environment

- Score NewData
- Labels NewData
- Verify NewData

Data: *Access data pool shared across environments (optional GDPR filters)*

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: CrispML implementation in R

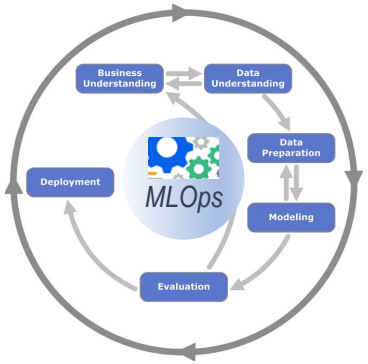


Crispml: Selfcontained, standalone, scalable REST Docker container

- CrispML
  - All CRISP-DM methods designed for remote invocation
  - REST interface to ingestion, training, scoring
  - Plumber (other options: openCPU, rserver)
- Admin Console
  - Lightweight demo implementation of remote control
  - Shiny GUI app
  - Challenge: no direct access to data, only via REST
- Runtime Environment
  - Linux (any R platform), Kubernetes, …

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Demo based on DevOps and Containers

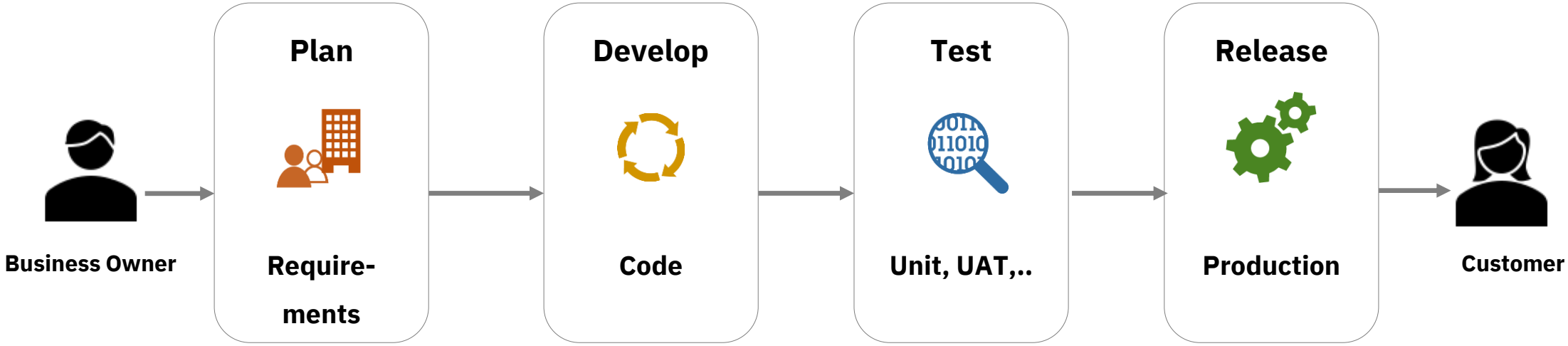


DevOps, Cloud

- ‘Traditional’ platform
  - E.g. Bitbucket, Jira, Confluence, Jenkins, Artifactory, VMware
- DevOps -> MLOps
  - Automatisierung der Pipeline: Gitlab, Tekton
- Containerized
  - Docker, Kubernetes
- IBM Cloud
  - Gitlab, Tekton, Kubernetes, logDNA, sysDIG, DB2

Thomas.strehl@at.ibm.com *Stand 20200120*

Why DevOps – Traditional software delivery lifecycle



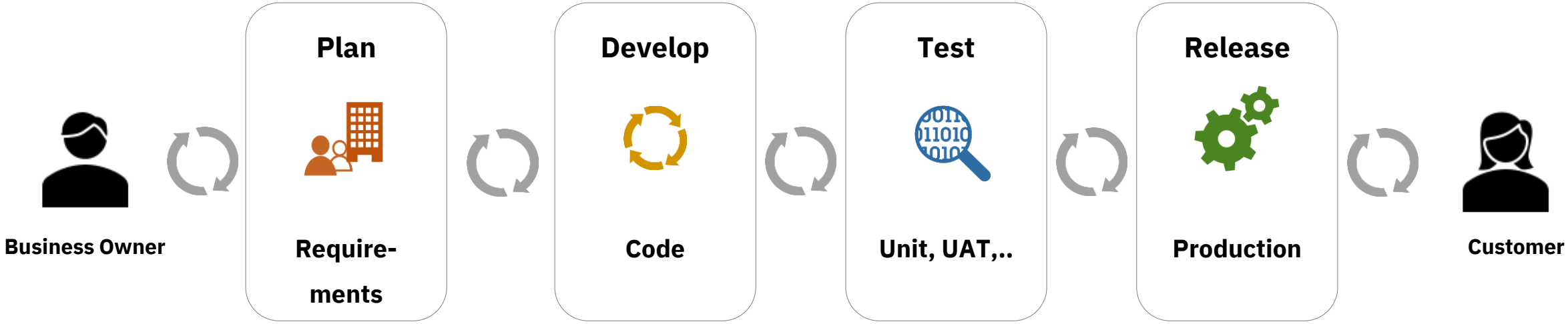
Failures due to inconsistent dev and production environments

Bottlenecks trying to deliver frequent releases to meet market demands

Complex, manual, processes for release lack repeatability and speed

Poor visibility into dependencies across releases, resources, and teams

Why DevOps – Transforming the software delivery lifecycle



Agility & Flexibility

Standardization

Fail fast & Fail early

Automation

DevOps: Continuous flow in Enterprise systems

3 DevOps dimensions

6 DevOps practice areas

4 DevOps software lifecycle

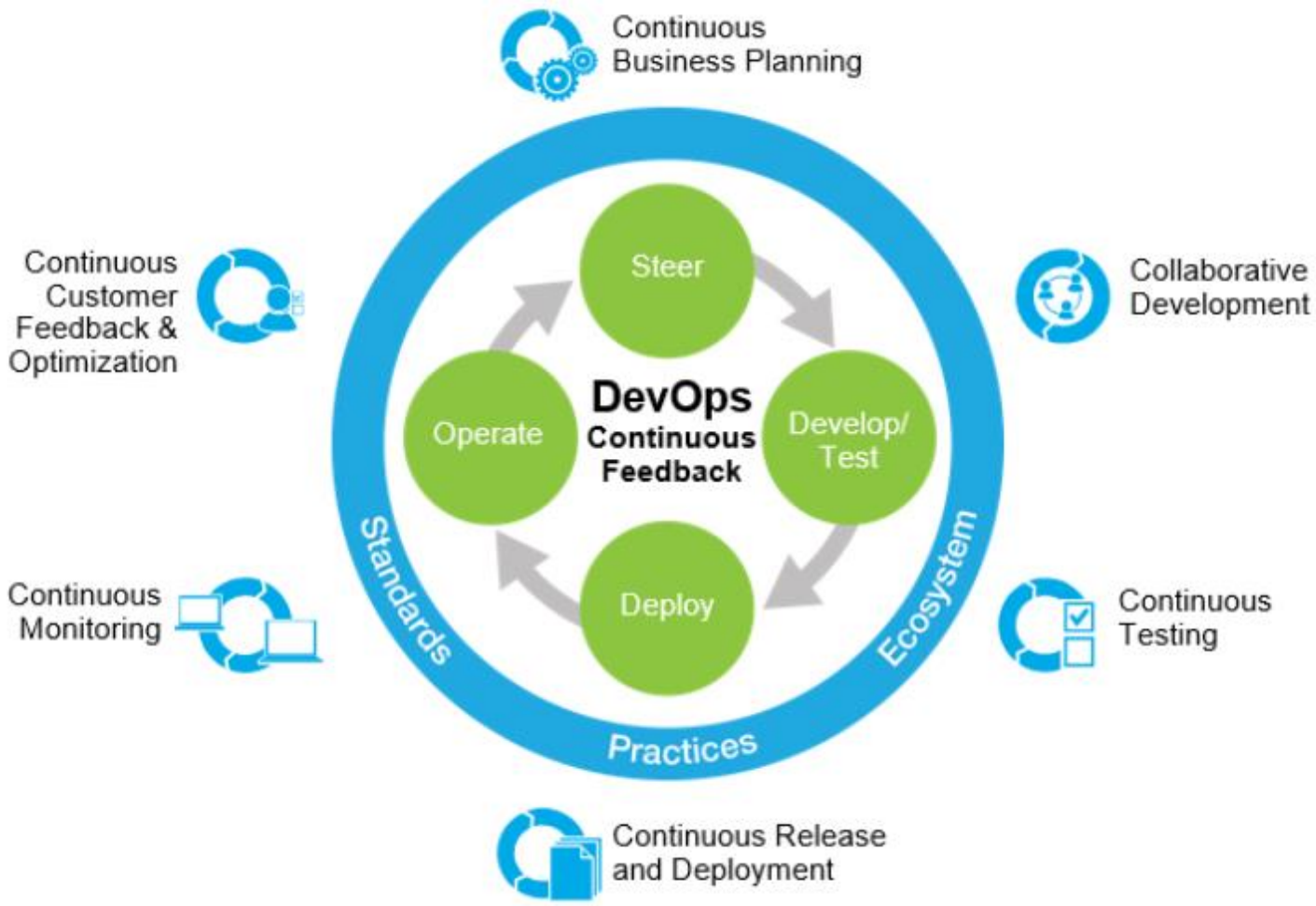
DevOps **Dimensions**

People

Processes

Technology

DevOps **Practice Areas**



DevOps Principles: Continuous everything

Collaboration for speed

- Collaborative steering
- Collaborative Dev-Ops
- Feedback loops

Dashboard everything

- Continuous monitoring
- Visibility to the teams

Automate everything

- Continuous Delivery
- Continuous Integration
- Infra as Code

Test everything

- Continuous testing
- Test automation

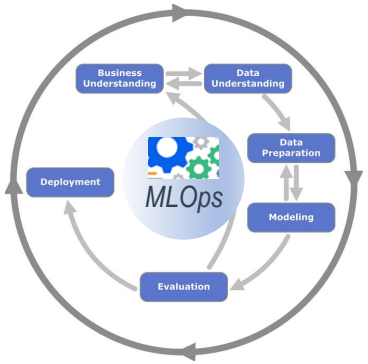
Monitor and audit everything

- Continuous monitoring
- Logging and monitoring

DevOps: Automation, automation, automation

- If someone has to do the same thing more than once, it’s a candidate for automation
- If something is hard, do it repeatedly
- Develop and Test against production-like systems
- Iterative and frequent deployments using repeatable and reliable processes
- Continuously monitor and validate operational quality characteristics
- Encourage a culture of experimentation and valuing team improvement
  - Minimizing business risk – fail small and fast
- All DevOps principles also apply to MLOps -> CrispMl approach

REnterprise: Containerizing: Docker + Kubernetes



CrispML:

