

Automated Documentation of Business Domain Assignments and Cloud Application Information from an Application Development Pipeline

Master Thesis: Final presentation

Nicolas Corpancho Villasana 08.02.2019

Chair of Software Engineering for Business Information Systems (sebis)

Faculty of Informatics

Technische Universität München

www.matthes.in.tum.de

1. Motivation

2. Research

2.1 Research questions

2.2 Existing tools with related functionality

2.3 Literature review

3. Solution

3.1 Solution architecture

3.2 Concept

3.3 Sample scenario

3.4 Class diagram

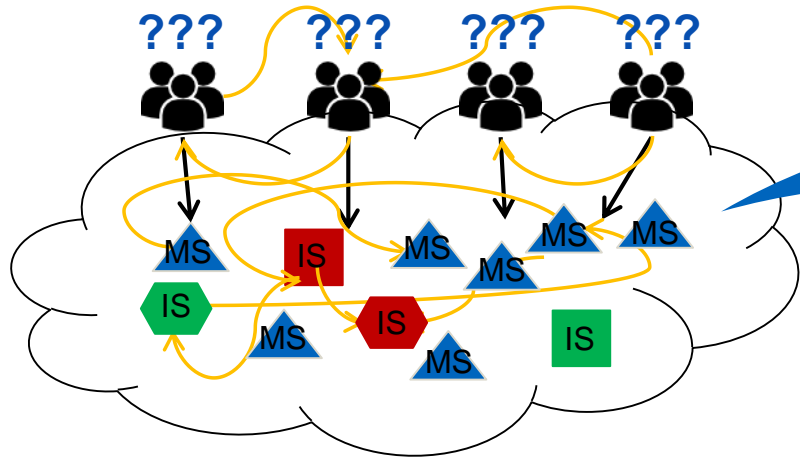
3.5 Live Demo

4. Evaluation

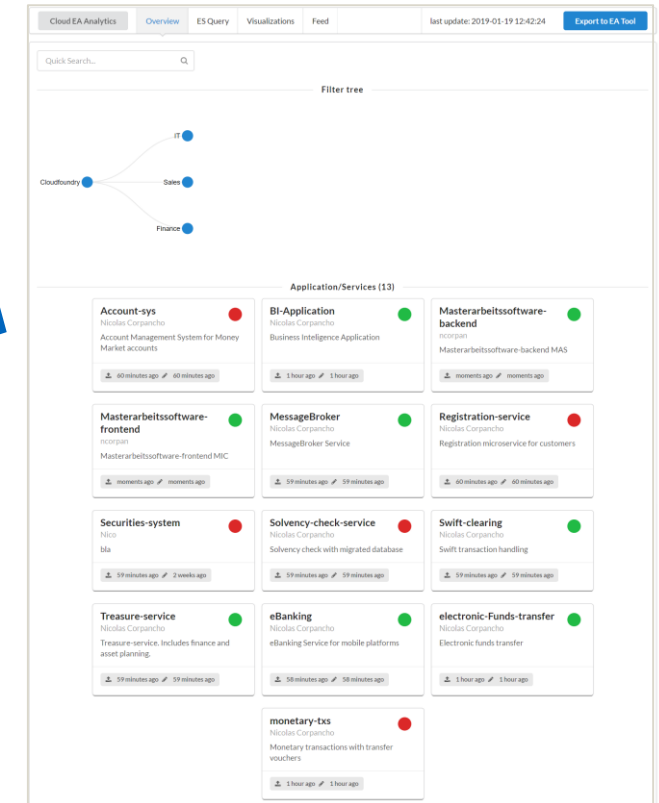
5. Conclusion

1. Motivation

Continuous Architecture Documentation



- Trend to large amount of microservices
- Shift to **Agile Development** and **Continuous Delivery (CD)**
- **Cloud environments and monitoring tools provide architecture information**



How can architecture documentation and assessment be integrated in the software development process?

1. Motivation

2. Research

2.1 Research questions

2.2 Existing tools with related functionality

2.3 Literature review

3. Solution

3.1 Solution architecture

3.2 Concept

3.3 Sample scenario

3.4 Class diagram

3.5 Live Demo

4. Evaluation

5. Conclusion

2.1 Research Questions

RQ1

How to obtain EA relevant information from the runtime behaviour of cloud based environments?

RQ2

How to assign the application landscape to business domains?

RQ3

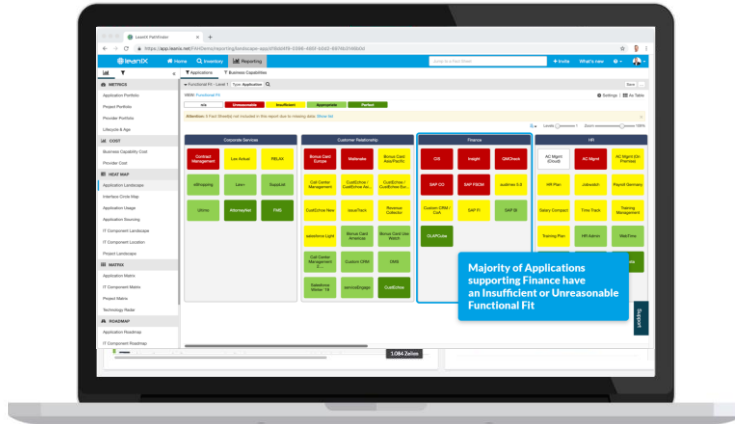
How to automate the assignment process with an integrated toolchain?

RQ4

How does a prototype implementation of the automated documentation process of cloud applications look like?

2.2 Existing tools with related functionality*

LeanIX

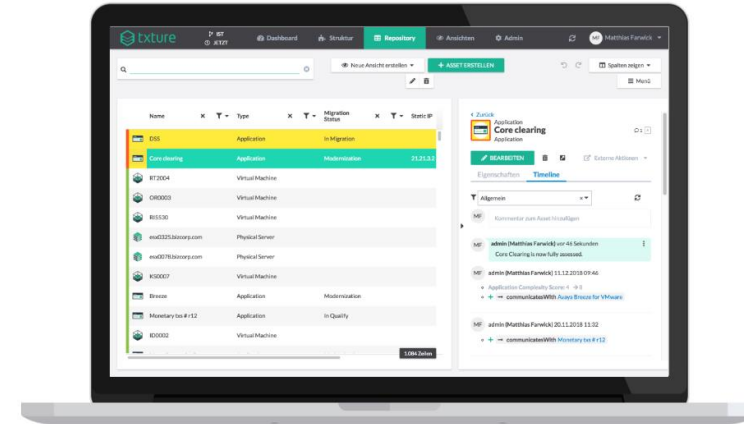


- Rest API for integration of different information sources
- R&D - Pivio integration



- No Continuous Delivery integration
- No focus on cloud environments
- No focus on runtime information

Txture



- Meta-model can be adapted at runtime
- Automated import mechanisms from various data sources
- Cloud environments integration



- No Continuous Delivery integration

2.3 Literature review

Year	Author	Title	CD	CC	RI
2012	Hauder et al.	Challenges for automated enterprise architecture documentation			X
2013	Roth et al.	Enterprise Architecture Documentation: Current Practices and Future Directions			
2010	Farwick et al.	Towards Living Landscape Models: Automated Integration of Infrastructure Cloud in Enterprise Architecture Management		X	X
2012	Buschle et al.	Automating Enterprise Architecture Documentation using an Enterprise Service Bus			X
2014	Holm et al.	Automatic data collection for enterprise architecture models			
2015	Välja et al.	A requirements based approach for automating enterprise it architecture modeling using multiple data sources			
2015	Farwick et al.	A situational method for semiautomated enterprise architecture documentation			
2016	Johnson et al.	Automatic probabilistic enterprise IT architecture modeling: A dynamic bayesian networks approach			
2018	Landthaler et al.	A Machine Learning Based Approach to Application Landscape Documentation			
2016	Bogner et al.	Towards Integrating Microservices with Adaptable Enterprise Architecture		X	

Current research endeavours lack in integrating cloud aspects (PaaS and SaaS) for a continuously automated EA documentation