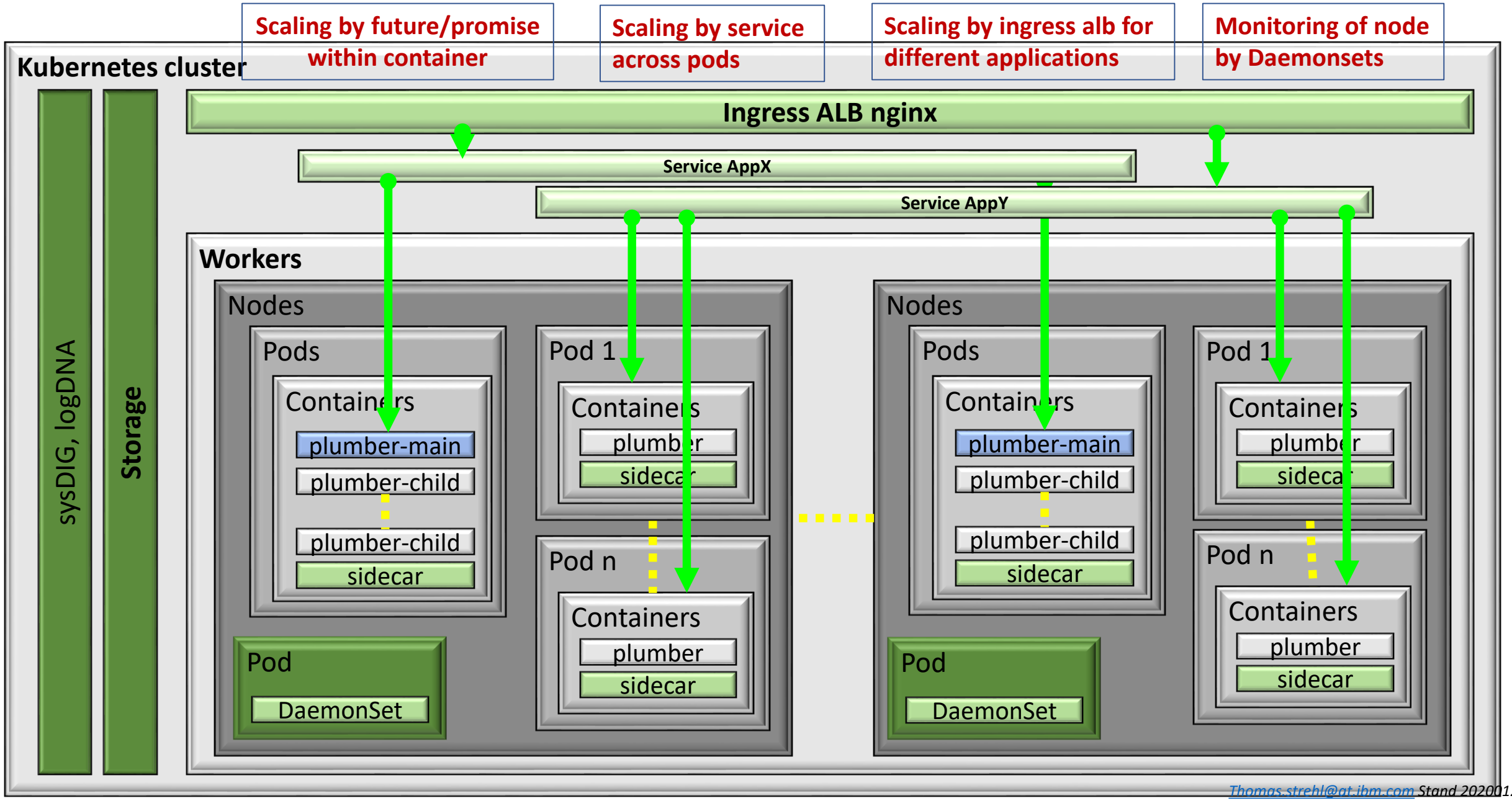
- Docker
  - Lightweight variant of virtual server
  - Start from downloadable template and enhance along ‘Dockerfile’
  - Persist as ‘image’ and instantiate as ‘container’
  - Template images available e.g. for ‘rshiny’ and ‘plumber’ applications

- Kubernetes
  - Orchestrator for containerized applications
  - Scaling, Loadbalancing, System Monitoring, Storage, Network, …

Thomas.strehl@at.ibm.com *Stand 20200120*

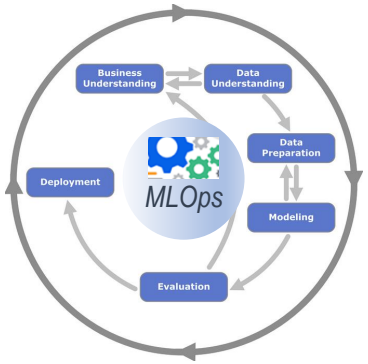
thomas.strehl@at.ibm.com @20200202

REnterprise: Scaling R on Kubernetes



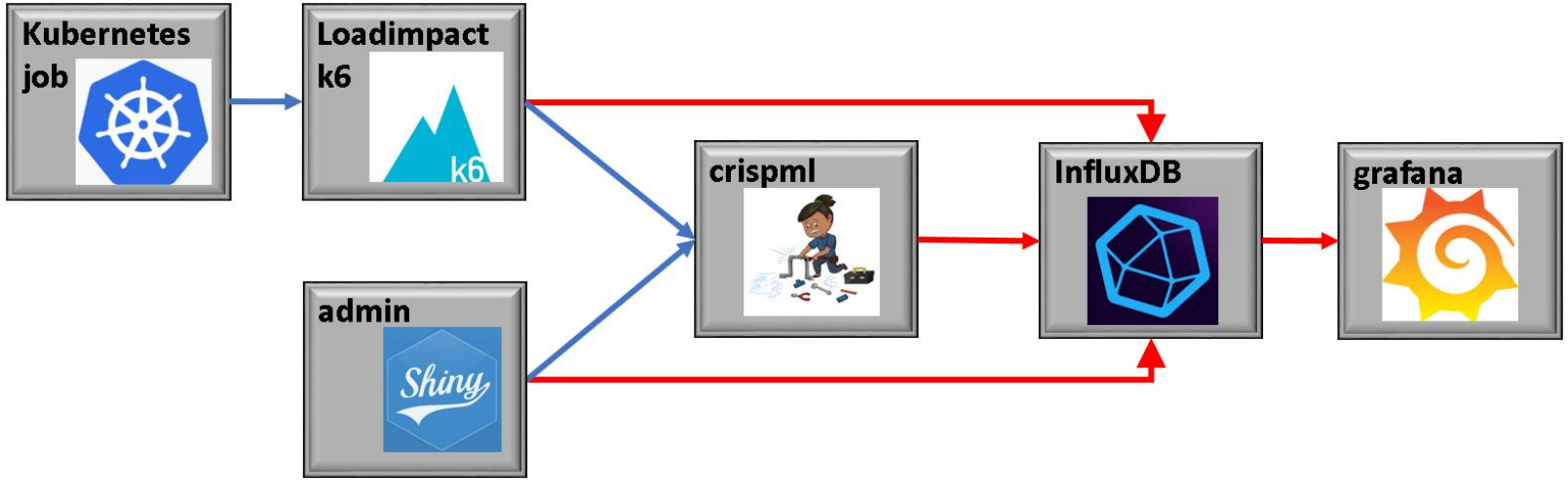
Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: Performance Testing



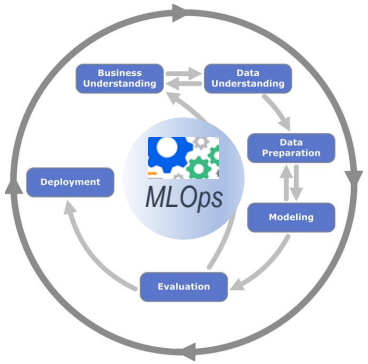
CrispML: loadimpact/k6 -> influxdb -> grafana

- Many options
  - jmeter, Locust(python), grinder(java), Gatling(scala), ……
- loadimpact/k6
  - java script, 3000+ stars on GitHub
  - Writes to influxdb, prebuilt Grafana dashboards, invoked as container



Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: Database and Persistent storage



CrispML: odbc, DBI, dbplyr -> DB2 (requires OS level driver)

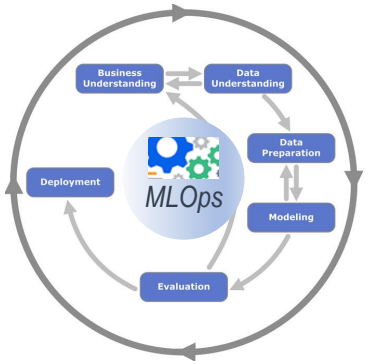
- Peformance sensitive
  - Throughput depends on network latency
- R-Packages
  - pool: DB connection pool
  - dbplyr: Execute dataframe operations in DB

CrispML: Kubernetes Persistent Volume Claim

- Kubernetes file system, DWH, Data Lake, Data Platform
  - Persist results (models, parameters, …)
  - Persist state across instances of R processes on different pod/nodes

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Application Logging



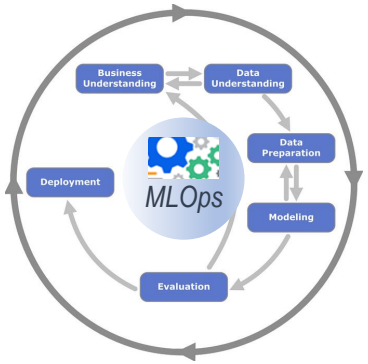
Crispml: file, stdout, stderr. InfluxdbR -> influxdb -> grafana

- Protocol
  - Flow of operation, quantitative messages (parameters, results, …)
- R-Packages
  - Rsyslog. Log4r, Logger, logging, lgr, futil.Logger, shinyEventLogger
- InfluxdbR
  - Read/write influxdb. Wrapper to IQL (influx query language)

- Kubernetes
  - stdout, stderr, /var/log/*.log -> elasticsearch, …
  - IBM Cloud: logDNA

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Application Monitoring



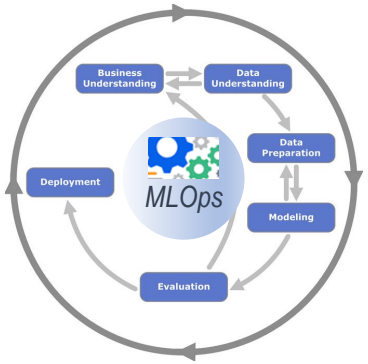
CrispML: entry/exit Log -> memory.profile() -> InfluxdbR -> influxdb -> grafana

- System resources as seen by application
  - Memory, gc activity, database connections, shiny sessions, plumber calls
- R-Packages
  - utils (gc, memory.profile), memuse(Sys.*); profmem, bench; hprof*
- gc(), memory.profile()
  - Slow (>150ms, >200ms)

- Kubernetes System Resource Monitoring:
  - prometheus - > grafana.
  - IBM Cloud: sysDIG

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: Demand handling and Build & Deploy