Agenda

1. Motivation

2. Research

2.1 Research questions

2.2 Existing tools with related functionality

2.3 Literature review

3. Solution

3.1 Solution architecture

3.2 Concept

3.3 Sample scenario

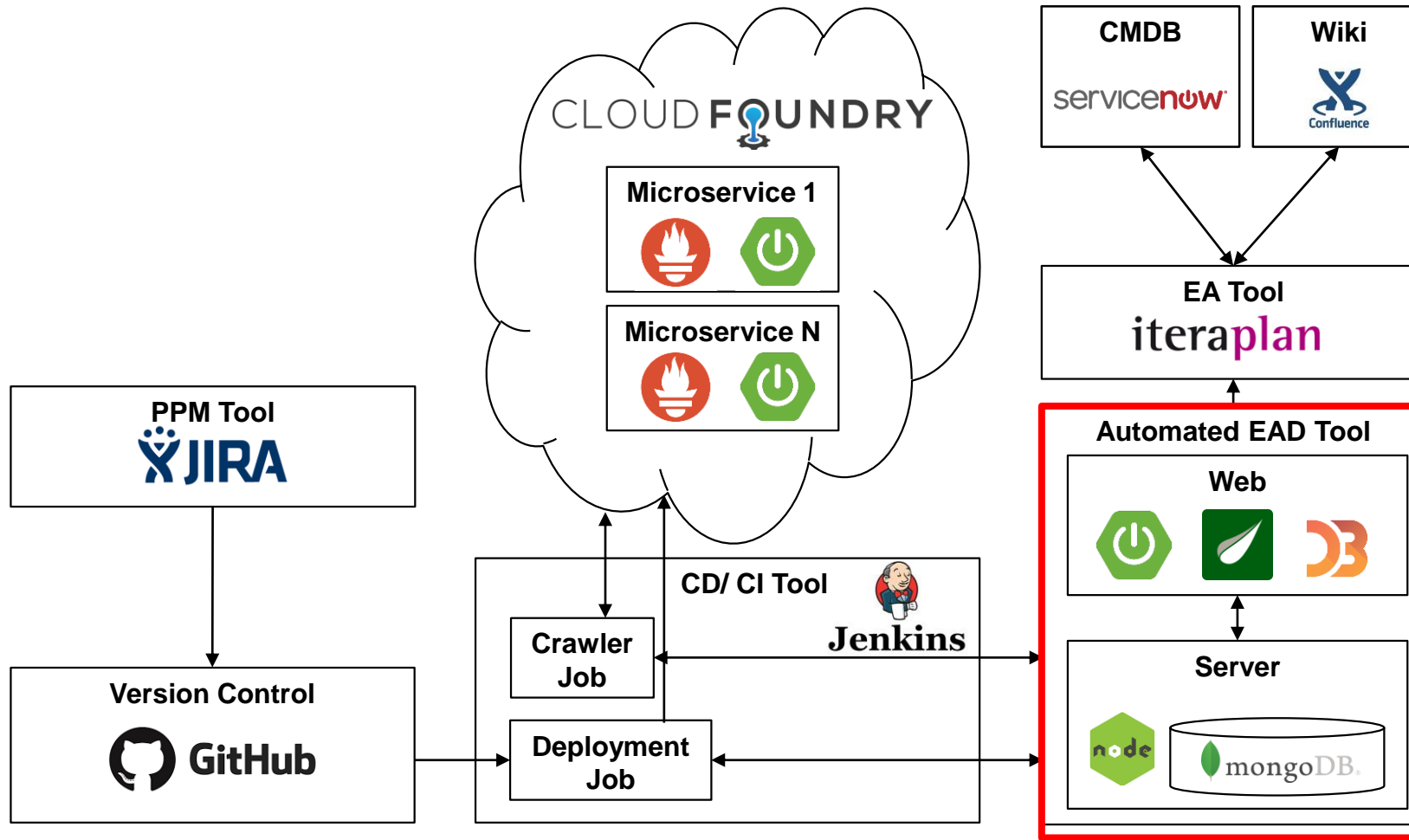
3.4 Class diagram

3.5 Live Demo

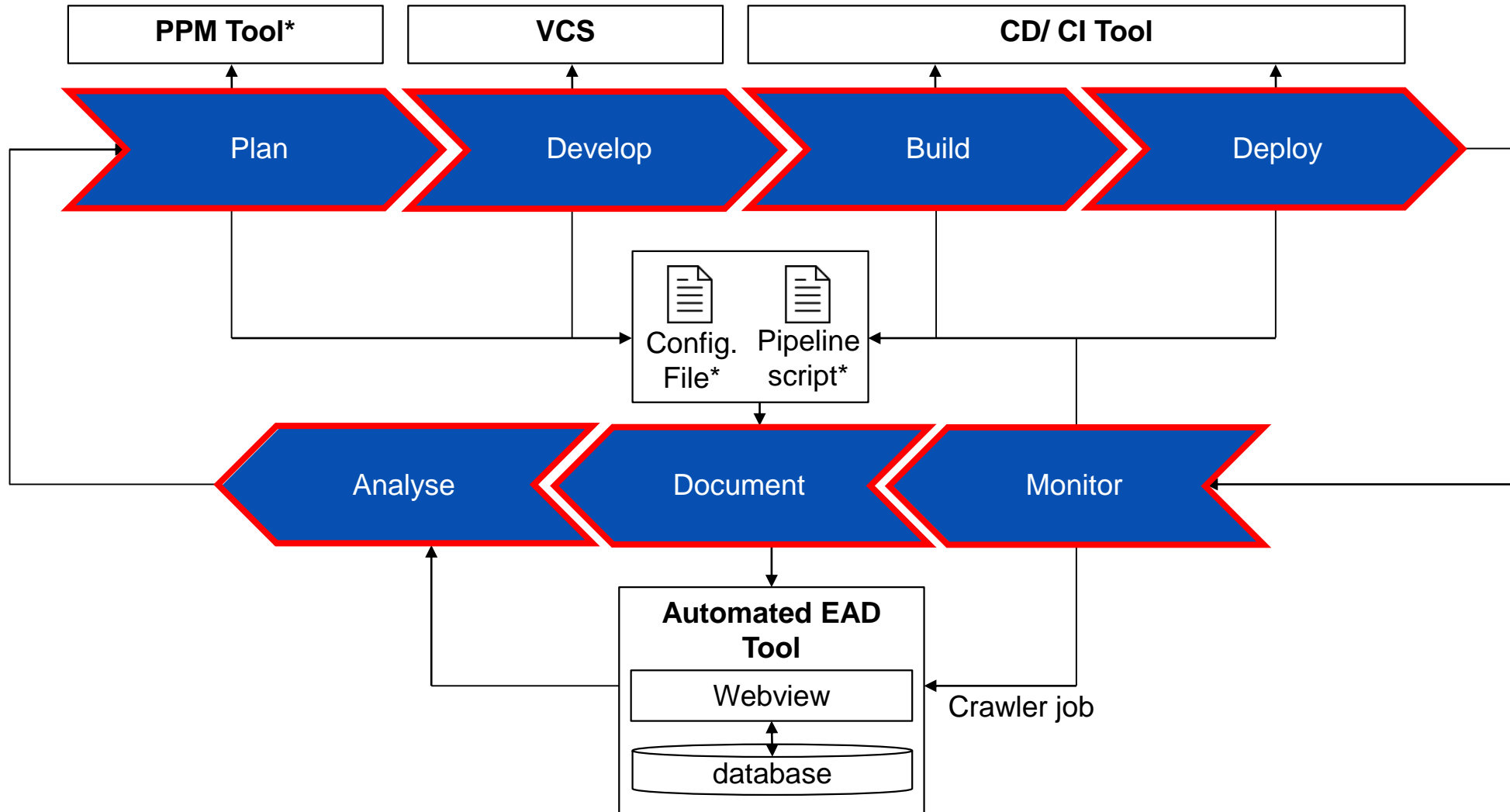
4. Evaluation

5. Conclusion

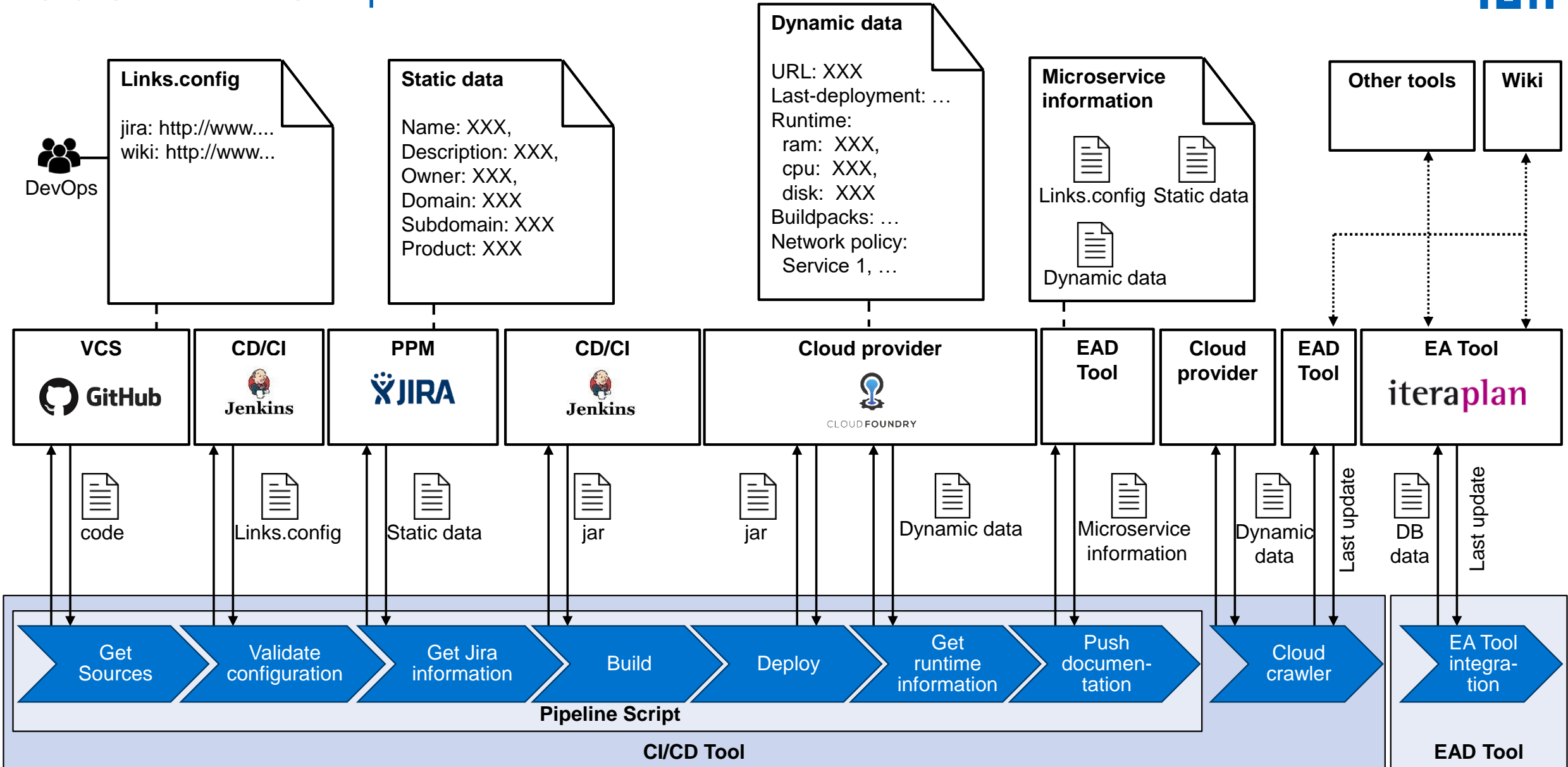
3.1 Solution – Solution Architecture



3.2 Solution - Concept

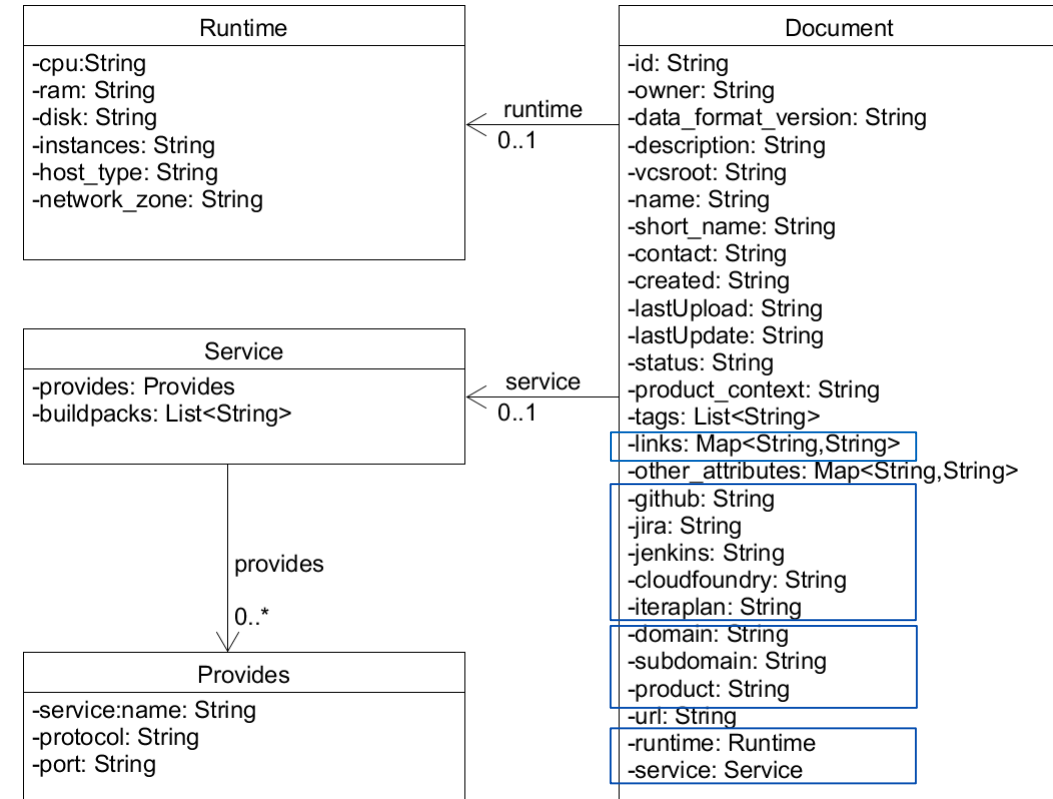


3.3 Solution - Sample scenario



3.4 Solution – Class diagram

- Every artifact is stored as a **Document** in the database
- Every Document can contain a **Runtime** object which gets updated by the cloud-crawler
- Every document can contain a **Service** object
- A Service can contain **buildpacks** (software dependencies) and **Provides** objects
- A **Provides** object represents a connected artifact
- Specific links are modelled as attributes of a Document
- Link to other tools are stored in a key-value-pair attribute
- Business information is also modelled as individual attributes in a Document



3.5 Live Demo



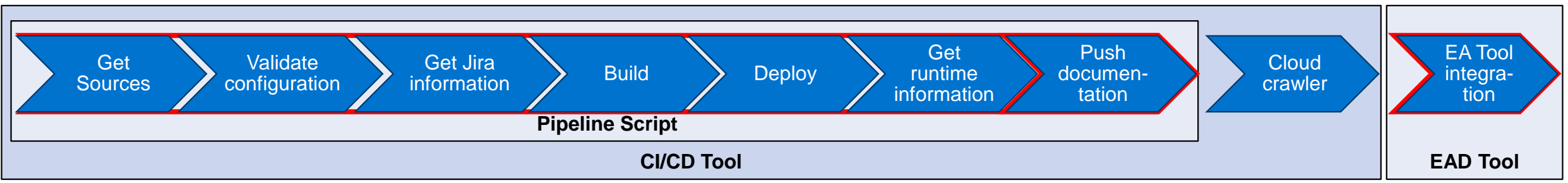
Cloud EA Analytics Overview ES Query Visualizations Feed last update: 2019-01-30 20:12:24 **Export to EA Tool**

Name	Description	Actions
Market Analysis	*order processing *market analysis *pricing analysis	☆ 🗑️ 📄

Sources	Validating Config	Build	Get Basic Jira Information	Get Business Jira Information	Deploy	Get Runtime Information	Push Documentation
2s	156ms	10s	118ms	179ms	1min 0s	5s	816ms

```
[Pipeline] echo
JSONSTRING: {"id": "10100", "key": "ED", "name": "EA Documentation", "owner": "kleehaus", "description": "This is the jira project for the EA Documentation", "short_name": "ED", "type": "software", "domain": "IT", "subdomain": "IT-2", "product": "EA Documentation", "status": "running", "runtime": {"ram": "169.6M of 1G", "cpu": "149.8%", "disk": "142.3M of 1G", "host_type": "cloudfoundry"}, "github": "https://github.com/Nicocovi/Microservice2", "jira": "http://vmmatthes32.informatik.tu-muenchen.de:6000/res/.../api/2/project/ED", "jenkins": "http://131.159.30.173:8081/job/Masterarbeitssoftware-frontend", "iteraplan": "http://vmmatthes32.informatik.tu-muenchen.de:8080/iteraplan/client/#/single/InformationSystem/1705", "service": {"buildpacks": ["client-certificate-mapper=1.8.0_RELEASE", "container-security-provider=1.16.0_RELEASE", "java-main", "java-opts", "java-security", "jvmskill-agent=1.16.0_RELEASE", "open-jd..."], "provides": [{"service_name": "Masterarbeitssoftware-backend"}]}}
```

Salesforce.com	Cloud based CRM (public cloud)	☆ 🗑️ 📄
SAP Classic-P10		☆ 🗑️ 📄
SAP CO-P10 # 6.0	SAP Controlling	☆ 🗑️ 📄
SAP Fi-P10 # 6.0	SAP finance	☆ 🗑️ 📄
SAP RD-P20	SAP Research & Development	☆ 🗑️ 📄



1. Motivation

2. Research

2.1 Research questions

2.2 Existing tools with related functionality

2.3 Literature review

3. Solution

3.1 Solution architecture

3.2 Concept

3.3 Sample scenario

3.4 Class diagram

3.5 Live Demo

4. Evaluation

5. Conclusion

4. Evaluation (1/5)

Goal:

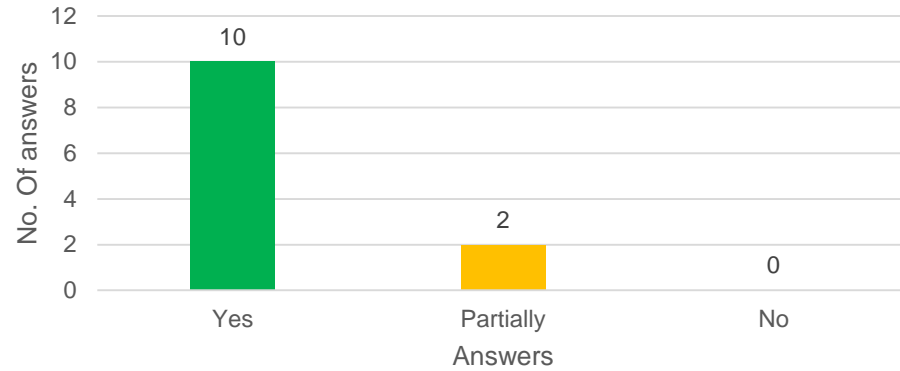
- Evaluate literature findings (requirements and challenges)
- Evaluate the approach to see if it is a valuable solution for improving the automation of Enterprise Architecture Documentation
- Evaluate the implemented prototype
- Identify improvements for the prototype

ID	Role	Exp. years	Enterprise
EA1	Enterprise Architect and Chief Architect	20	E1
EA2	Enterprise Architect	2	E1
EA3	Enterprise Architect	17	E1
EA4	Enterprise Architect and Product Owner	10	E1
EA5	Enterprise Architect	3	E1
EA6	Enterprise Architect and IT Management Expert	20	E2
EA7	Enterprise Architect	18	E1
PO1	Product Owner and Head of Product Architecture	11	E1
PO2	Product Owner	1	E1
PO3	Product Owner	3	E1
EA8	Enterprise Architect and Chief Architect	16	E3
EA9	Enterprise Architect and Chief Architect	30+	E4

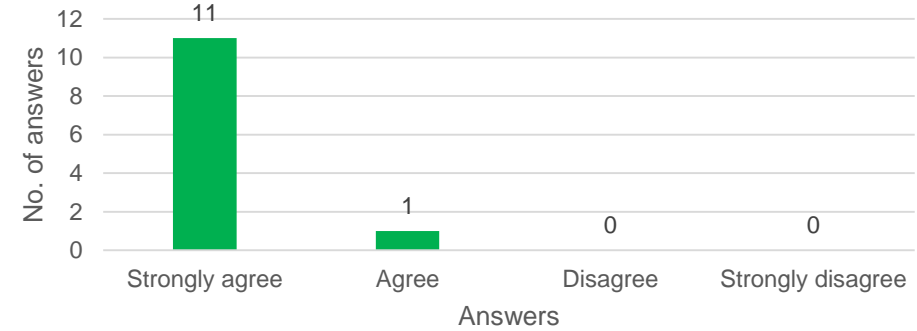
Experts with different roles and from different enterprises were interviewed to eliminate bias

4. Evaluation (2/5) - Enterprise Architecture

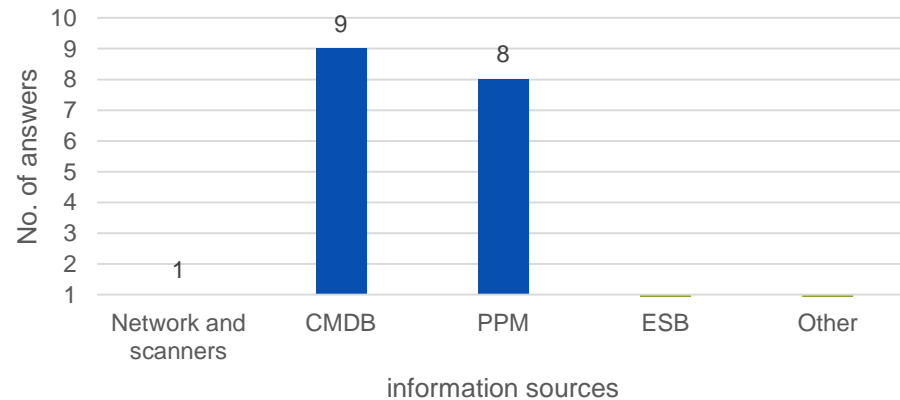
Information in EA Tool is outdated (N=12)



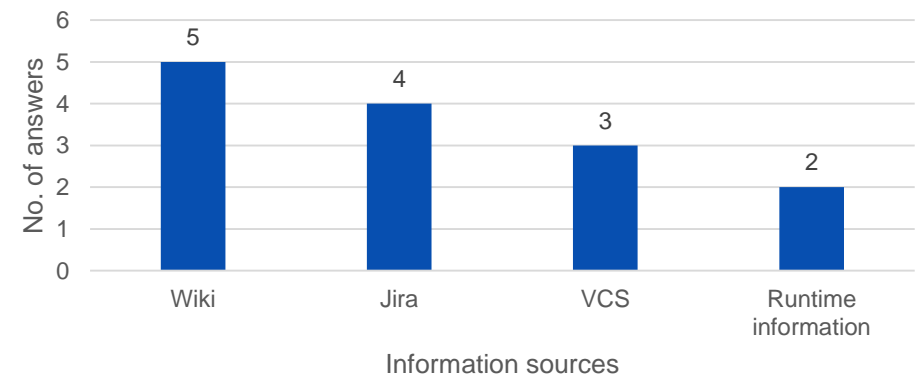
It is a lot of effort to document the IT landscape of your company (N=12)



Usage of EA information sources (N=12)



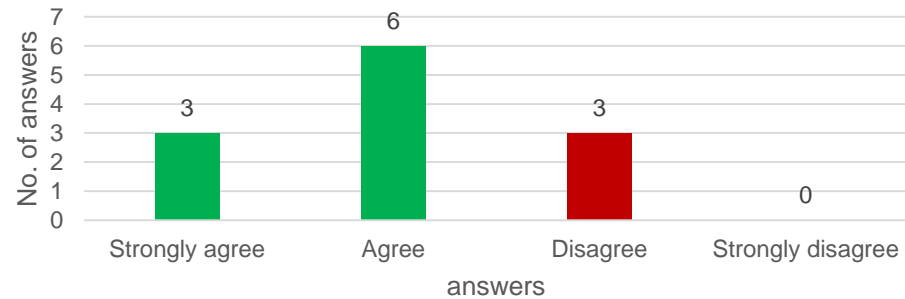
Information sources containing possible EA information (N=12)



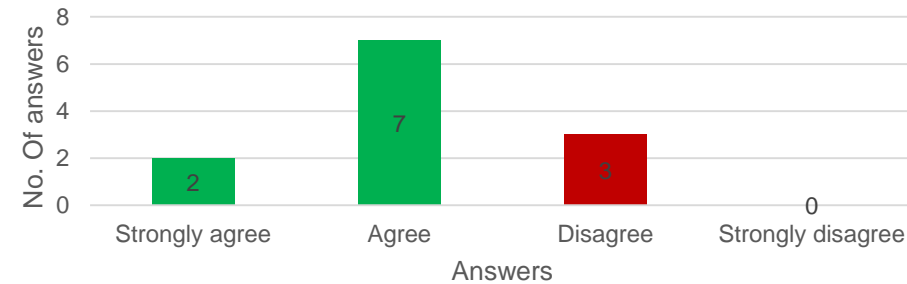
Literature findings were validated in case study

4. Evaluation (3/5) - Approach

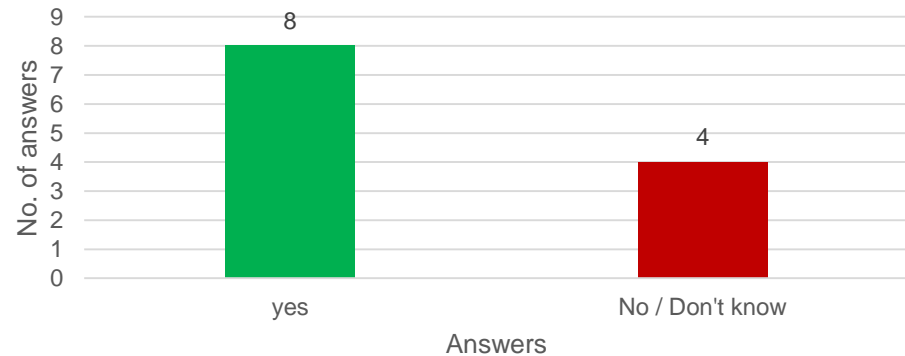
Imposing the team to incorporate a pipeline-script in the repository is easy to establish (N=12)



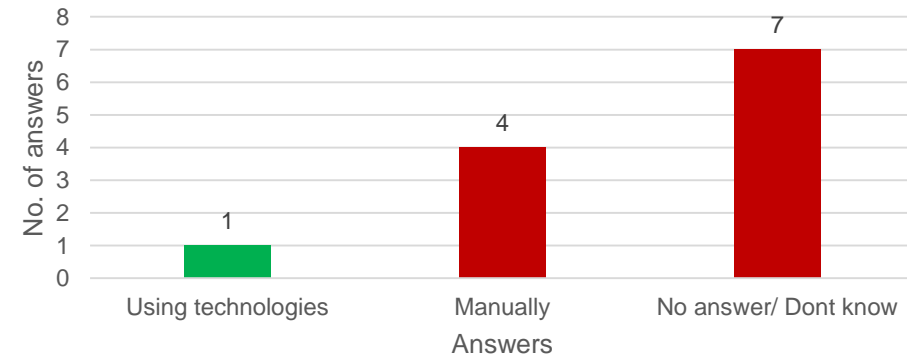
Imposing the team to use a predefined toolchain for the application development is easy to establish (N=12)



Usage of technologies for monitoring applications in the cloud (N=12)

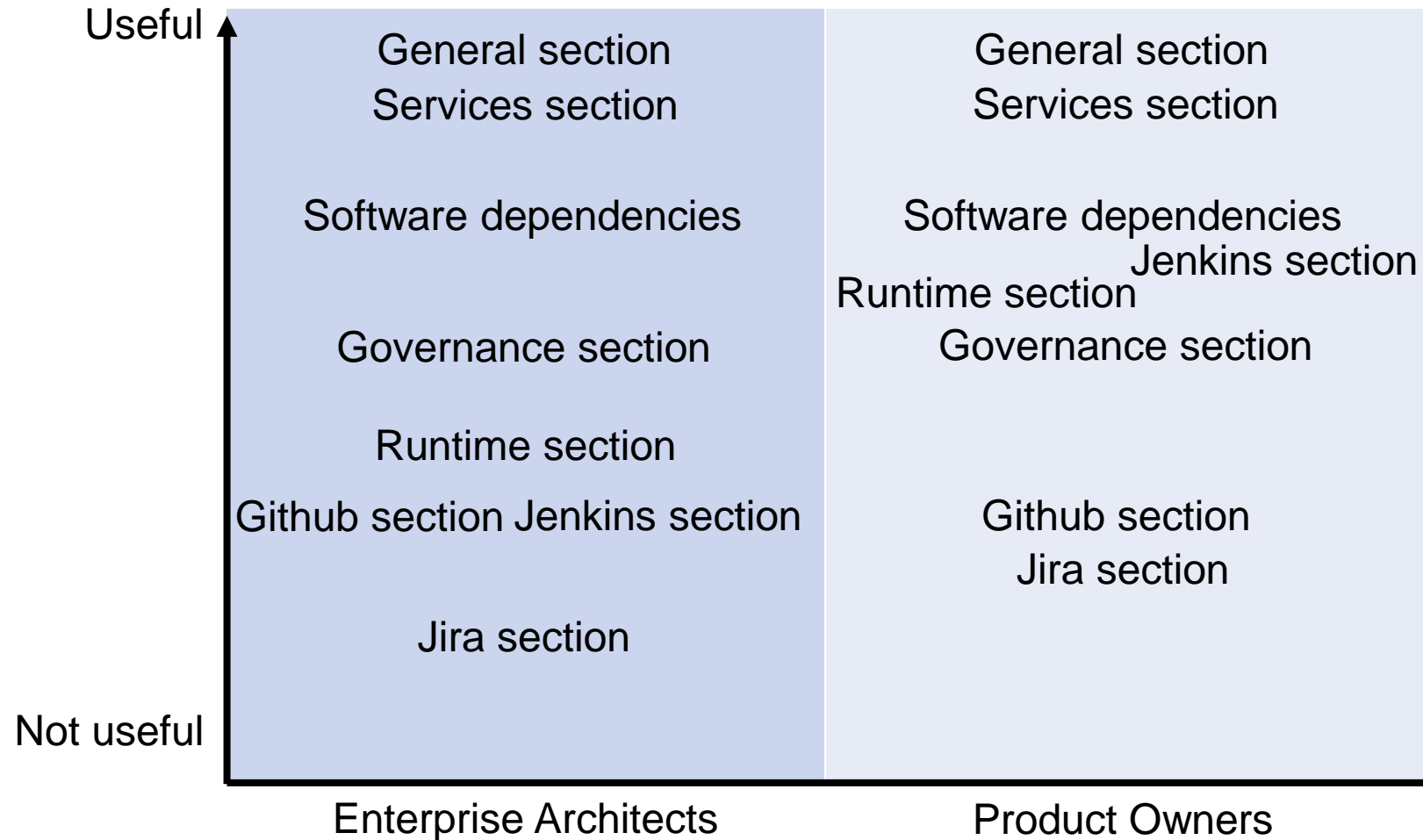


Integration of monitoring technologies for automated EAD (N=12)