CrispML: GitLab -> (Epic) -> UserStoryTask -> Branch -> Merge -> Version

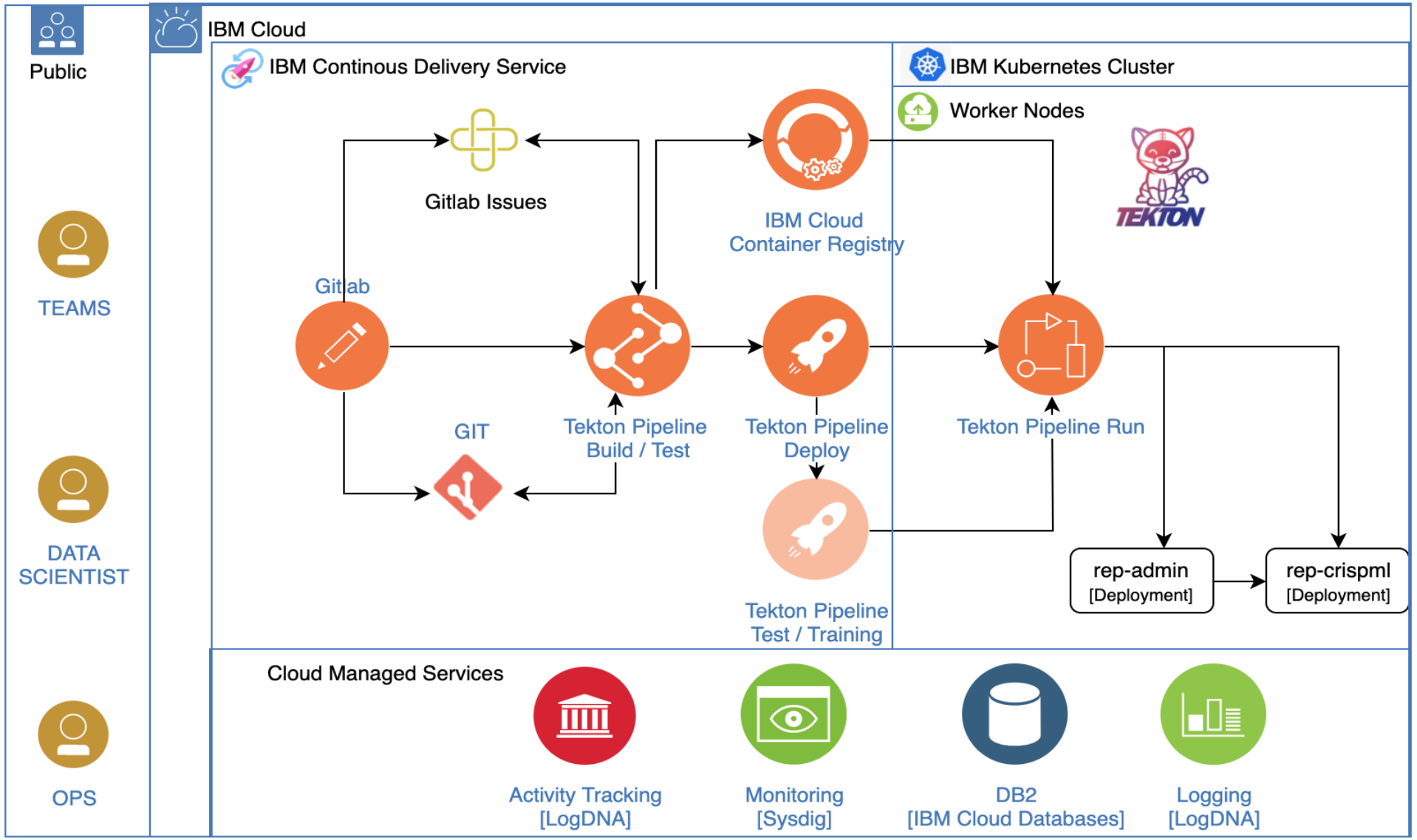
- Demand Handling

CrispML: GitLab -> Tekton -> Image Registry -> Kubernetes Deployment

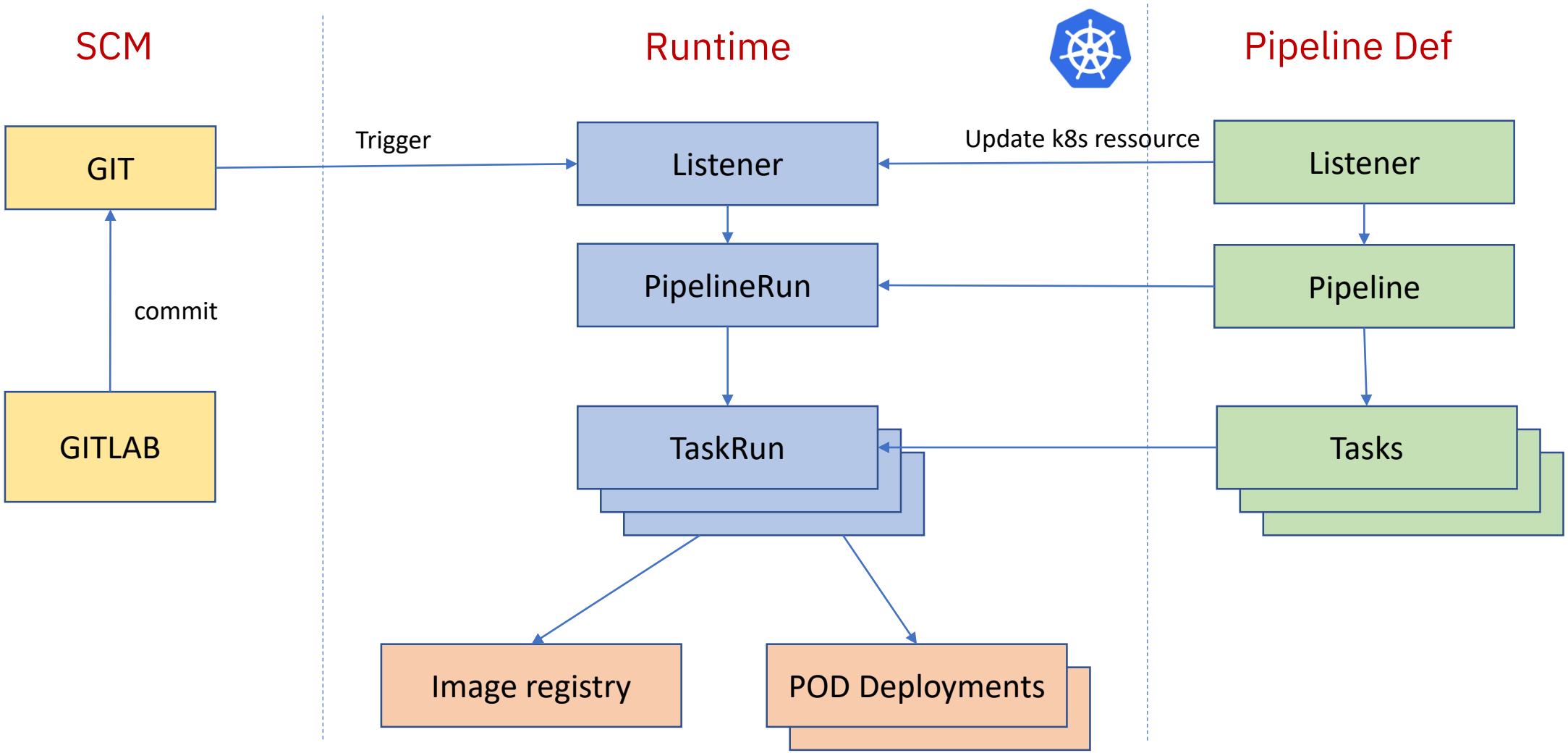
- Build and Deploy

Thomas.strehl@at.ibm.com *Stand 20200120*

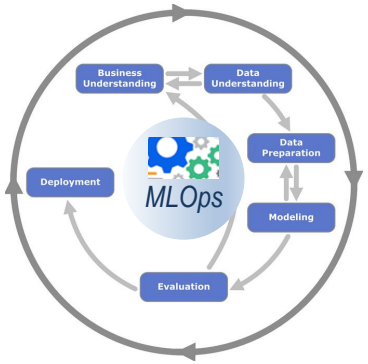
Setup showcase CrispML



CrispML build&delivery mit gitlab und Tekton Pipelines

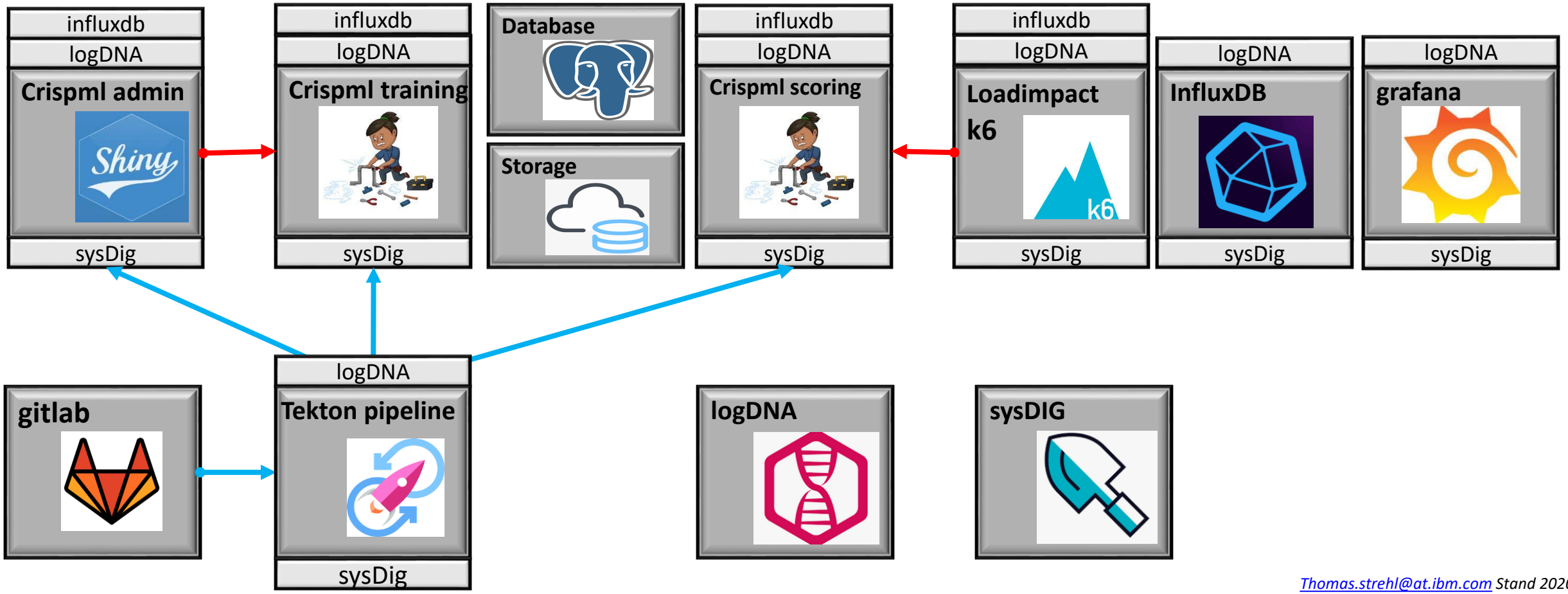


REnterprise: CrispML Kubernetes Services



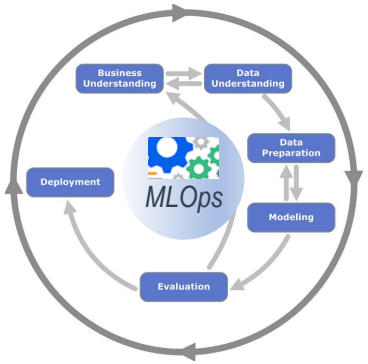
Deployed Containers: CrispML*, K6, InfluxDB, Grafana

IBM Cloud Services: logDNA, sysDIG, GitLab, Tekton, DB2, Storage



Thomas.strehl@at.ibm.com Stand 20200120

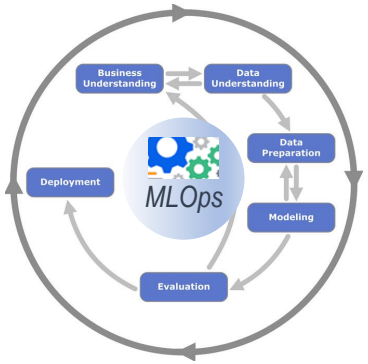
REnterprise: DEMO: CrispML, MLOps and K8S in action



- CrispML
  - R- Code: Plumber REST service and remote invocation by Rshiny
- MLOps
  - GitLab Repository, Issue Board, Commit in Rstudio
  - Tekton Trigger, build & deployment pipeline to kubernetes
- Loadimpact/k6 Performance Test
  - Docker file and Performance test script
  - Kubernetes invocation performance test job and scaling from 1 to 3 instances
  - logDNA application logs
  - sysDIG Kubernetes system Resources
  - influxDB + grafana: live response time report

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: CrispML - Shiny application



REnterprise - CrispML Console Live

Frontend to REST API of Training/Prediction Engine

ListDB2Tables

DataIngest

percentData

1

Modelling

Training Data: TimeFrame

1 6 12

1 3 5 7 9 11 12

DataPrepare

DataCurate

ModelTrain

Model

rf

Weight

100

ModelAssess

Prediction

Predict Data: TimeFrame

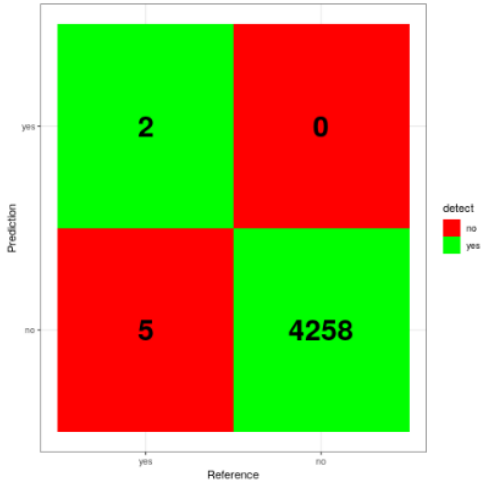
1 7 12

1 3 5 7 9 11 12

CrispML: Response from remote REST interfaces

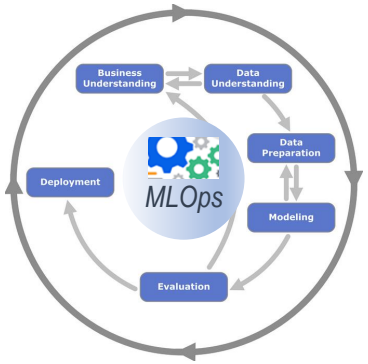
First select Tab, then Press corresponding Button, then Wait until Output appears on Tab

DB2Tables | DataIngest | DataPrepare | DataCurate | ModelTrain



Thomas.strehl@at.ibm.com *Stand 20200120*

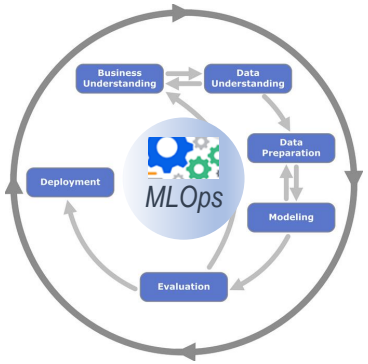
REnterprise: CrispML - Plumber REST service (train model)




```
# ------------------------------------------------------------------------------
# 1. repDataIngest: Read complete dataset in csv and write slized sample into DB2
# ------------------------------------------------------------------------------
#* Ingest and persist training data
#* @param slices Number of time slices (aka days) to create
#* @param percentData Percentage of data to retain for further processing
#* @json Render output to json
#* @get /crispml/repDataIngest
repDataIngest <- function(slices=12, percentData=0.01) {

  # log to stderr
  logMsg("repDataIngest entry")

  # perfLogEntry/perfLogExit: Performance data for InfluxDB: Read Testdata from zip file contained in docker image
  dfEntry <- perfLogEntry(method="repDataIngest", dataSet="read.csv", mlModel="creditcardfraud.zip")
  zipFile <- "../data/creditcardfraud.zip"
  df <- read.csv(unz(description=zipFile, filename="creditcard.csv"), sep=",")
  perfLogExit(dfEntry=dfEntry, db="CrispmlDB", measurement="Crispml")

  # Slice dataset into 12 equal sized packages thereby simulating 12 days
  nRecs <- dim(df)[1]
  Frame <- rep(1:12, each=floor(nRecs/12))
  df$Frame <- c(Frame, rep(slices,(nRecs-length(Frame))))

  # perfLogEntry/perfLogExit: Performance data for InfluxDB: Write sample of data to DB2
  dfEntry <- perfLogEntry(method="repDataIngest", dataSet="writeDB2", mlModel=toString(nRecs*as.numeric(percentData)))
  df <- df[sample(1:nRecs,floor(dim(df)[1]*as.numeric(percentData))),]
  dbCon <- repDB2Connect()
  # recreate table if already exists
  r <- DBI::dbWriteTable(dbCon, "CCF_ING", df, append=FALSE, overwrite=TRUE)
  dbDisconnect(dbCon)
  perfLogExit(dfEntry=dfEntry, db="CrispmlDB", measurement="Crispml")

  # log to stderr
  logMsg("repDataIngest exit")

  # return dataframe with the information provided by summary for numeric columns
  c <- as.data.frame(sapply(df[, c(1:30)],summary))
  return(toJSON(c))
}
```

Thomas.strehl@at.ibm.com *Stand 20200120*

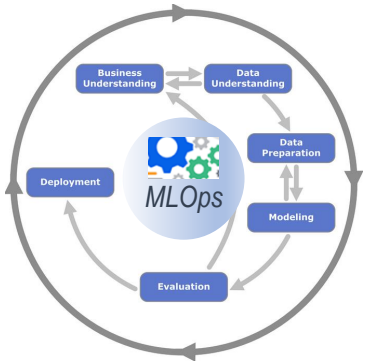
REnterprise: CrispML - Shiny call to remote REST API




```
# ------------------------------------------------------------------
# repDataIngest:
# ------------------------------------------------------------------
erDataIngest <- eventReactive(input$btnDataIngest, {
  p <- round(input$numPercentData/100,2)
  args <- paste0("?percentData=", p)
  logMsg(paste("eventReactive", "btnDataIngest", args), sessionID)
  c <- content(GET(url=paste0(crispml_url, "/repDataIngest", args)))[[1]]
  fromJSON(c)
}, ignoreNULL=TRUE, ignoreInit=TRUE)

output$dtoDataIngest <- renderDataTable({erDataIngest()})
```

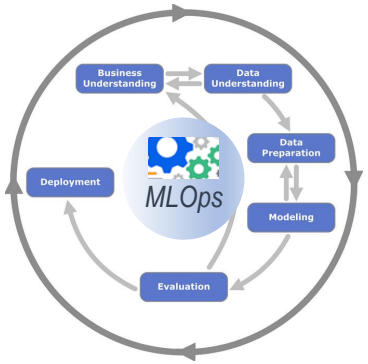
Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - GIT commit in RStudio



Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: MLOps – Tekton build & deploy pipeline to K8S



IBM Cloud | Search resources and offerings... | Catalog | Docs | Support | Manage

Toolchains / REP-Tekton-Pipeline / REP-Tekton-Delivery / Delivery Pipeline Tekton PipelineRun

pipelinerun-f15ba576-ddf4-4dcb-86a1-9febe18f3fa1 2 days ago

Succeeded All Tasks have completed executing

Triggered by thomas.strehl Meetup live

- pipeline-build-task
- pipeline-validate-task
- pipeline-deploy-task
  - clone-repo Completed
  - pre-deploy-ch... Completed
  - deploy-to-kub... Completed

clone-repo Completed

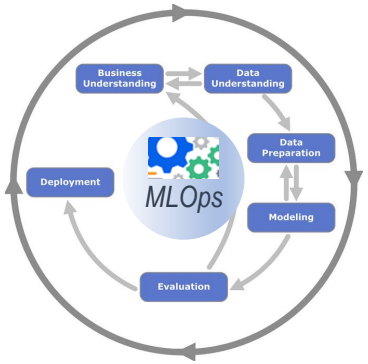
Logs | Status | Details

```
Cloning https://TetkonPipe:FNDeSac4yG961MXSLmwK@eu-de.git.cloud.ibm.com/rep/rep-admin
```

Step completed

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - Build Pipeline: Reference to source



IBM Cloud | Projects | Groups | Activity | Milestones | Snippets | Search or jump to...

R rep-admin

- Project
- Repository
  - Files
  - Commits
  - Branches
  - Tags
  - Contributors
  - Graph
  - Compare
  - Charts
- Issues 3
- Merge Requests 1
- Wiki
- Collapse sidebar

REP › rep-admin › Commits › ee7e12d0

Commit ee7e12d0 authored 16 hours ago by THOMAS STREHL | Browse files | Options

Meetup live

parent c0f78db3 master

No related merge requests found

Changes 1

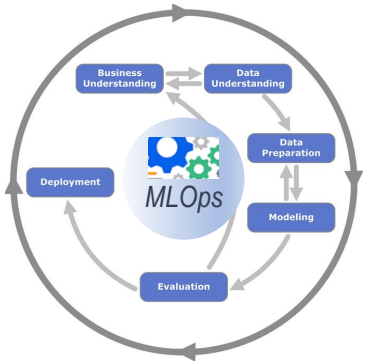
Showing 1 changed file with 1 addition and 1 deletion | Hide whitespace changes | Inline | Side-by-side

admin/ui.R | View file @ ee7e12d0

```
...  ...  @@ -6,7 +6,7 @@ options(shiny.reactlog=TRUE)
6    6    shinyUI(fluidPage(
7    7
8    8      # Application title
9         -  titlePanel("REnterprise - CrispML Console"),
     9    +  titlePanel("REnterprise - CrispML Console Live"),
10   10     p(strong("Frontend to REST API of Training/Prediction Engine")),
11   11
12   12     # Sidebar
```

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: K8S - Dashboard: Deployments



kubernetes

Search

Workloads > Deployments

Cluster

- Cluster Roles
- Namespaces
- Nodes
- Persistent Volumes
- Storage Classes

Namespace: repnamespace

Overview

Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

Discovery and Load Balancing

- Ingresses
- Services

CPU Usage

CPU (cores)

0

11:50 11:51 11:52 11:53 11:54 11:55 11:56 11:57 11:58 11:59 12:00 12:01 12:02 12:03 12:04

Memory Usage

Memory (bytes)

500 Mi

0 Mi

11:50 11:51 11:52 11:53 11:54 11:55 11:56 11:57 11:58 11:59 12:00

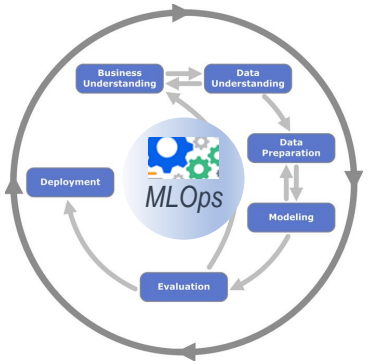
Deployments

Name	Labels	Pods	Age	Images
rep-grafana	context: REnterprise	1 / 1	7 days	de.icr.io/regnamespace/rep-grafana:latest
rep-influxdb	context: REnterprise	1 / 1	7 days	de.icr.io/regnamespace/rep-influxdb:latest
rep-crispml	context: REnterprise	1 / 1	10 days	de.icr.io/regnamespace/rep-crispml:master-0-ffede7f6-20200225151727
rep-admin	context: REnterprise	1 / 1	10 days	de.icr.io/regnamespace/rep-admin:master-0-ee7e12d0-20200225185534

1 – 4 of 4

Thomas.strehl@at.ibm.com *Stand 20200120*

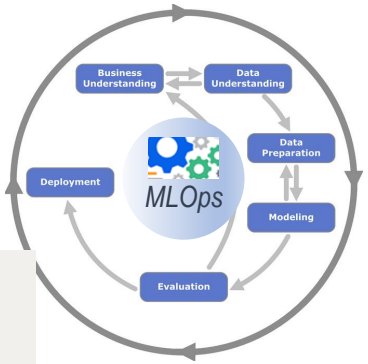
REnterprise: InfluxDB - Performance Logging with InfluxdbR




```
# ------------------------------------------------------------------------------------------
# Performance Logging:
# ------------------------------------------------------------------------------------------
perfLogEntry <- function(method, dataSet, mlModel) {

  tBeg <- Sys.time() # 0.002 ms
  pid <- Sys.getpid() # 0.011 ms
  dfEntry <- data.frame(stamp=tBeg, mode="Entry", method=method, dataSet=dataSet, mlModel=mlModel,
                        pid=pid, memTotal=0, gcN=0,gcV=0,Duration=0) # 50ms (10ms for con)
  return(dfEntry)
}


perfLogExit <- function(dfEntry, db="CrispmlDB", measurement="Crispml") {

  tEnd <- Sys.time()
  mp <- as.data.frame(t(memory.profile())) # 205ms
  gc <- gc() # 150ms
  dfExit <- dfEntry
  dfExit$memTotal <- round(sum(mp)/1000000,3)
  df$gcN <- gc[1]
  df$gcV <- gc[2]
  dfExit$mode <- "Exit"
  dfExit$Duration <- tEnd - dfEntry$stamp

  influx_write(con=conInfluxDB(), x=dfExit, db=db, measurement=measurement, precision="u",
               time_col="stamp", tag_cols=c("mode","method","dataSet","mlModel","pid"))
}
```