All experts stated that the presented process automates the EAD of applications running on a cloud-based environment

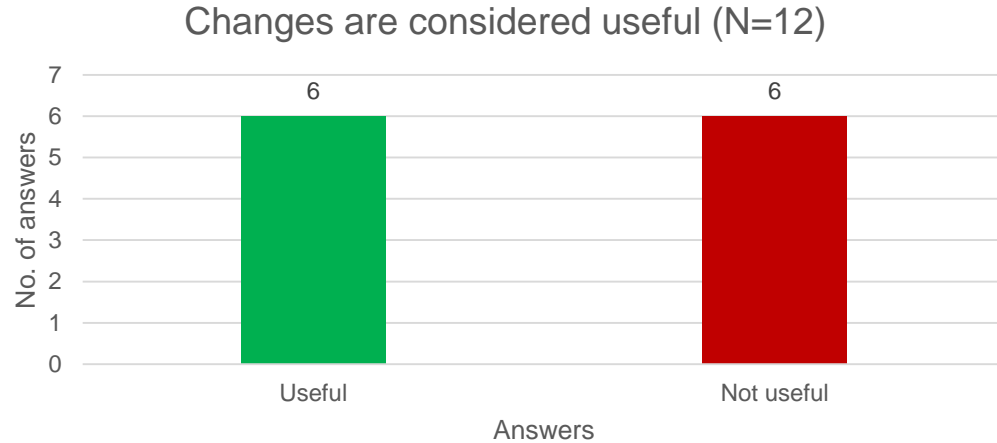
4. Evaluation (4/5) – Prototype sections



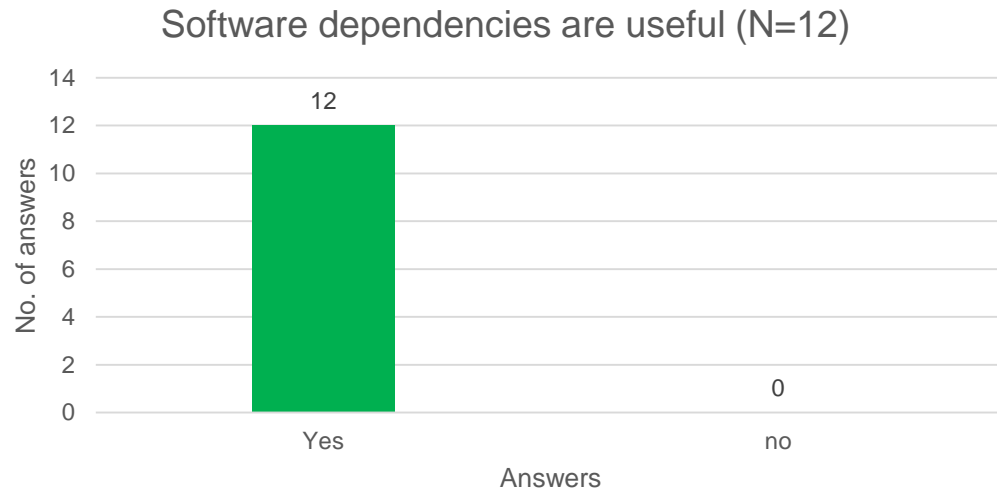
Different sections are perceived as valuable regarding the roles of the user

4. Evaluation (5/5) - Prototype

Not expected results:



How can enterprise architects decide if changes are outdated if changes are not considered useful?



All interviewed industry partners showed a high interest in software dependencies for management of software frameworks.

Agenda

1. Motivation

2. Research

2.1 Research questions

2.2 Existing tools with related functionality

2.4 Literature review

3. Solution

3.1 Solution architecture

3.2 Concept

3.3 Sample scenario

3.4 Class diagram

3.5 Live Demo

4. Evaluation

5. Conclusion

5. Conclusion (1/3) – Summary

Research questions:

- **RQ1:** How to obtain EA relevant information from the runtime behavior of cloud based environments?
- **RQ2:** How to assign the application landscape to business domains?
- **RQ3:** How to automate the assignment process with an integrated toolchain?
- **RQ4:** How does a prototype implementation of the automated documentation process of cloud applications look like?



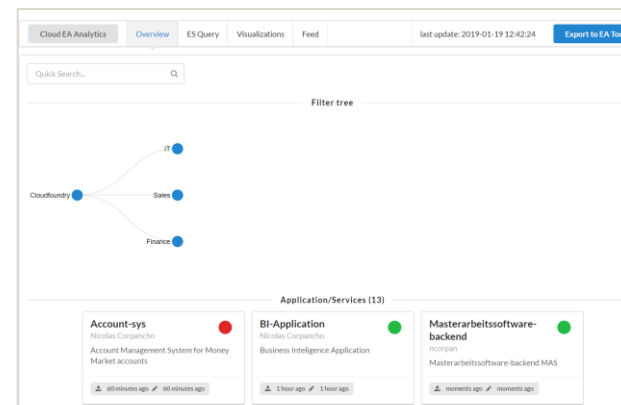
Most cloud infrastructures provide runtime information without agents, shadow IT is prevented. However installing agents unveil further information like API requests.



Either add information in config file or integration of PPM tool in Build-Deployment pipeline. **Assumption:** PPM tool provides domain information



Assignment through configuration file or name mapping
Result: Config file produces further overhead. Mapping via name is more popular, however name needs to be stable!



5. Conclusion (2/3) – Limitations and key findings

Limitations:

- Web-browser does not support ECMAScript 6
- Database component was not supported (MongoDB and ElasticSearch6)
- Server component adaptation for MySQL database component
- No fully EA cloud discovery possible due to access rights

Key findings:

- Several possibilities to (semi-)automate the EAD process
- IT Governance is needed to enable automated EAD (project structure definition, toolchain definition, etc.)

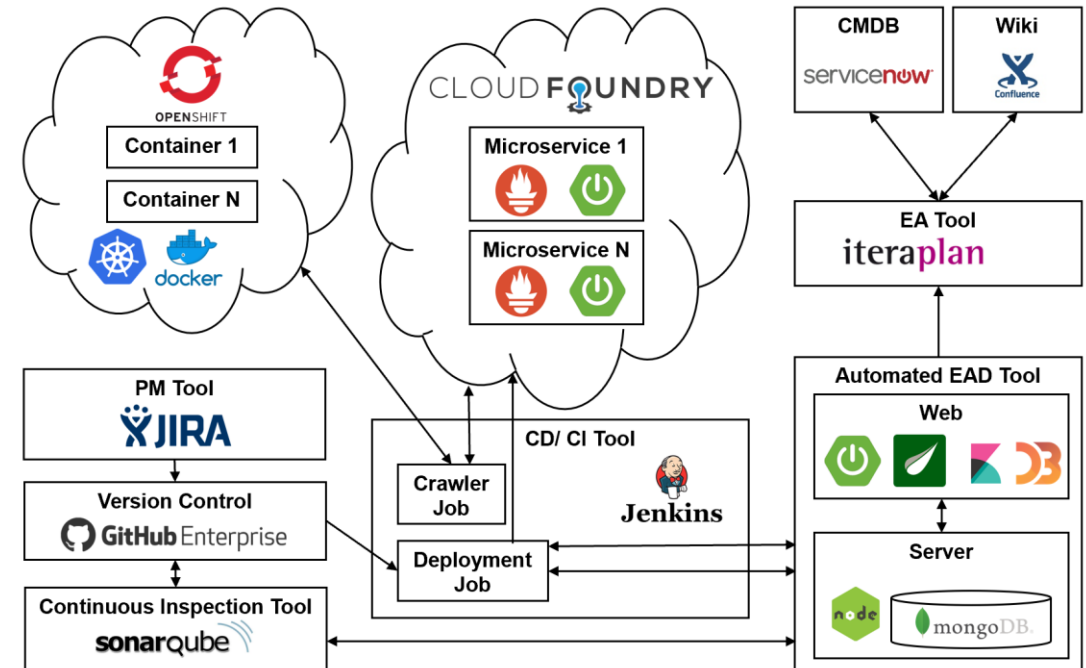
5. Conclusion (3/3) – Outlook

Outlook:

- Integration of a Continuous Inspection Tool (Sonarqube)
- Integration of other cloud environments

Future Use Cases:

- **Cloud readiness** verification through a complete implementation of 12 factor app criteria
- **Elasticity** evaluation through a complete implementation of resilience pattern
- **Data privacy compliance** (GDPR compliance): analysis of stored information
- **Business Impact Analysis (BIA)** of applications via mapping of **Business Process** and **Business Capabilities**



References (1/2)

- [1] M. Hauder, F. Matthes, and S. Roth. “Challenges for automated enterprise architecture documentation.” In: Lecture Notes in Business Information Processing 131 LNBIP (2012), pp. 21–39. ISSN: 18651348. DOI: 10.1007/978-3-642-34163-2_2.
- [2] M. Farwick, R. Breu, M. Hauder, S. Roth, and F. Matthes. “Enterprise architecture documentation: Empirical analysis of information sources for automation.” In: Proceedings of the Annual Hawaii International Conference on System Sciences(2013), pp. 3868–3877. ISSN: 15301605. DOI: 10.1109/HICSS.2013.200.
- [3] S. Roth, M. Hauder, M. Farwick, R. Breu, and F. Matthes. “Enterprise architecture documentation: Current practices and future directions.” In:Wi2013 (2013),pp. 912–925. ISSN: 00219673. DOI:10.1148/rg.327125019.
- [4] M. Farwick, B. Agreiter, R. Breu, M. Häring, K. Voges, and I. Hanschke. “Towards living landscape models: Automated integration of infrastructure cloud in Enterprise Architecture Management.” In:Proceedings - 2010 IEEE 3rd InternationalConference on Cloud Computing, CLOUD 2010(2010), pp. 35–42. ISSN: 2159-6182. DOI:10.1109/CLOUD.2010.20.
- [5] M. Buschle, S. Grunow, F. Matthes, M. Ekstedt, and S. Roth. Automating Enterprise Architecture Documentation using an Enterprise Service Bus Americas Conference onInformation Systems Automating Enterprise Architecture Documentation using anEnterprise Service Bus. Tech. rep. 1. 2012.
- [6] H. Holm, M. Buschle, R. Lagerström, and M. Ekstedt. “Automatic data collection for enterprise architecture models.” In:Software and Systems Modeling13.2 (2014),pp. 825–841.issn: 16191374.doi:10.1007/s10270-012-0252-1.

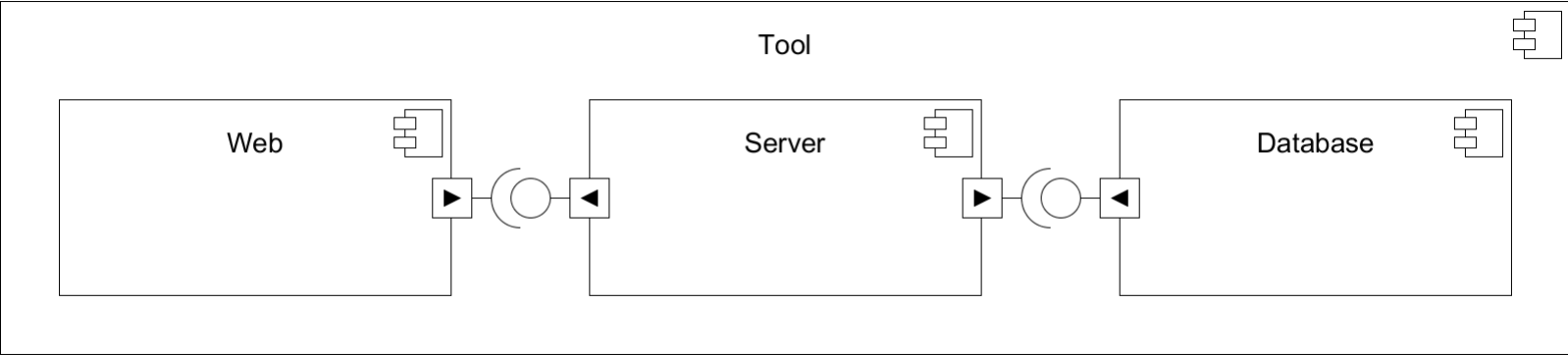
- [7] M. Välja, R. Lagerström, M. Ekstedt, and M. Korman. “A requirements based approach for automating enterprise IT architecture modeling using multiple datasources.” In: Proceedings of the 2015 IEEE 19th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations, EDOCW 2015(2015), pp. 79–87. ISSN: 2325-6583. DOI:10.1109/EDOCW.2015.33.
- [8] M. Farwick, C. M. Schweda, R. Breu, and I. Hanschke. “A situational method for semi-automated enterprise architecture documentation (SoSyM abstract).” In: 2015 ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems, MODELS 2015 - Proceedings(2015), p. 448. ISSN: 16191374. DOI:10.1109/MODELS.2015.7338278.
- [9] P. Johnson, M. Ekstedt, and R. Lagerstrom. “Automatic Probabilistic Enterprise IT Architecture Modeling: Dynamic Bayesian Networks Approach.” In: Proceedings- IEEE International Enterprise Distributed Object Computing Workshop, EDOCW2016-Septe (2016), pp. 122–129. ISSN: 15417719. DOI:10.1109/EDOCW.2016.7584351.
- [10] J. Landthaler, Ö. Uludağ, G. Bondel, A. Elnaggar, S. Nair, and F. Matthes. “A machine learning based approach to application landscape documentation.” In: Lecture Notes in Business Information Processing 335 (2018), pp. 71–85. ISSN: 18651348. DOI:10.1007/978-3-030-02302-7_5.
- [11] J. Bogner and A. Zimmermann. “Towards Integrating Microservices with Adapt-able Enterprise Architecture.” In: Proceedings - IEEE International Enterprise Distributed Object Computing Workshop, EDOCW2016-Septe (2016), pp. 158–163. ISSN: 15417719. DOI:10.1109/EDOCW.2016.7584392.