Thomas.strehl@at.ibm.com *Stand 20200120*

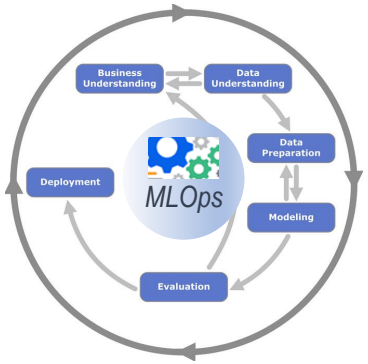
REnterprise: logDNA - CrispML application log




```
Feb 27 10:16:32 rep-crispml 20200227-091632.614 __plumber__ repModelTrain entry CCF_CUR 1 6 rf 10
Feb 27 10:16:32 rep-crispml 20200227-091632.616 __plumber__ Connecting to DB2 at db2w-qwtmrdf.eu-de.db2w.cloud.ibm.com
Feb 27 10:16:32 rep-crispml                      stamp mode        method dataSet mlModel pid memTotal gcN gcV     Duration
Feb 27 10:16:32 rep-crispml 1 2020-02-27 09:16:24.9263 Exit repModelTrain CCF_CUR      rf   1    2.384   0   0 7.42671 secs
Feb 27 10:16:32 rep-crispml 20200227-091632.760 __plumber__ Connecting to influxdb at: rep-influxdb-service    REP_CRISPML_ENV: Deployment
Feb 27 10:16:32 rep-crispml Success: (204) No Content
Feb 27 10:16:32 rep-crispml 20200227-091632.860 __plumber__ repModelTrain exit
Feb 27 10:16:33 rep-crispml 20200227-091633.200 __plumber__ repModelTrain entry CCF_CUR 1 6 rf 10
Feb 27 10:16:33 rep-crispml 20200227-091633.208 __plumber__ Connecting to DB2 at db2w-qwtmrdf.eu-de.db2w.cloud.ibm.com
Feb 27 10:16:33 rep-crispml                        stamp mode        method dataSet mlModel pid memTotal gcN gcV       Duration
Feb 27 10:16:33 rep-crispml 1 2020-02-27 09:16:32.614767 Exit repModelTrain CCF_CUR     db2   1    2.383   0   0 0.2947791 secs
Feb 27 10:16:33 rep-crispml 20200227-091633.375 __plumber__ Connecting to influxdb at: rep-influxdb-service    REP_CRISPML_ENV: Deployment
Feb 27 10:16:33 rep-crispml Success: (204) No Content
Feb 27 10:16:34 rep-crispml                        stamp mode        method dataSet mlModel pid memTotal gcN gcV       Duration
Feb 27 10:16:34 rep-crispml 1 2020-02-27 09:16:33.206232 Exit repModelTrain CCF_CUR     db2   1    2.401   0   0 0.3378892 secs
Feb 27 10:16:34 rep-crispml 20200227-091634.071 __plumber__ Connecting to influxdb at: rep-influxdb-service    REP_CRISPML_ENV: Deployment
Feb 27 10:16:34 rep-crispml Success: (204) No Content
Feb 27 10:16:46 rep-crispml                        stamp mode        method dataSet mlModel pid memTotal gcN gcV      Duration
Feb 27 10:16:46 rep-crispml 1 2020-02-27 09:16:33.450734 Exit repModelTrain CCF_CUR      rf   1    2.384   0   0 13.06827 secs
Feb 27 10:16:46 rep-crispml 20200227-091646.923 __plumber__ Connecting to influxdb at: rep-influxdb-service    REP_CRISPML_ENV: Deployment
Feb 27 10:16:46 rep-crispml Success: (204) No Content
Feb 27 10:16:47 rep-crispml 20200227-091647.086 __plumber__ repModelTrain exit
Feb 27 10:16:48 rep-crispml                        stamp mode        method dataSet mlModel pid memTotal gcN gcV      Duration
Feb 27 10:16:48 rep-crispml 1 2020-02-27 09:16:34.186166 Exit repModelTrain CCF_CUR      rf   1    2.402   0   0 14.02401 secs
Feb 27 10:16:48 rep-crispml 20200227-091648.633 __plumber__ Connecting to influxdb at: rep-influxdb-service    REP_CRISPML_ENV: Deployment
Feb 27 10:16:48 rep-crispml Success: (204) No Content
Feb 27 10:16:48 rep-crispml 20200227-091648.708 __plumber__ repModelTrain exit
```

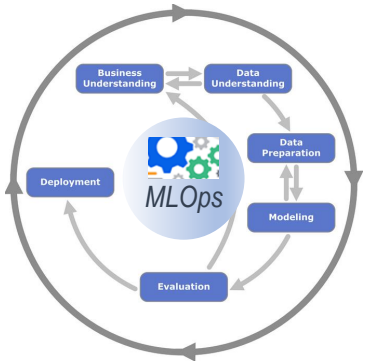
Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: k6 - Dockerfile




```
FROM loadimpact/k6

RUN mkdir /k6-tests
COPY  ./k6-tests /k6-tests

# This is set in loadimpact/k6 Dockerfile
ENTRYPOINT ["k6"]

ENV INFLUX_HOST=rep-influxdb-service
ENV INLFUX_PORT=8086
ENV INFLUX_DB=DS_INFLUXDB

# Set working dir to scripts dir
WORKDIR /k6-tests/scripts
CMD ["run", "--out", "influxdb=http://rep-influxdb-service:8086/DS_INFLUXDB", "-d 100s", "-u 1", "sleep.js"]
```

REnterprise: k6 - LoadTest Script and K8S job yaml

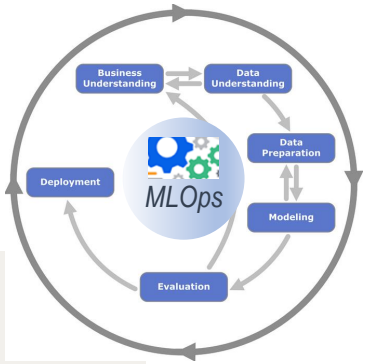



```
(base) [at062084@oc8262044446 scripts]$ cat repModelTrain.js
import http from "k6/http";
import { check, group, sleep } from "k6";
import { Counter, Rate, Trend } from "k6/metrics";

export default function() {
  group ("grpTrain", function() {
    let res4 = http.get("http://rep-crispml-service:8000/crispml/repModelTrain?table=CCF_CUR&from=1&to=6&model=rf&weight=10");
    let checkRes4 = check(res4, {"status is 200": (r) => r.status === 200});
    //checkFailureRate.add(!checkRes4);
  });
};
```

```
(base) [at062084@oc8262044446 rep-k8s]$ cat job.rep-k6.yaml
apiVersion: batch/v1
kind: Job
metadata:
  name: rep-k6-job
  namespace: repnamespace
  labels:
    app: rep-k6-job
    context: REnterprise
spec:
  template:
    spec:
      containers:
      - name: rep-k6-job
        image: de.icr.io/regnamespace/rep-k6:latest
        args: ["run", "--out", "influxdb=http://rep-influxdb-service:8086/DS_INFLUXDB", "-u", "1", "-d","60","repModelTrain.js"]
      restartPolicy: Never
```

Thomas.strehl@at.ibm.com *Stand 20200120*

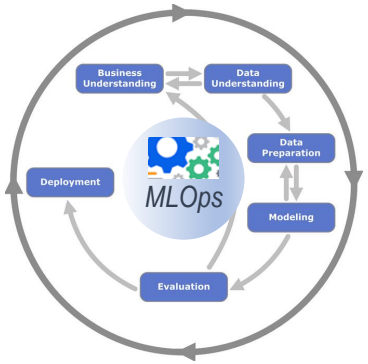
REnterprise: k6 - Performance test Report




```
       init [------------------------------------------------------------] starting
       ▉ grpTrain
Feb 27 10:07:40 rep-k6-job
         ✓ status is 200
Feb 27 10:07:40 rep-k6-job
       checks.....................: 100.00% ✓ 6   ✗ 0
       data_received..............: 49 kB   820 B/s
       data_sent..................: 936 B   15 B/s
       group_duration.............: avg=9.36s    min=9s       med=9.33s    max=9.78s    p(90)=9.66s    p(95)=9.72s
       http_req_blocked...........: avg=1.48ms   min=296.02µs med=387.68µs max=6.9ms    p(90)=3.75ms   p(95)=5.33ms
       http_req_connecting........: avg=221.7µs  min=182.81µs med=202.37µs max=333.28µs p(90)=276.78µs p(95)=305.03µs
       http_req_duration..........: avg=9.36s    min=9s       med=9.33s    max=9.77s    p(90)=9.65s    p(95)=9.71s
       http_req_receiving.........: avg=223.37µs min=152.25µs med=156.8µs  max=540.47µs p(90)=360.25µs p(95)=450.36µs
       http_req_sending...........: avg=92.1µs   min=67.76µs  med=79.33µs  max=140.92µs p(90)=127.1µs  p(95)=134.01µs
       http_req_tls_handshaking...: avg=0s       min=0s       med=0s       max=0s       p(90)=0s       p(95)=0s
       http_req_waiting...........: avg=9.36s    min=9s       med=9.33s    max=9.77s    p(90)=9.65s    p(95)=9.71s
       http_reqs..................: 6       0.099996/s
       iteration_duration.........: avg=9.36s    min=9s       med=9.33s    max=9.78s    p(90)=9.66s    p(95)=9.72s
       iterations.................: 6       0.099996/s
       vus........................: 1       min=1 max=1
       vus_max....................: 1       min=1 max=1
```

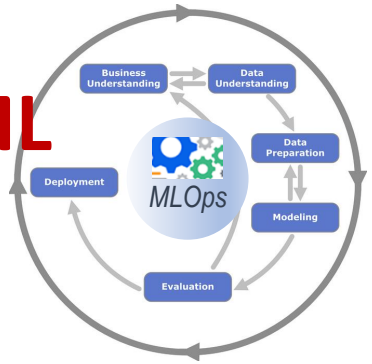
Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: InfluxDB+Grafana: k6 & Plumber respone time



Thomas.strehl@at.ibm.com Stand 20200120

REnterprise: K8S Dashboard: Scale to 3 pods running CrispML



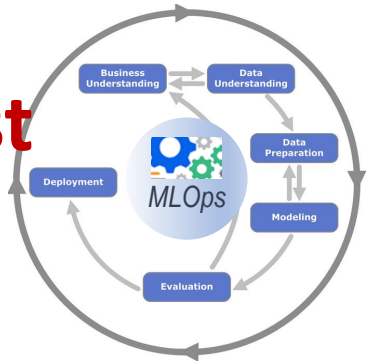
Pods

	Name	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Age
✓	rep-crispml-5f6dd9495d-n5rgs	app: rep-crispml pod-template-hash: 5f6dd9495d	10.135.151.76	Running	0	259.00m	208.32Mi	-
✓	rep-crispml-5f6dd9495d-zvdjk	app: rep-crispml pod-template-hash: 5f6dd9495d	10.135.151.85	Running	0	363.00m	214.83Mi	-
!	rep-crispml-6bc8c99bbf-8wfxl	app: rep-crispml pod-template-hash: 6bc8c99bbf	10.135.151.76	Waiting: ErrImagePull	0	-	-	-
✓	rep-k6-job-2pcw8	controller-uid: 7e79fd93-dc29-4e7a-9c92-aca734d99ecc job-name: rep-k6-job	10.135.151.76	Running	0	45.00m	55.55Mi	-
✓	rep-admin-55696bc9fd-9sp6v	app: rep-admin pod-template-hash: 55696bc9fd	10.135.151.76	Running	0	1.00m	175.98Mi	18 hours
✓	rep-crispml-5f6dd9495d-hp9x5	app: rep-crispml pod-template-hash: 5f6dd9495d	10.135.151.76	Running	0	1.28	965.84Mi	22 hours
✓	rep-grafana-84bb9d6f79-nc4k9	app: rep-grafana pod-template-hash: 84bb9d6f79	10.135.151.84	Running	0	2.00m	24.87Mi	2 days
✓	rep-influxdb-5fdfc5b48c-c5fkh	app: rep-influxdb pod-template-hash: 5fdfc5b48c	10.135.151.84	Running	0	9.00m	73.79Mi	2 days

1 – 8 of 8

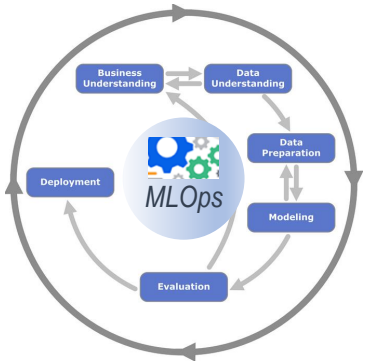
Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: sysDIG – K8S Resources during pod scale up test



Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - GitLab Repository



REP › rep-crispml › **Details**

R **rep-crispml**
Project ID: 39851

Star 0 | Fork 0 | Clone

Add license | 45 Commits | 1 Branch | 46 Tags | 414.9 MB Files

Plumber REST API to invoke function based on CRISP-DM methodology

master | rep-crispml / | + | History | Find file

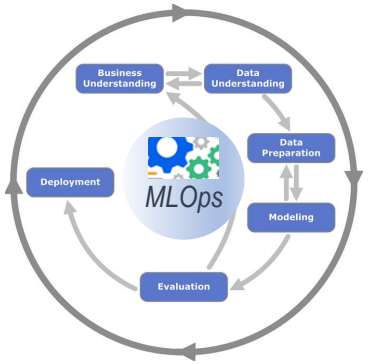
Added try grid for rf to produce more cpu load with doMC=4
THOMAS STREHL authored 13 hours ago — ffe12ef4

README | Add CHANGELOG | Add CONTRIBUTING

Name	Last commit	Last update
.tekton	pipelinerun	3 days ago
data	Add DB2 connection via odbc. Add test data	1 week ago
dsdriver	Added driver for DB. Adpated Dockerfile for DB2. crispT...	1 week ago
rep	Added try grid for rf to produce more cpu load with doM...	13 hours ago
share	storage for share added	1 week ago
tekton	tekton pipeline added	4 days ago
Dockerfile	Adapts to logging	1 day ago
README.md	Update README.md - Fixes #1	3 days ago
build.sh	Initial plumber image serving 'echo' REST api	4 weeks ago
deployment.yml	More REST calls implemented. Several bugfixes. Env for ...	3 days ago
odbc.ini	Added driver for DB. Adpated Dockerfile for DB2. crispT...	1 week ago

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - GitLab Issue Board



IBM Cloud | Projects | Groups | Activity | Milestones | Snippets | Search or jump to...

R rep-crispml

REP › rep-crispml › Issue Boards

Project
Repository
Issues 5
List
Boards
Labels
Milestones
Merge Requests 0
Wiki
Snippets
Settings

Development | Search or filter results... | Add list | Add issues

Open 3
- US Readme improvement #5
- US someMoretodos #6
- US MLDataExtractor #3 Friday 1w

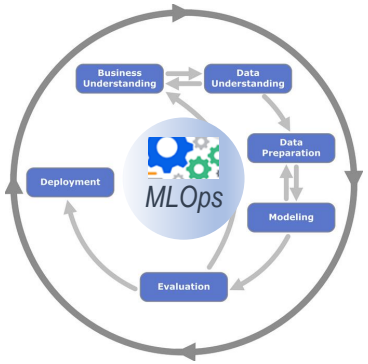
To Do 1
- Defect error 612 To Do #4 Friday 3h

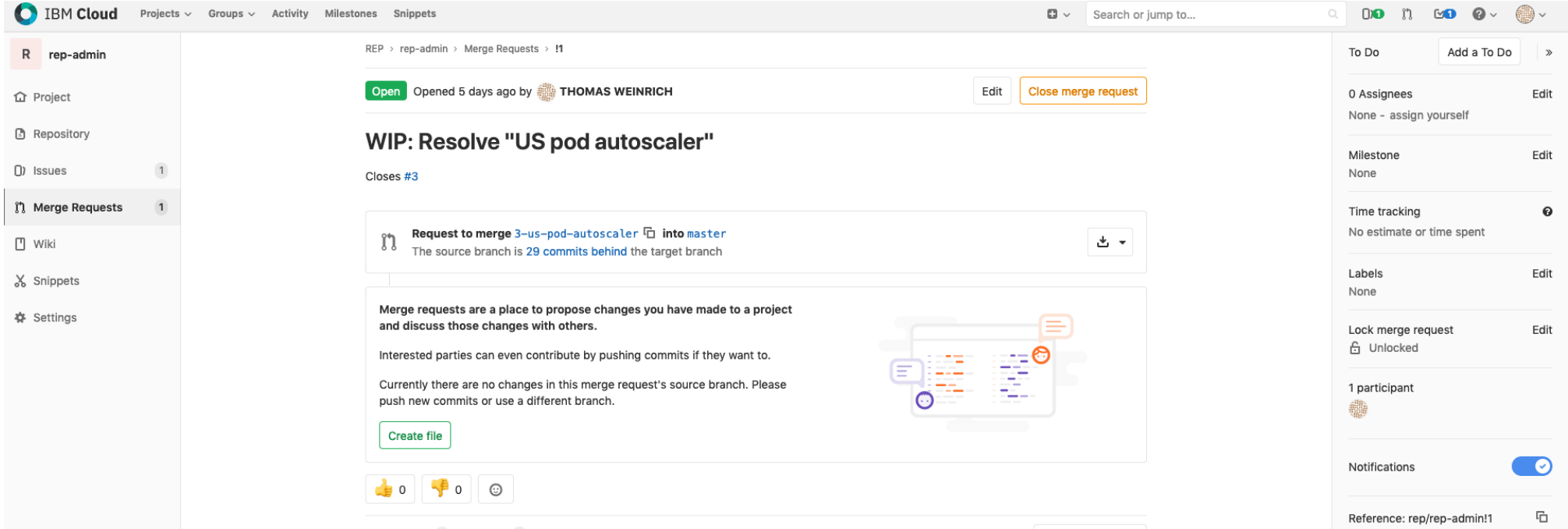
Doing 1
- US MLTrainingAPI Doing #2

Closed 1
- Issue Readme #1 Feb 21 5m

Thomas.strehl@at.ibm.com *Stand 20200120*

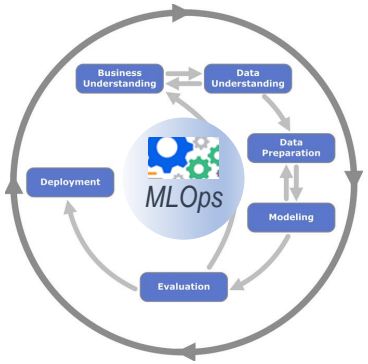
REnterprise: MLOps - GitLab merge Request





Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - Tekton Pipelines



Overview

Connections

Manage

Toolchains /

REP-Tekton-Pipeline

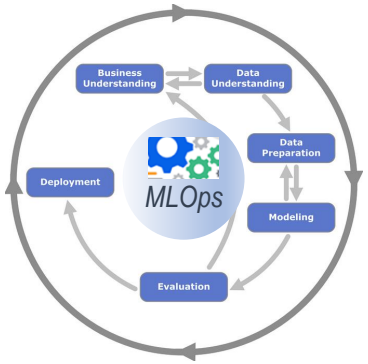
Resource Group: REP-ResourceGroup **Location:** Frankfurt Add tags

Your toolchain is ready! Quick start: You can now add tool integrations. For step-by-step instructions, see the tutorial for this toolchain.

THINK	CODE	DELIVER
Issues rep-admin ✓ Configured	**Git** rep-admin ✓ Configured	**Delivery Pipeline** REP-Tekton-Delivery ✓ Configured
Issues rep-crispml ✓ Configured	**Git** rep-crispml ✓ Configured	**Delivery Pipeline Pri...** REP-Tekton-Worker ✓ Configured

Thomas.strehl@at.ibm.com *Stand 20200120*

REnterprise: MLOps - Tekton Pipeline Trigger



Definitions | Worker | **Triggers** | Environment prop...

Triggers specify what happens when a specified event occurs. Manual Triggers map to a Tekton EventListener resource. Git Triggers map git webhook events to a Tekton EventListener. Timed triggers invoke the mapped Tekton EventListener at the specified time. In all cases the available listeners are those defined in the pipeline definition.

Add trigger

Manual Admin

- [x] Enable concurrent runs by this trigger

EventListener: listener

Git Trigger

- [x] Enable concurrent runs by this trigger

Repository: rep-admin (https://eu-de.git.cloud.ibm.com/rep/rep-admin.git)

Branch: master

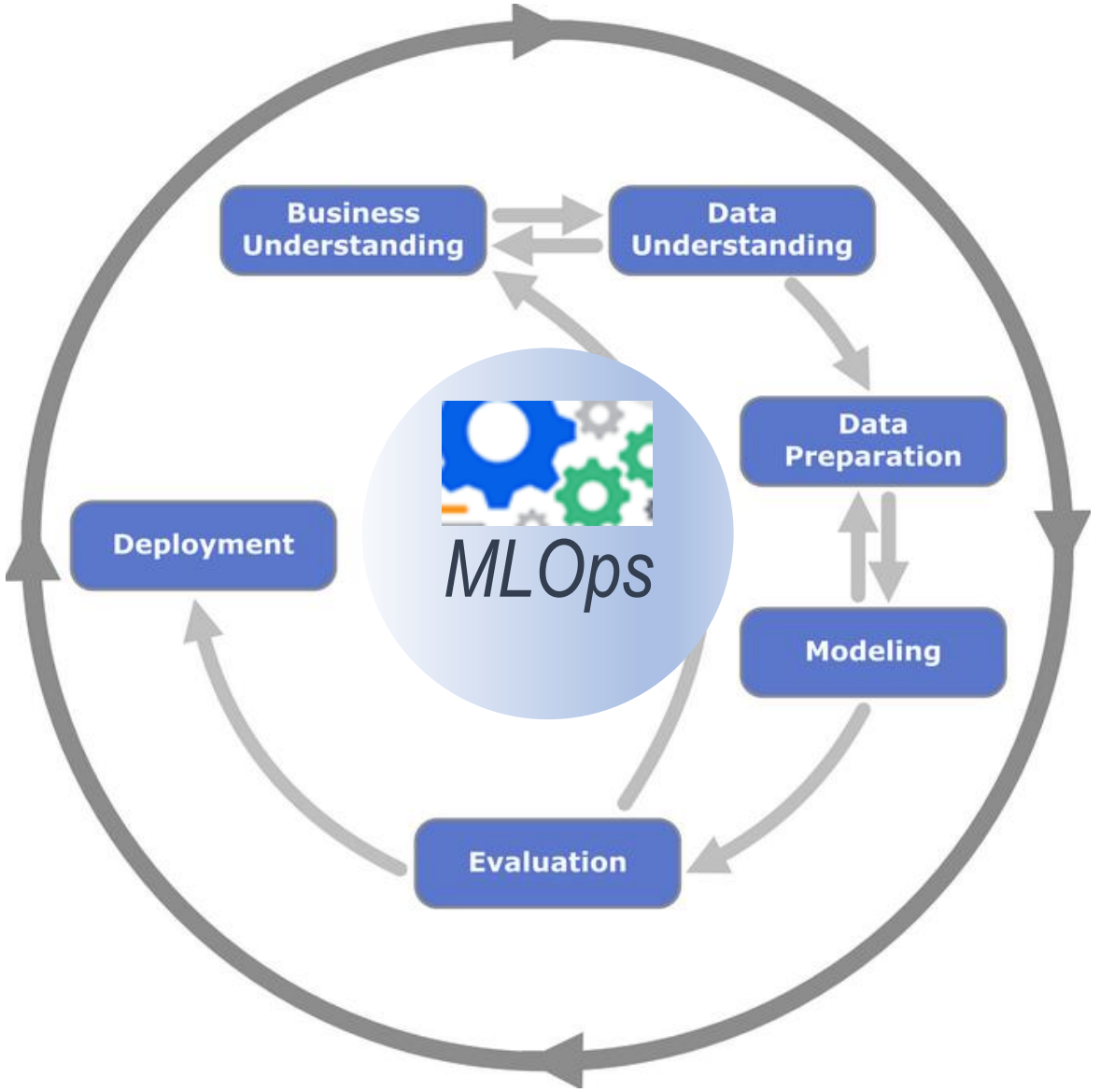
Run jobs automatically for Git events on the chosen branch

- [x] When a commit is pushed
- [ ] When a merge request is opened or updated
- [ ] When a merge request is closed