Thank you for your attention!

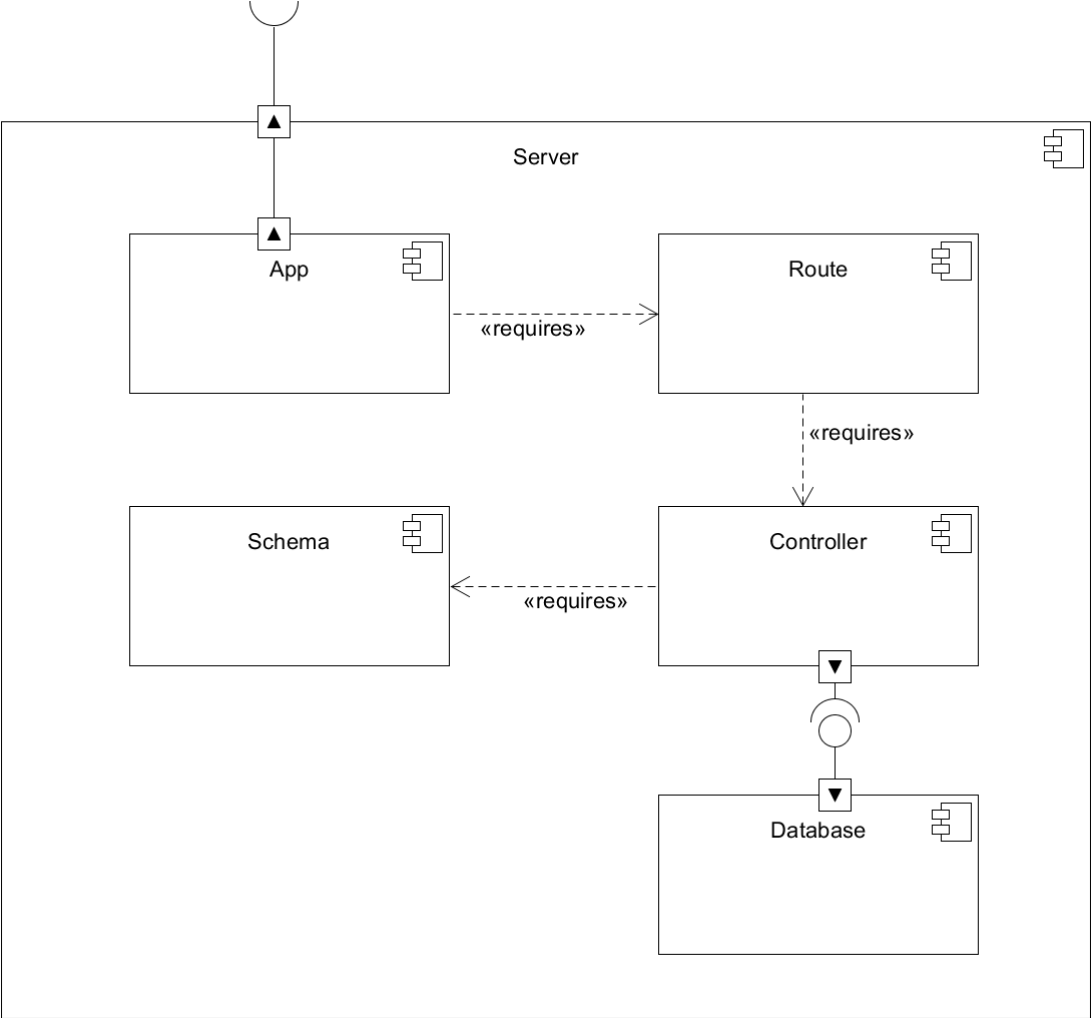
Do you have any questions?

Backup

Component diagram



Server component



Optional Prerequisites

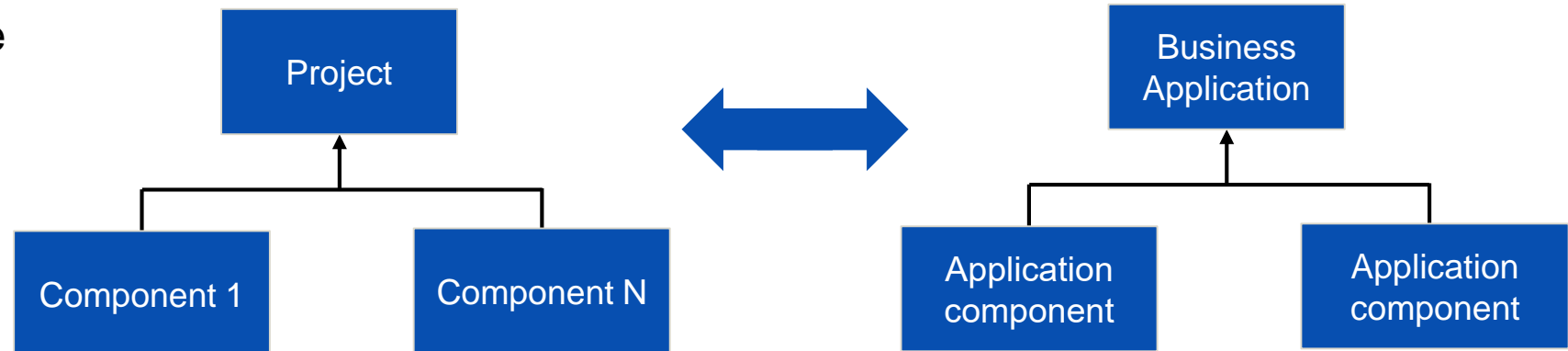
1. Configuration file containing link to different tools

```
Links.config  
  
jira: http://www....  
cmdb:http://www....  
wiki: http://www...
```

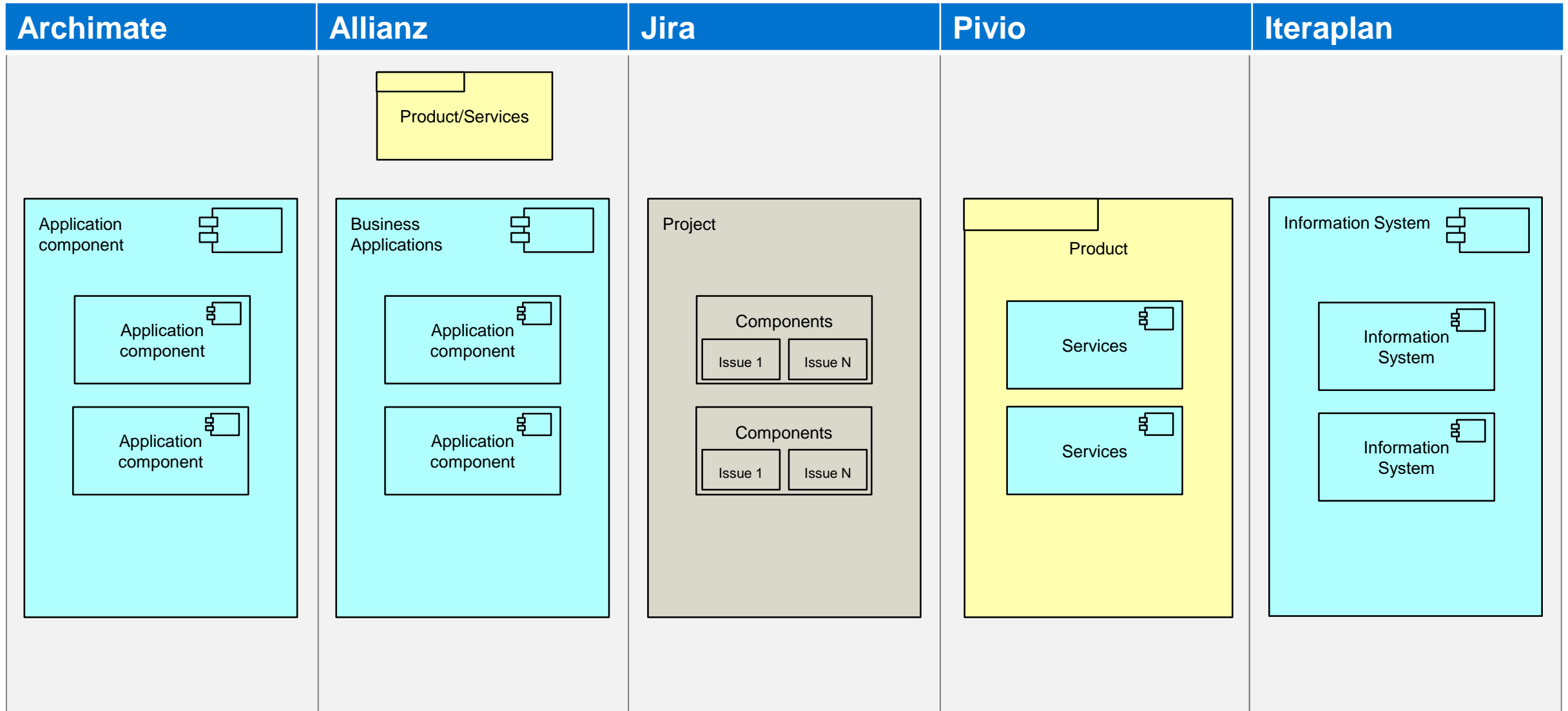
2. Groovy script in repository to enable build pipeline

```
pipeline.groovy  
  
node {  
  def xxx =  
    stage('Sources') {...}  
    stage('Validate...'){...}  
  ...  
}
```

3. Jira project complaince



Definitions mapping



Groovy script



```
def callPost(String urlString, String queryString) {
    def url = new URL(urlString)
    def connection = url.openConnection()
    connection.setRequestMethod("POST")
    connection.doInput = true
    connection.doOutput = true
    connection.setRequestProperty("content-type", "application/json;charset=UTF-8")

    def writer = new OutputStreamWriter(connection.outputStream)
    writer.write(queryString.toString())
    writer.flush()
    writer.close()
    connection.connect()

    new groovy.json.JsonSlurper().parseText(connection.content.text)
}
node {
    deleteDir()
    stage('Sources') {
        checkout([
            $class      : 'GitSCM',
            branches    : [[name: "refs/heads/master"]],
            extensions   : [[class: 'CleanBeforeCheckout', localBranch: "master"]],
            userRemoteConfigs: [[
                credentialsId: 'cbf178fa-56ee-4394-b782-36eb8932ac64',
                url          : "https://github.com/Nicocovi/MS-Repo"
            ]]
        ])
    }
    dir("") {
        stage("Build"){
            sh "gradle build"
        }
    }
    stage("Get Jira Information"){
        //TODO
    }
}
```

```
stage('Deploy') {
    def branch = ['master']
    def name = "sping-microservice1"
    def path = "build/libs/gs-spring-boot-0.1.0.jar"
    def manifest = "manifest.yml"

    if (manifest == null) {
        throw new RuntimeException('Could not map branch ' + master + ' to a manifest file')
    }
    withCredentials([[
        $class      : 'UsernamePasswordMultiBinding',
        credentialsId : '98c5d653-dbd4-4b52-81ba-50c2ac04e4f1',
        usernameVariable: 'CF_USERNAME',
        passwordVariable: 'CF_PASSWORD'
    ]]) {
        sh 'cf login -a https://api.run.pivotal.io -u $CF_USERNAME -p $CF_PASSWORD --skip-ssl-validation'
        sh 'cf target -o ga72hib-org -s masterarbeit'
        sh 'cf push sping-microservice1 -f '+manifest+' --hostname '+name+' -p '+path
    }
}
stage("Push Documentation"){
    try {
        callPost("http://192.168.99.100:9123/document", "{\"id\": \"0987654321\", \"name\": \"Kick-off-App\", \"owner\": \"Nico\", \"description\": \"bla\", \"short_name\": \"serviceAZ12\", \"type\": \"service\"}") //Include protocol
    } catch(e) {
        // if no try and catch: jenkins prints an error "no content-type" but post request succeeds
    }
}
}
}
```

2.3 Open source project - Pivio

The screenshot displays the Pivio service registry interface. At the top, there are navigation tabs for 'pivio', 'Overview', 'Query', and 'Feed'. Below this is a search bar labeled 'Quick Search...'. The main content area shows 'Matching Artifacts (16)' with four service cards:

- Awesome Microservice**: Owner: lambda, Type: Simple microservice, updated 'vor 1 Monat'.
- Demokick**: Owner: Miriam, Type: Simple microservice, updated 'vor 2 Tagen'.
- Microservice 1**: Owner: Nicolas, Type: bla bla bla, updated 'vor 2 Wochen'.
- Microservice 2**: Owner: Nicolas, Type: bla bla bla, updated 'vor 2 Wochen'.