EventListener: listener

Thomas.strehl@at.ibm.com *Stand 20200120*

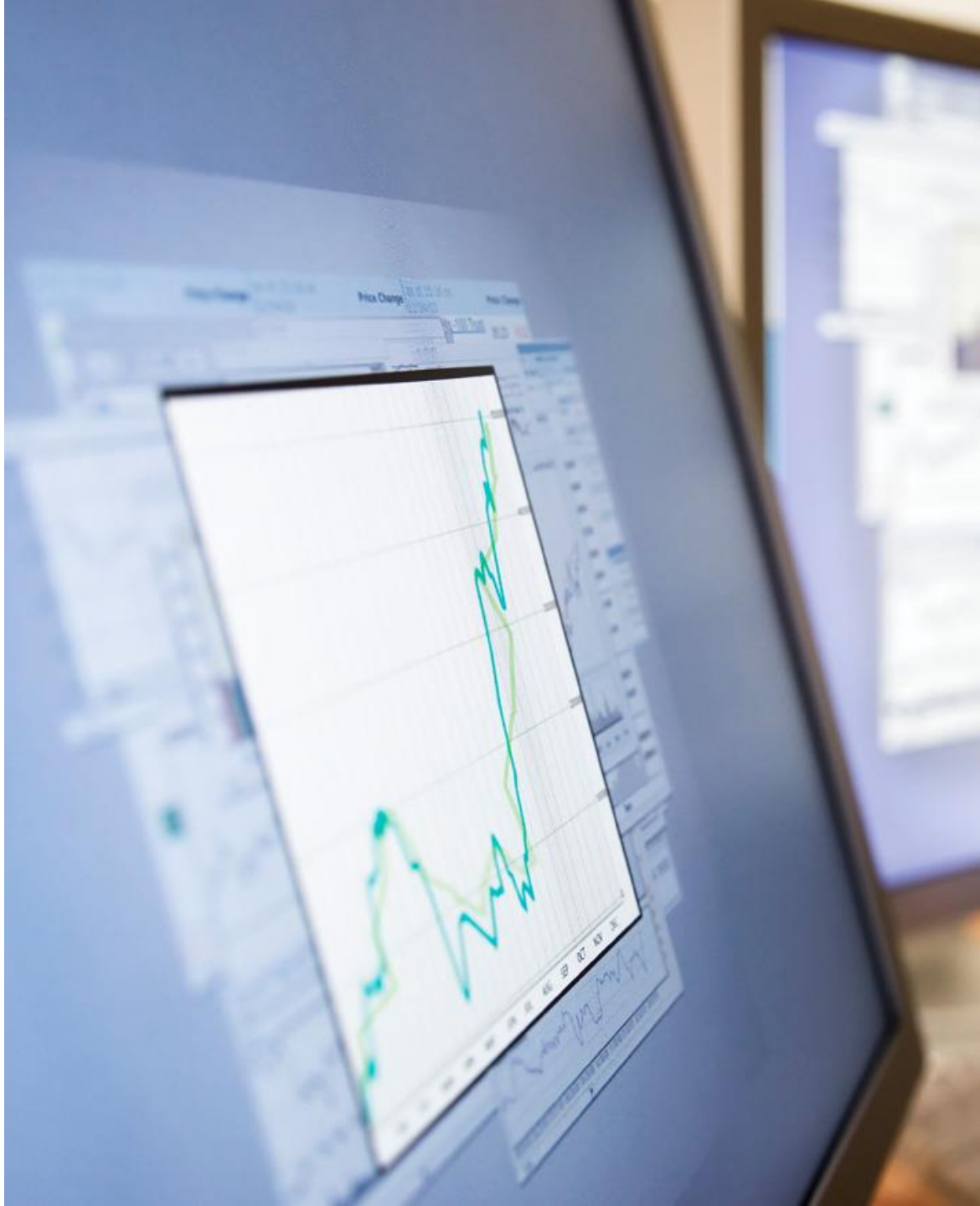
FIN



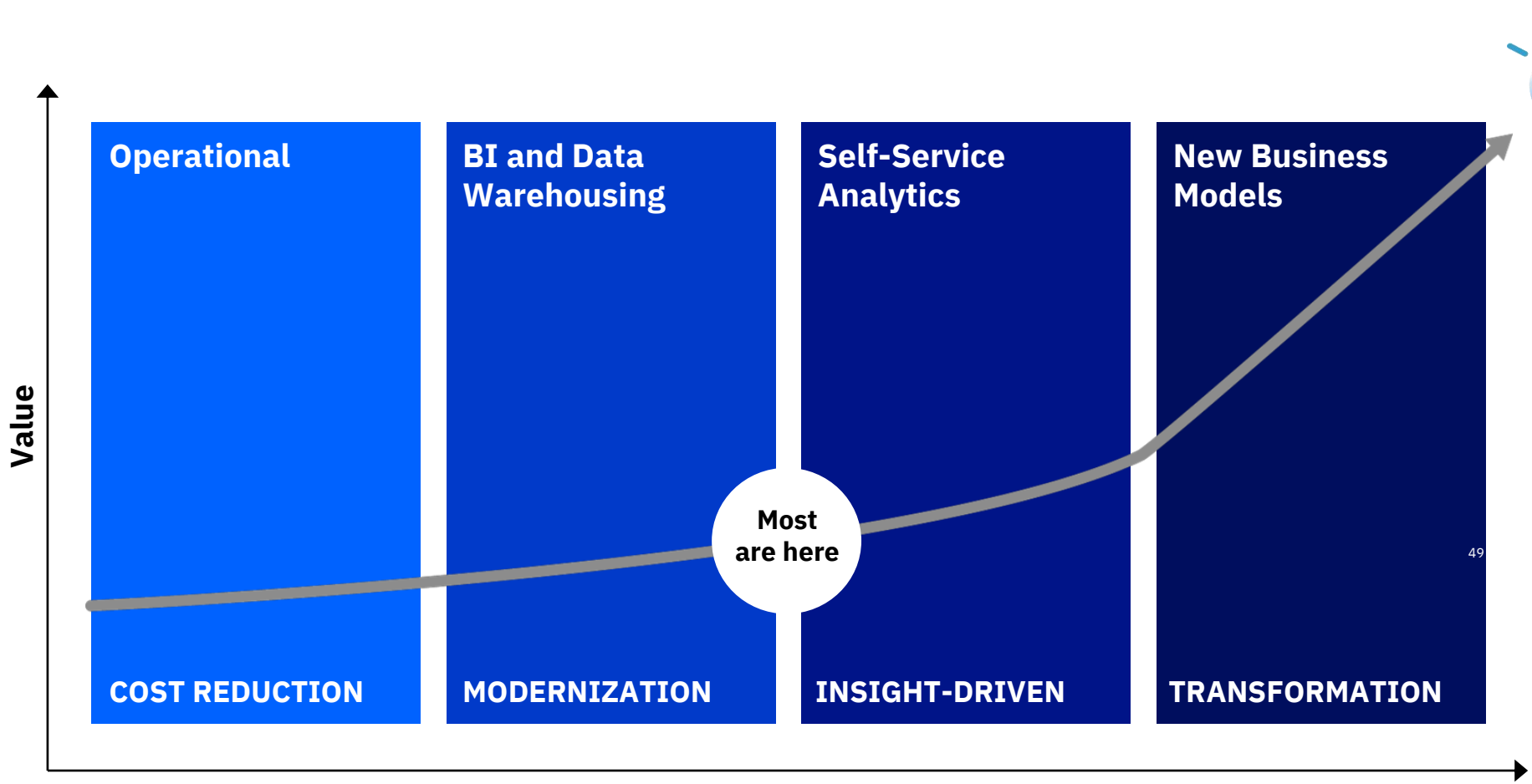
Thomas.Strehl@at.ibm.com Thomas.Weinrich@at.ibm.com Rudolf.Pailer@at.ibm.com @20200120

Towards an Enterprise grade Machine Learning pipeline with R

Contributions to a machine learning oriented pipeline in an enterprise environment



I want AI !



85%
view AI as a strategic opportunity

MITSloan

BUT..., business stakeholders do not trust AI.

60%

of companies see **regulatory constraints** as a barrier to implementing AI.

– *IBM IBV AI 2018*

63%

cite availability of **technical skills** as a challenge to implementation.

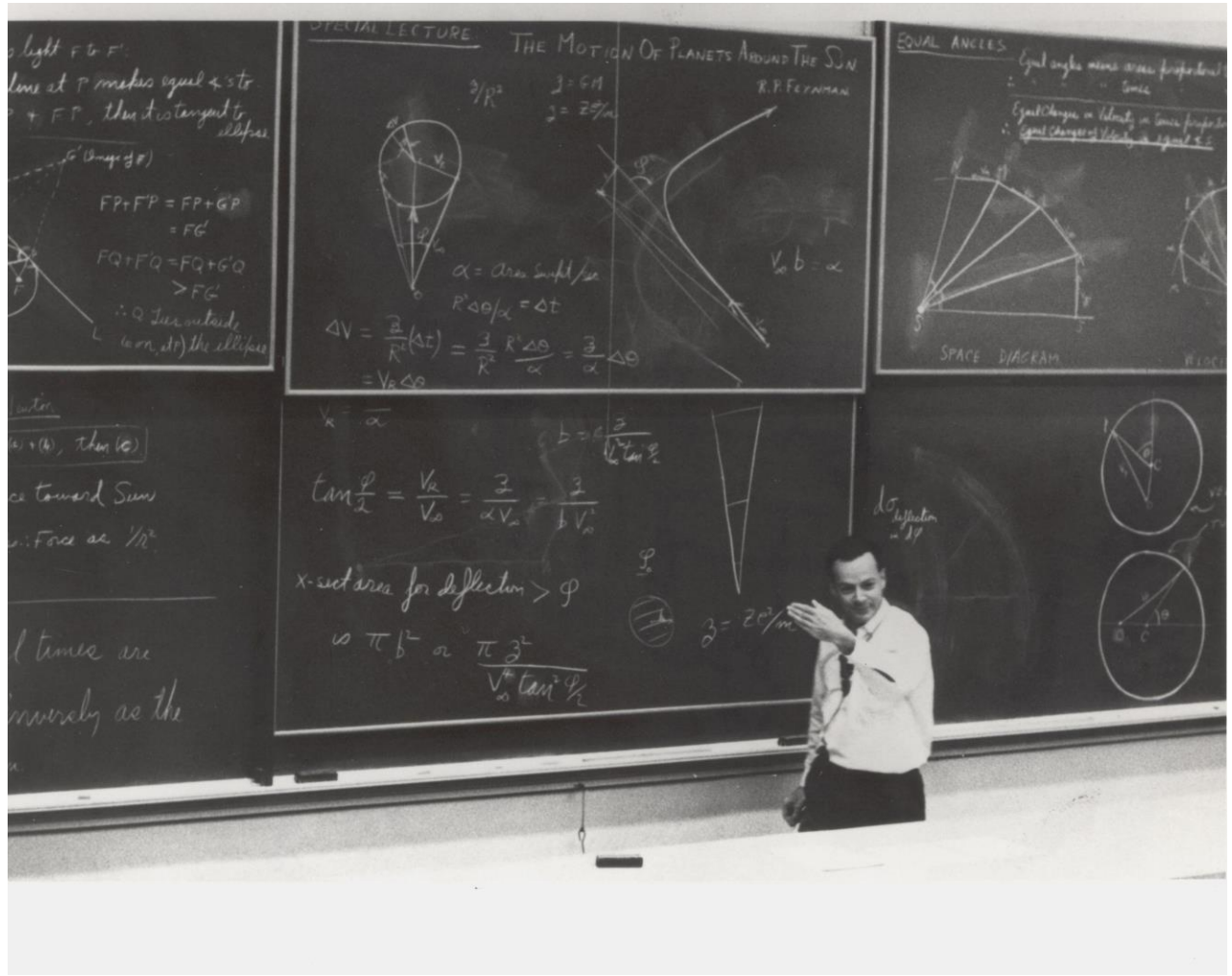
- IBM IBV AI 2018

Without expensive Data Science resources handholding multiple AI models in a production application:

1. No way to **validate** if AI models are **compliant with regulations** and will achieve expected business outcomes before deploying
2. Difficult to **track and measure** indicators of business success in production
3. Resource intensive and unreliable processes for **ongoing business monitoring and compliance**
4. Impossible for business users to **feedback** subtle domain knowledge into model lifecycle

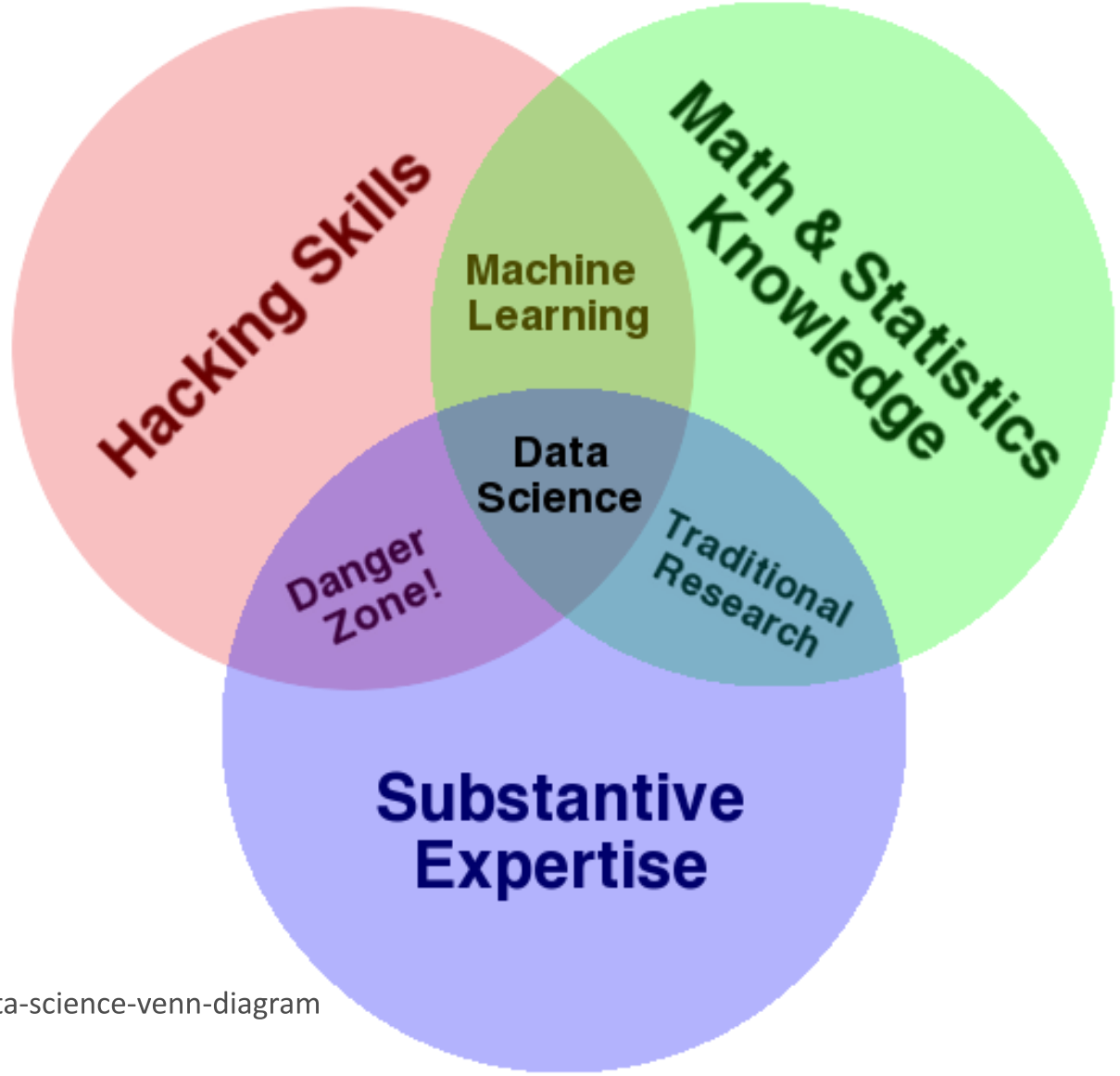
IBM

I have a Jupyter Notebook – Problem Solved



https://en.wikipedia.org/wiki/Feynman%27s_Lost_Lecture#/media/File:HD.3A.053_(10481714045).jpg

Skill Requirements in Data Science & AI Projects



- http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram

Hidden Depth in Machine Learning Systems

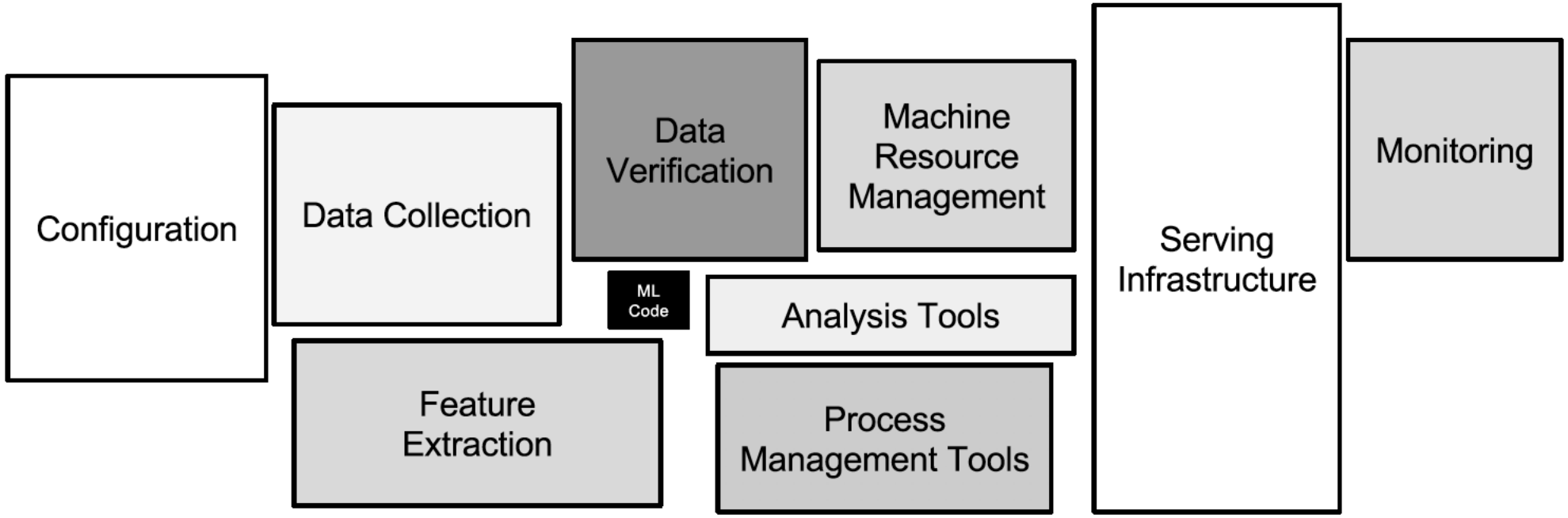
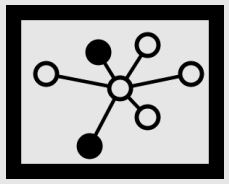


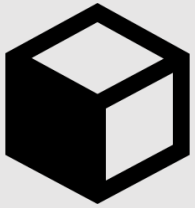
Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

D. Sculley et. al. “Hidden Technical Debt in Machine Learning Systems”,
https://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf

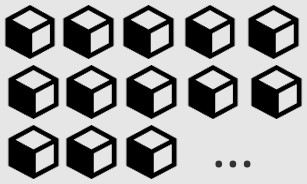
Machine Learning Life Cycle



Data Governance



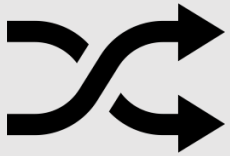
Modelling & Evaluation



Model Versioning

Model Deployment

Model Monitoring



Dynamic Model Selection & Retraining

Data Science Solutions are **not static** by definition!

Agile Governance & Steering – Results are regularly shared, focus can be adjusted

One Sprint **4** weeks

Executive Sponsors with Core & Extended Team

Demonstration / Review
Presentation of Pre-final Results

Strategic Themes
Prioritized Use Cases

Presentation Interim Results
e.g. Concepts, Prototypes, MVP

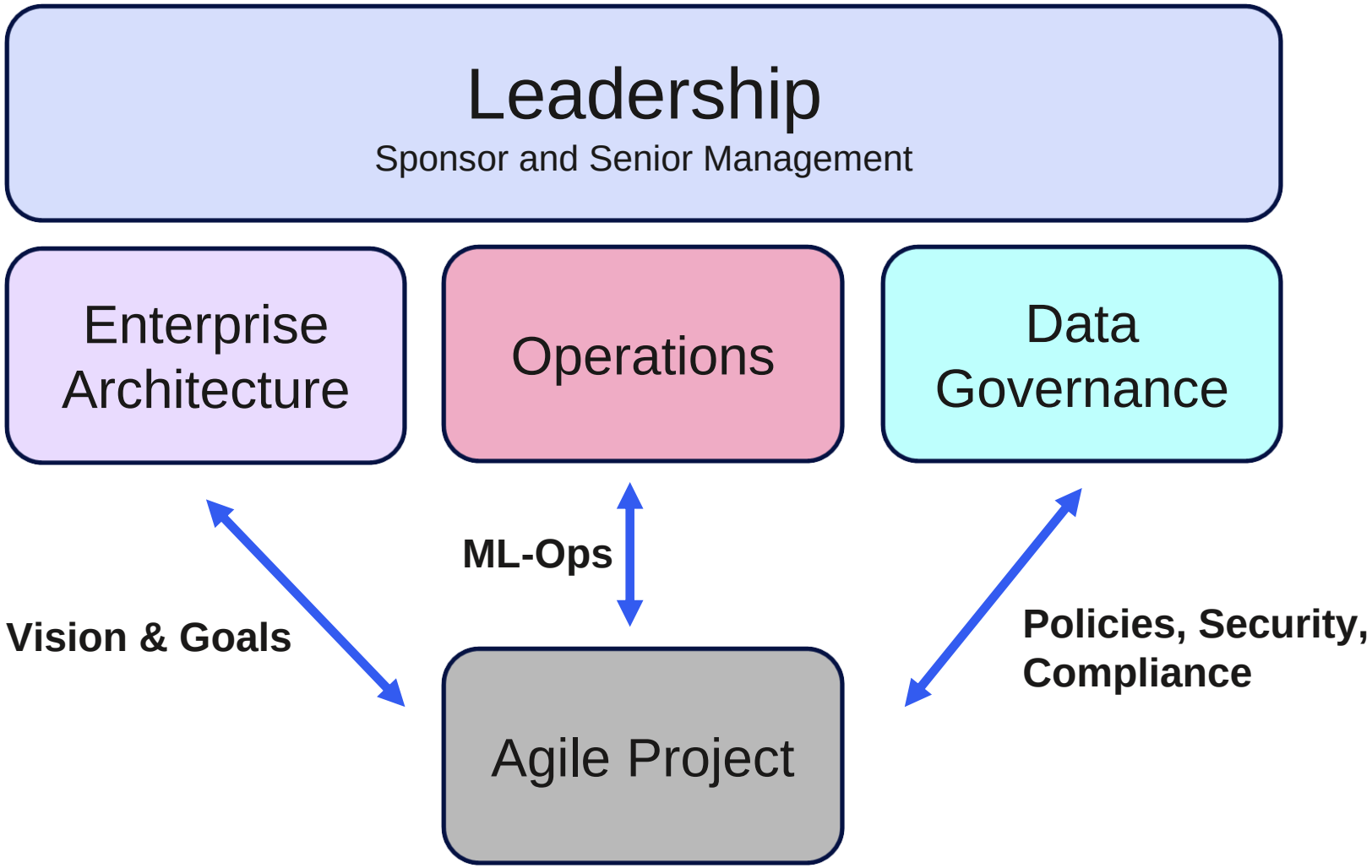
Daily Stand-up Meeting

Retrospective
Potential for optimization

Core Team & SMEs

Planning
Prioritized Deep Dive Topics

Example Governance Bodies



IBM

Compliance

Requirements

- EU General Data Protection Regulation - GDPR
- Industry Specific Regulations
  - Bankwesengesetz (BWG), Telekommunikationsgesetz (TKG), …
- General security and data protection considerations

Solutions

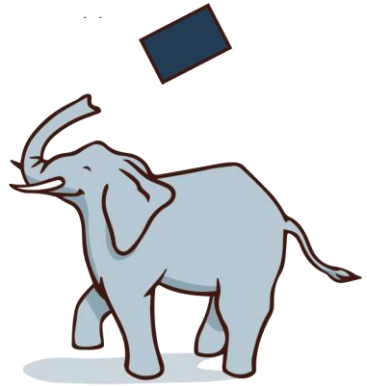
- Data Access Control
- Pseudonymization
- Anonymization
- Data aggregation (e.g. k-anonymity, background knowledge attack)
- Encryption (data at rest, data on the move)
- Audit Logs

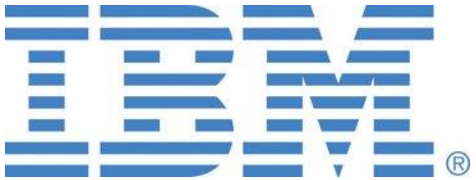
Data Governance

Extract, transform and load data
- Definition and management of data ingestion pipelines

Registration, Metadata & Discovery
- Find and understand ingested data sets
- Metadata for data sets
- Versioning of data sets
- Provenance and lineage of data sets

Access control
- Define users and roles
- Protect data against unauthorized access





Watson Knowledge Catalog