The bottom section shows a detailed view of the 'Awesome Microservice' card. It includes a 'General' tab with a description, owner (lambda), and type (service). A 'Service' tab shows 'Provides' and 'Depends on' relationships. A 'Runtime' tab shows resource requirements like RAM, CPU, Disk, Host, and Network Zone. A 'Delete Document' button is visible at the bottom left.

Pivio:

- Service registry for humans
- To have a catalogue of the available services
- Running in cloud environments
- Developed by Chief Architect of E-Post Development GmbH

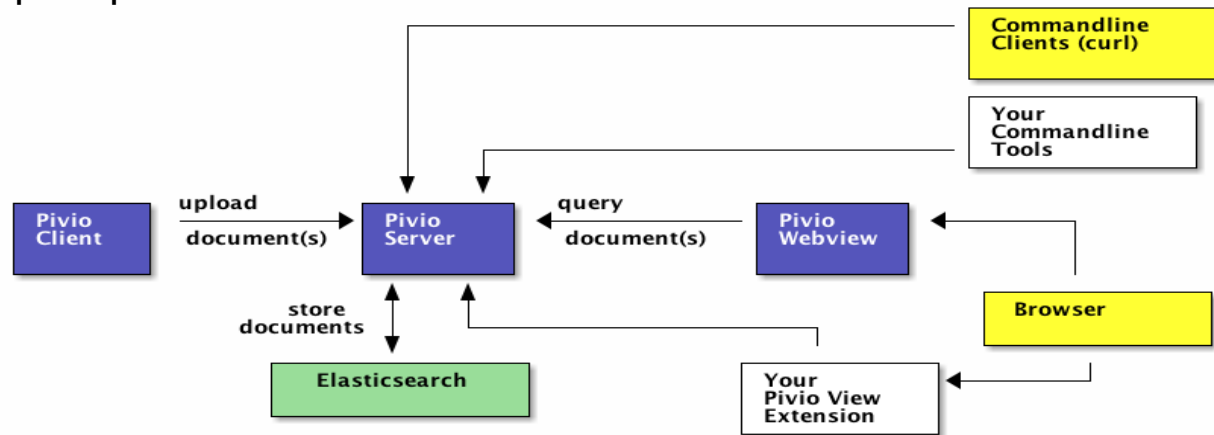


- **Integration of discovery**
- Metamodel focus on microservices
- Dynamic metamodel enabled via schemaless
- Runtime information integration



- No Continuous Delivery Integration
- No support of EA discovery

- What is Pivio ?
Pivio is a service registry for humans.
- Why Pivio ?
 - Overview for platforms, especially for microservice environment.
 - Reusability of services
 - A growing number of services means also a challenge not only for developers.
 - Which service runs where? What does it do? Who is responsible for that?
- Concept of pivio:



Legend



Pivio data model

Pivio needs certain mandatory fields:

- **id:** Unique id in pivio.
- **name:** The name of the artefact.
- **short_name:** A very brief name for the service.
- **type:** The type of this artefact. Values could be **service**, **library** or **mobile_app**.
- **owner:** Which team is responsible for this artefact.
- **description:** What does this service do?

4. Derived EAD requirements from the literature review and case study (1/3)

ID	Description	Source
Data Quality Requirements		
DC1	No maintenance of data. The data is not considered uptodate. The system must provide mechanisms to ensure data actuality.	[1], [2], [3], [4], [6], [7], [9], [10], [11]
DC2	The data granularity of the integrated information sources is too granular. The system must provide mechanisms to align the granularity of data.	[1], [2], [3], [6], [7]
DC3	Data completeness: EA information is scarce	[1], [2], [3], [6], [7]
DC4	Data correctness: Error proned data due to manual gathering	[1], [2], [3], [6], [7], [9], [10]
DC5	The system must be able to provide relationship information between and within the EA layers	Case study evaluation
Functional System Requirements		
FR1	The system must be able to calculate the defined KPIs from runtime information	[4], [8], [11]
ADR5	Additional KPIs calculation: TCO, MTBF, MMTR, MTTF and real time data of users on the individual applications	Case study evaluation
FR2	Integration of different information sources	[4], [8], [11]
FR3	EA Tool needs an public API for an integration of several information sources	[1], [3], [6], [8]
FR4	Dynamic metamodel: Adaptable metamodel of system	[1], [2], [3], [4],[5], [6], [7], [8], [11]

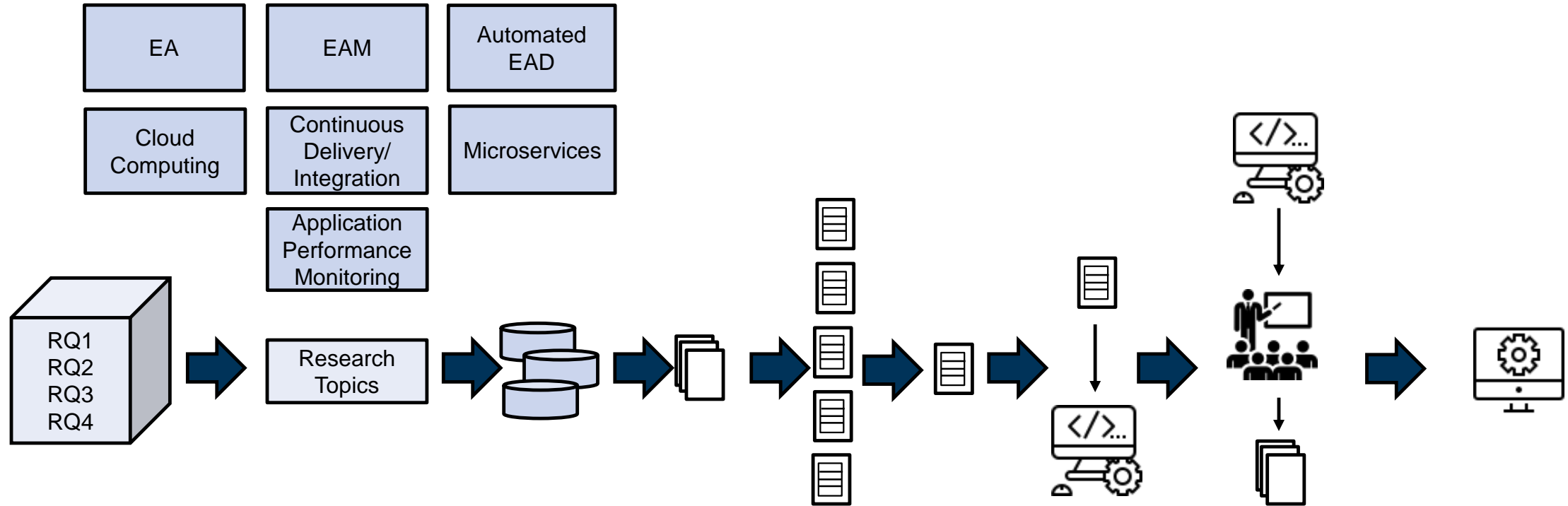
4. Derived EAD requirements from the literature review and case study (2/3)

ID	Description	Source
Organizational Requirements		
ADR1	Business Impact Analysis of applications	Case study evaluation
OR1	Business added value: The system must provide a added value for the business (e.g. ROI)	[1], [3], [4]
ADR2	Data privacy compliance (GDPR compliance)	Case study evaluation
Integration/Data Source Requirements		
ADR6	Integration of cloud environments (PaaS and SaaS)	[4], [8], [11]
ADR3	Automated verification of the 12 factor app	Case study evaluation
ADR4	Automated verification of a resilience pattern	Case study evaluation

2. Additional requirements from literature

ID	Description	Source
Architectural Requirements		
AR1	The collection of EA data must be federated from the repositories of the data owners (departments etc.)	Fischer et al., 2007; Farwick et al., 2010
Organizational Requirements		
OR2	An organizational process must be in place that regulates the maintenance of EA Models	Fischer et al., 2007; Moser et al., 2009; Hanschke, 2009
OR3	Each data source must have an owner	Fischer et al., 2007; Hanschke, 2009,
Integration/Data Source Requirements		
IR1	The system must be able to detect changes in the real world enterprise architecture	Moser et al., 2009; ter Doest and Lankhorst, 2004
IR2	The system must have a machine understandable internal data structure	Tanner and Feridun, 2009
Non-functional Requirements		
NFR1	The system must scale for large data input	Hafner and Winter, 2008
Data Quality Requirements		
DC4	The system must provide mechanisms that allow for the automated propagation of changes	Dam et al., 2010
DC5	The system must be able to identify and resolve data identity conflicts from different sources via identity reconciliation	Fischer et al., 2007
DC6	The system must provide mechanisms that help the QA team to ensure data consistency	Hafner and Winter, 2008

2.1 Research methodology



1. Scope of Research	2. Topic conceptualization	3. Literature search	4. Literature analysis	1. Solution suggestion	2. Implementation	3. Evaluation	4. Results
Deliverable: RQs on automated documentation of Business Domain assignments and cloud application information	Deliverable: Relevant search terms	Deliverable: Relevant sources	Deliverable: Comparison of different approaches on automated EAD	Deliverable: Approach of an automated EAD	Deliverable: Development of a prototype	Deliverable: Feedback of the approach and feedback	Deliverable: Improved Prototype for an automated EAD

Screenshots

Pivio overview

Cloud EA Analytics | Overview | ES Query | Visualizations | Feed | last update: 2019-01-19 12:42:24 | [Export to EA Tool](#)

Product6

Filter tree

```
graph LR; Cloudfoundry((Cloudfoundry)) --- IT((IT)); Cloudfoundry --- Sales((Sales)); Cloudfoundry --- Finance((Finance)); IT --- IT1((IT-1)); IT1 --- Product2((Product2)); IT1 --- Product6((Product6)); Product6 --- MSB((Masterarbeitssoftware-backend)); Product6 --- MSF((Masterarbeitssoftware-frontend)); MSB --- github1((github)); MSB --- jenkins1((jenkins)); MSF --- jira((jira)); MSF --- github2((github)); MSF --- jenkins2((jenkins));
```

Application/Services (2)

Masterarbeitssoftware-backend ●

ncorpan

Masterarbeitssoftware-backend MAS

↑ moments ago ✎ moments ago

Masterarbeitssoftware-frontend ●

ncorpan

Masterarbeitssoftware-frontend MIC

↑ moments ago ✎ moments ago

Detailed View (1/7)

General

Masterarbeitssoftware-frontend

Description

Masterarbeitssoftware-frontend MIC

Shortname (MIC) Type software

Business specific

Domain IT Subdomain IT-1 Product Product6

Owner ncorpan

Changes

vor 2 Wochen vor 2 Wochen

Links

- Jira <http://vmmatthes32.informatik.tu-muenchen.de:6000/rest/api/2/project/ED>
- Github <https://github.com/Nicocovi/Microservice2>
- Jenkins <http://131.159.30.173:8081/job/Masterarbeitssoftware-frontend>
- CF-link masterarbeitssoftware-frontend.cfapps.io
- Iteraplan null

Additional Information

key ED

Runtime

Instances 1/1 RAM 142.7M of 1G CPU 0.4% Disk 142.2M of 1G

Host cloudfoundry

Running costs \$0.03 per hour

Metrics monitoring (Agent)

URL masterarbeitssoftware-frontend.cfapps.io

Prometheus metrics masterarbeitssoftware-frontend.cfapps.io/prometheus

Response time 1.15676799E8

HTTP Calls 13586

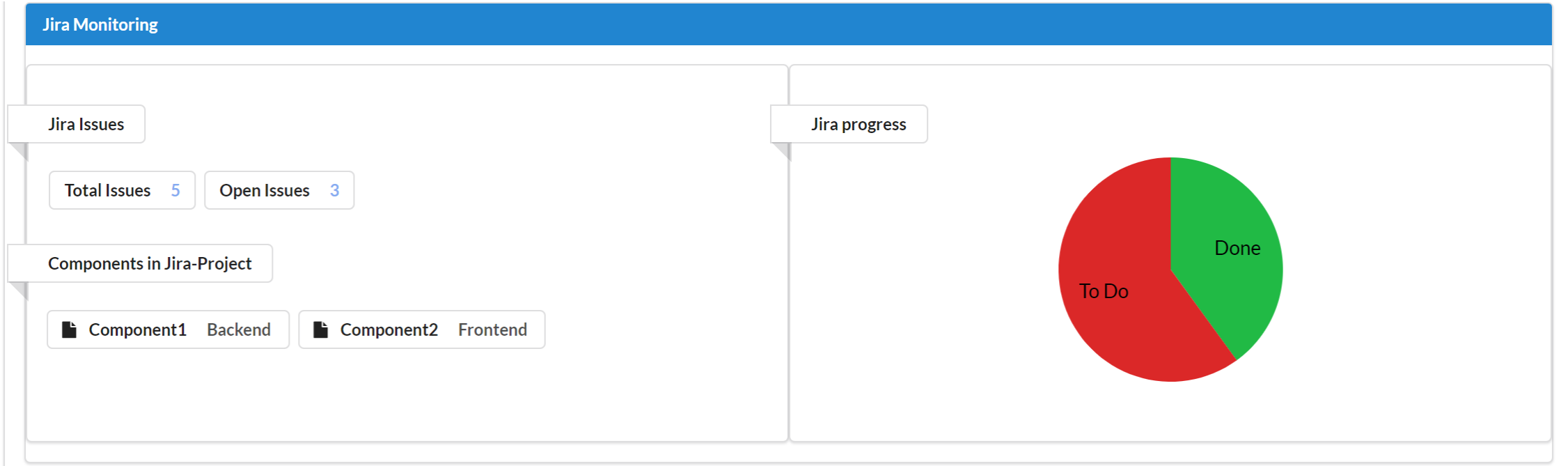
Services

Service name Masterarbeitssoftware-backend

Software Dependencies

Buildpacks

- client-certificate-mapper=1.8.0_RELEASE
- container-security-provider=1.16.0_RELEASE
- java-main
- java-opts
- java-security
- jvmskill-agent=1.16.0_RELEASE
- open-jd...

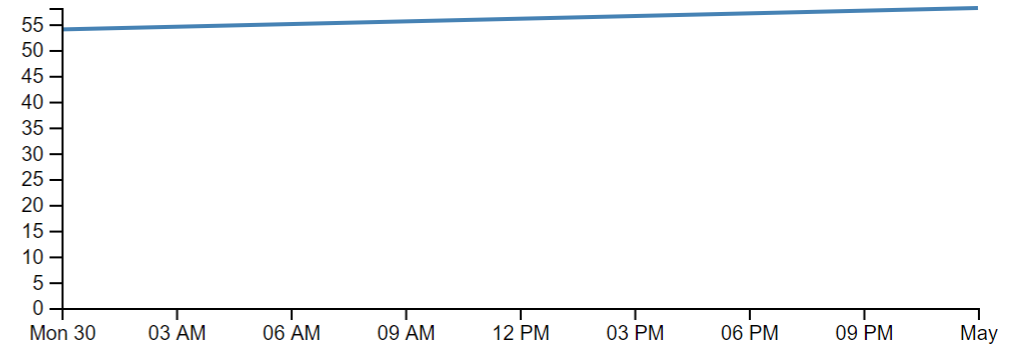


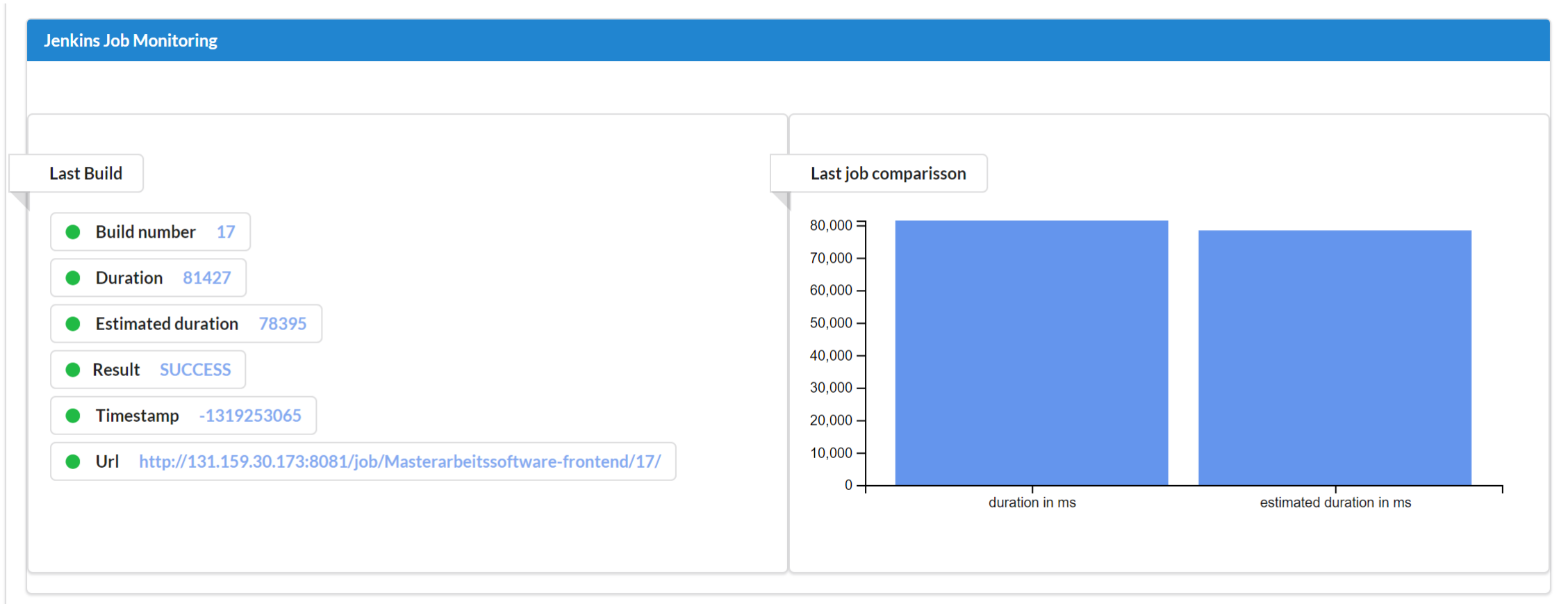
Github Monitoring

Contributors 1

Lines of code 1064

Commit activity





Detailed View (5/6)

Governance Monitoring

The twelve-factor methodology

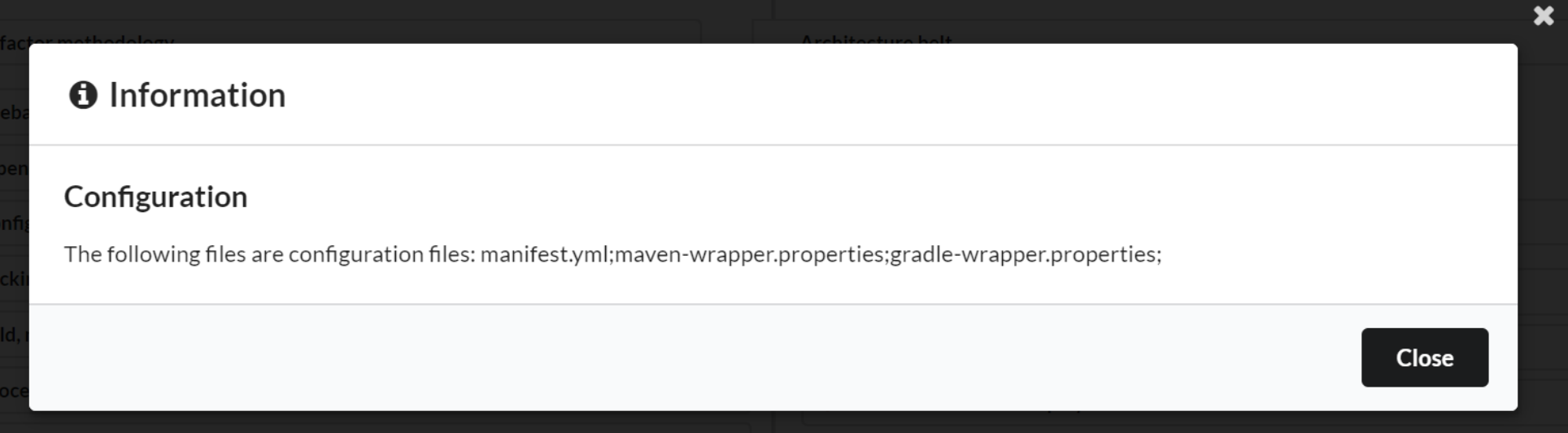
- I. Codebase Yes
- II. Dependencies Yes
- III. Configuration Yes
- IV. Backing services Yes
- V. Build, release, run Yes
- VI. Processes Yes
- VII. Port binding No
- VIII. Concurrency Not implemented
- IX. Disposability Not implemented
- X. Dev/prod parity No
- XI. Logs Yes
- XII. Admin processes Not implemented

Architecture belt

No architecture belt for this application

Resilience

- I. AZD Cloud No
- II. Redundancy No
- III. Zero Downtime deployment No
- IV. Retry No
- V. Isolation No
- VI. Caching No
- VII. Fallback No
- VIII. Loose Coupling Not implemented



The image shows a dark-themed application window with a white information dialog box. The dialog box has a title bar with an information icon and the text "Information". Below the title bar, the word "Configuration" is displayed. The main content of the dialog box contains the text: "The following files are configuration files: manifest.yml;maven-wrapper.properties;gradle-wrapper.properties;". At the bottom right of the dialog box, there is a dark button with the text "Close".

i Information


Configuration

The following files are configuration files: manifest.yml;maven-wrapper.properties;gradle-wrapper.properties;

Close

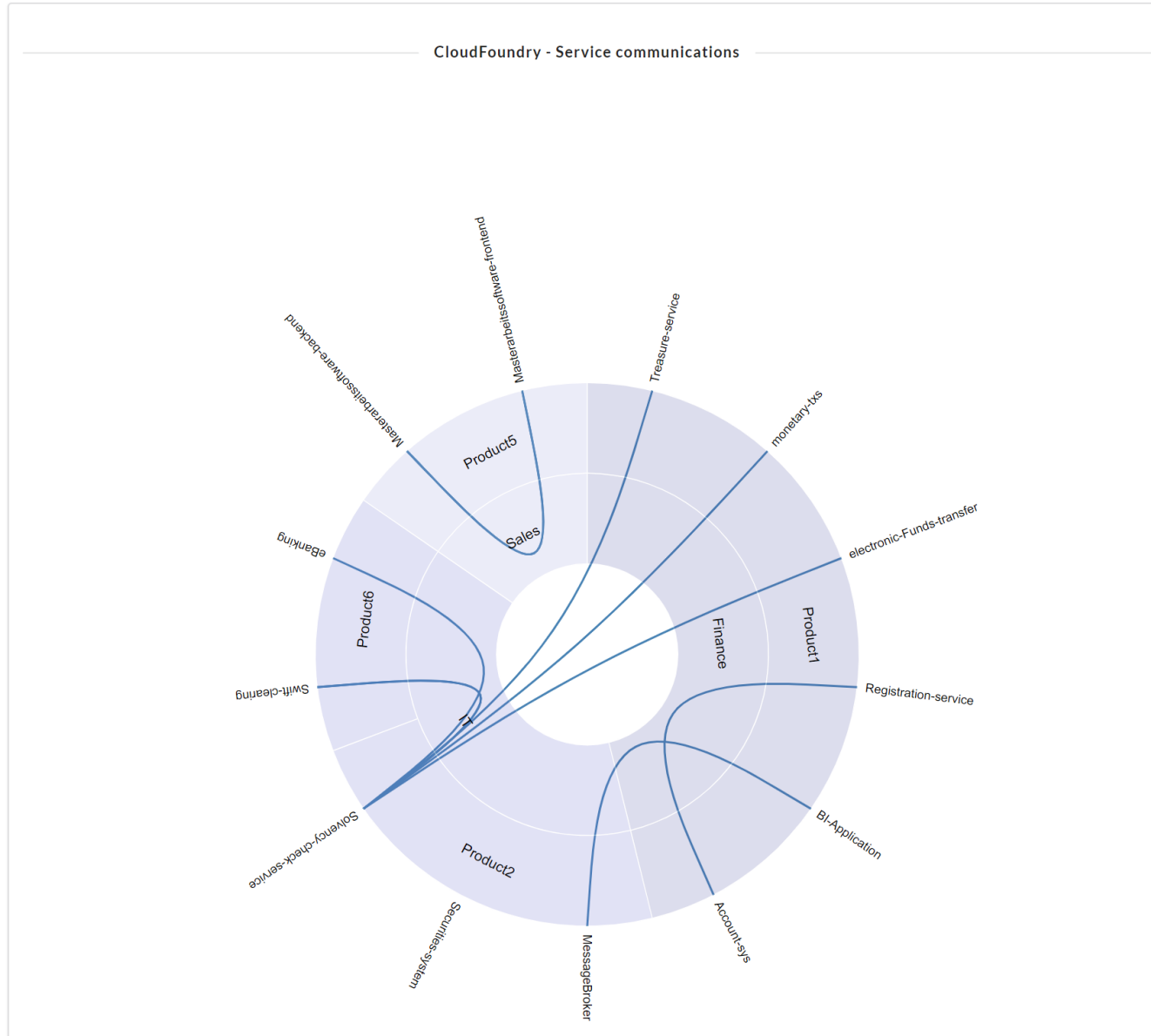
Actions

Functions

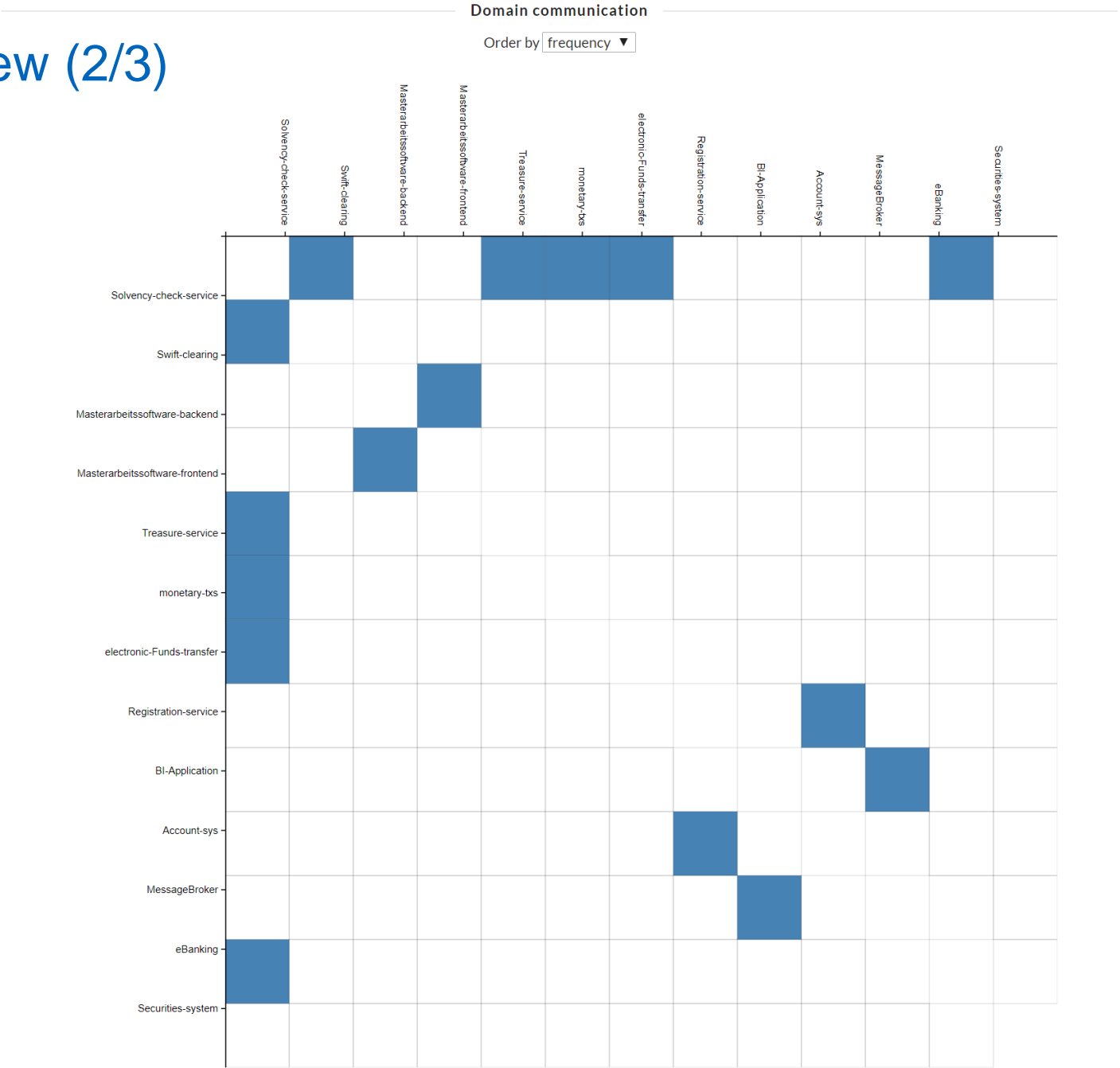
 Delete Application

Visualizations View (1/3)

Cloud EA Analytics Overview ES Query **Visualizations** Feed last update: 2019-01-19 13:12:23 [Export to EA Tool](#)



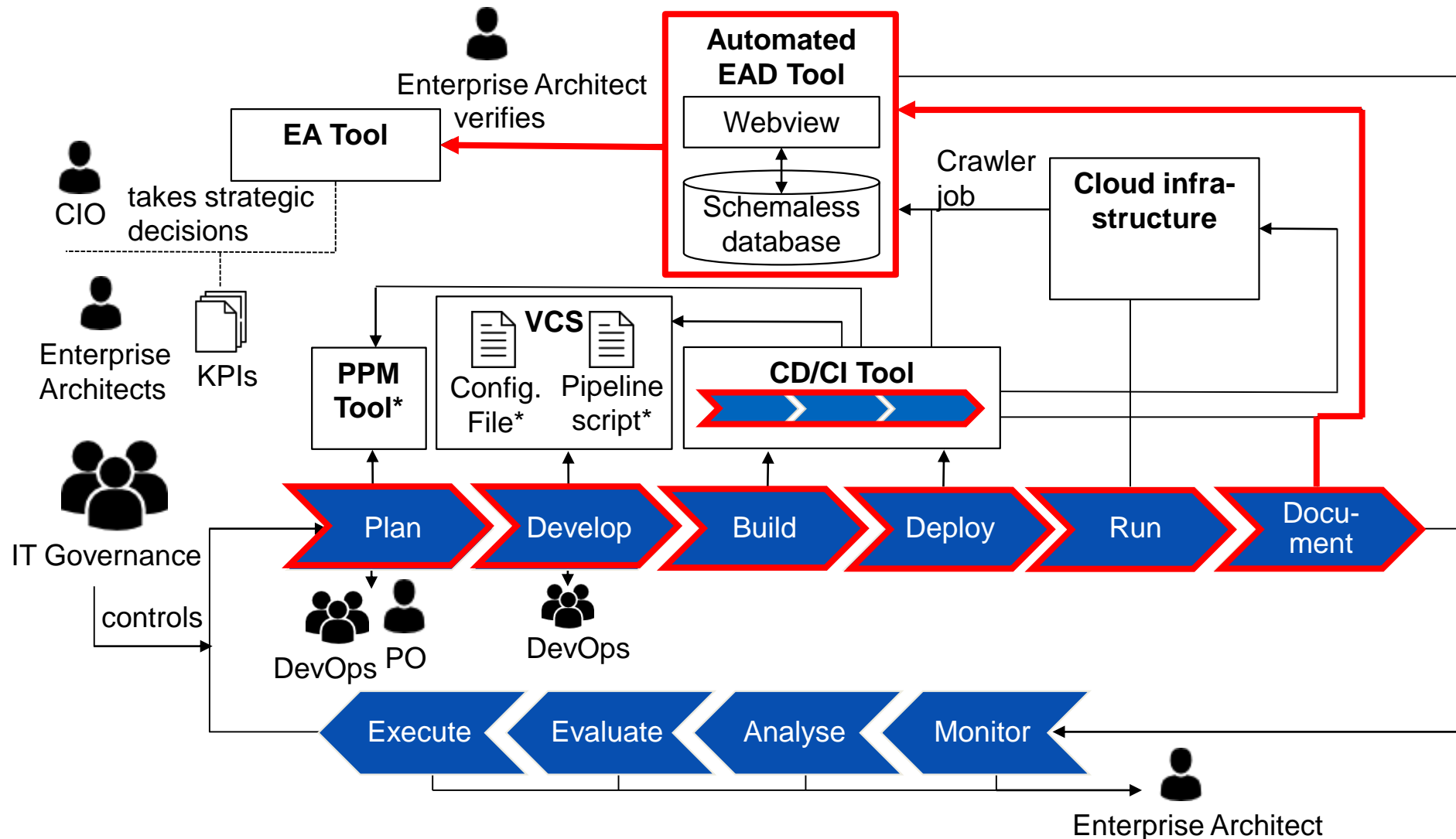
Visualizations View (2/3)



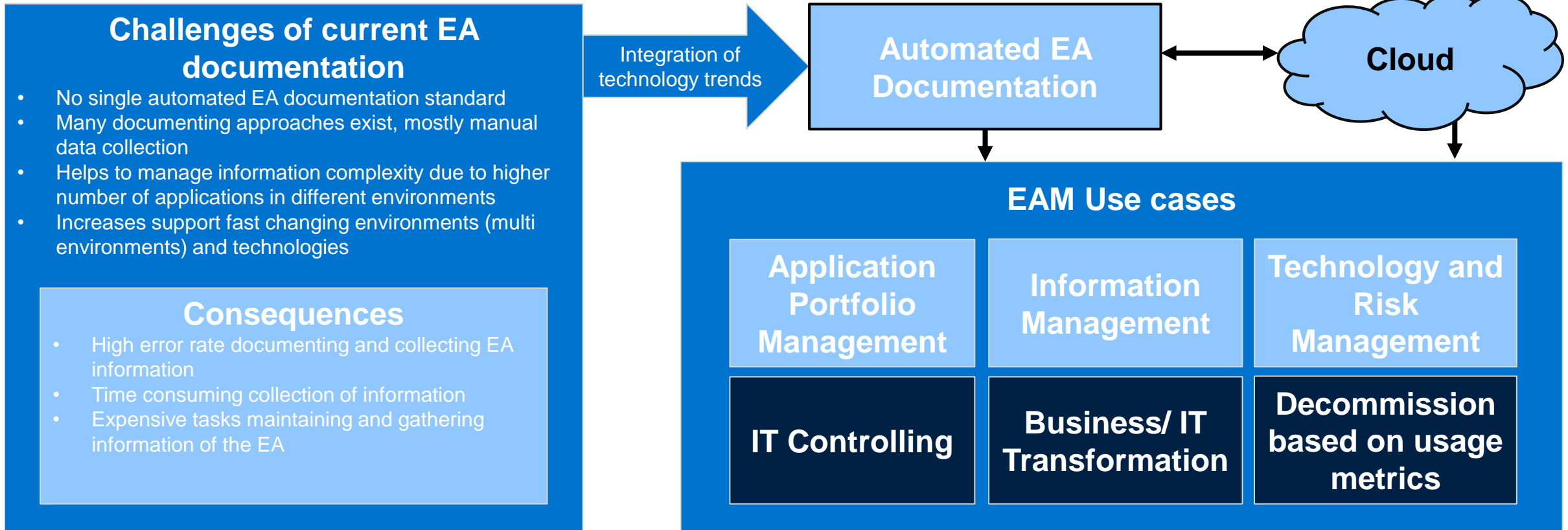
Monitoring KPIs

- Cloud clustering as additional visualization
- Backlog-items KPI: How many items are missing. Application portfolio purposes
- Status of applications: running, stopped or crashed.
- Number of deployments per time unit: Which applications change frequently?
- Traffic KPI: Decommission purposes
- Traffic heatmaps: Which applications are important. Relevant for planification and costs. Ratio costs maintenance and costs
- LOC: Maintenance vs Complexity (related to maintenance costs)
- Additional KPIs

3.1 Solution (1/2) - Concept



1. Derivation for EAM



Improved EAM use cases can be derived from an automated EA documentation process

4. Evaluation (4/4) - Prototype

Challenges:

- Web-browser does not support ECMAScript 6
- Database component was not supported (MongoDB and ElasticSearch6)
- Server component adaptation for MySQL database component
- No fully EA cloud discovery possible due to access rights

Not expected results:

- Changes were not considered relevant
- Software dependencies section: Management of software frameworks

ID	Requirement
AR1	Business Impact Analysis of applications
AR2	Data privacy compliance (GDPR compliance)
AR3	Automated verification of the 12 factor app
AR4	Automated verification of a resilience pattern
AR5	Additional KPIs: TCO, MTBF, MMTR, MTTF and real time data of users on the individual applications

Prototype can be extended by many requirements for further automated verifications

2.5 Derived EAD requirements and challenges from the literature review

ID	Requirement	Sources
RL1	Integration of different information sources	[1],[2],[3],[4],[5][6],[7],[8],[9],[10],[11]
RL2	Dynamic metamodel: Adaptable metamodel of EA tools	[1],[2],[3],[4],[5][6],[7],[8],[11]
RL3	Business added value: Several stake	[1], [3], [4]
RL4	EA Tool support: EA Tool needs an public API for RL1	[1], [3], [6], [8]
RL5	Integration of cloud environments (PaaS and SaaS)	[4], [8], [11]
RL6	Integration of runtime KPIs and monitoring information e.g.Prometheus	[4], [8], [11]

ID	Data challenge	Sources
DC1	Data granularity: Data too granular for EA	[1], [2], [3], [6], [7]
DC2	Data actuality: Data is not maintained, therefore no uptodate	[1], [2], [3], [4], [6], [7], [9], [10], [11]
DC3	Data completeness: EA information is scarce	[1], [2], [3], [6], [7]
DC4	Data correctness: Error pruned data due to manual gathering	[1], [2], [3], [6], [7], [9], [10]

Future automated EAD solutions should cover the above mentioned requirements and challenges