



Architecture Audit

For Small and Medium Enterprises

Get a broad but detailed view in your software

**BACK
ENDHANCE**

Your Auditor

Never give a sword to a man
who can't dance.

~ Confucius





Marcus Held

Independent Software Architect

Ex-Head of Development, Keynote Speaker, Author

+ Experience

Marcus' 13 years of experience in software development and his various roles in the development process give him a comprehensive understanding of company software. As the Head of Development at a Vodafone subsidiary with 30 employees, he is ideally suited to assess the state of the development department. He shares his knowledge as a keynote speaker and author at conferences and in trade magazines.

**220
Million**

User

used the software that Marcus wrote

10

Industries

looked after in recent years

15

Years

of professional experience

8

Roles experienced,

including Head of Development, Lead Developer, SRE Lead, PO and Scrum Master



The goal for [the] one-day workshop was to analyse the current situation to lay the foundation to make decisions on towards which direction we should develop our tech stack and development team. Before we started, my main concern was that we wouldn't be able in just one day to produce sufficient results, thus ending up without a valid roadmap. But that wasn't the case at all. Instead, we had a very effective workshop, which was also a lot of fun and exactly met our expectations. [...]

I am very satisfied with the result and would recommend Marcus to anyone who needs conceptual guidance in structuring their tech stack and/or technology team.

Arne Paul Oltmann

Co-Founder & CEO @ limbiq system GmbH



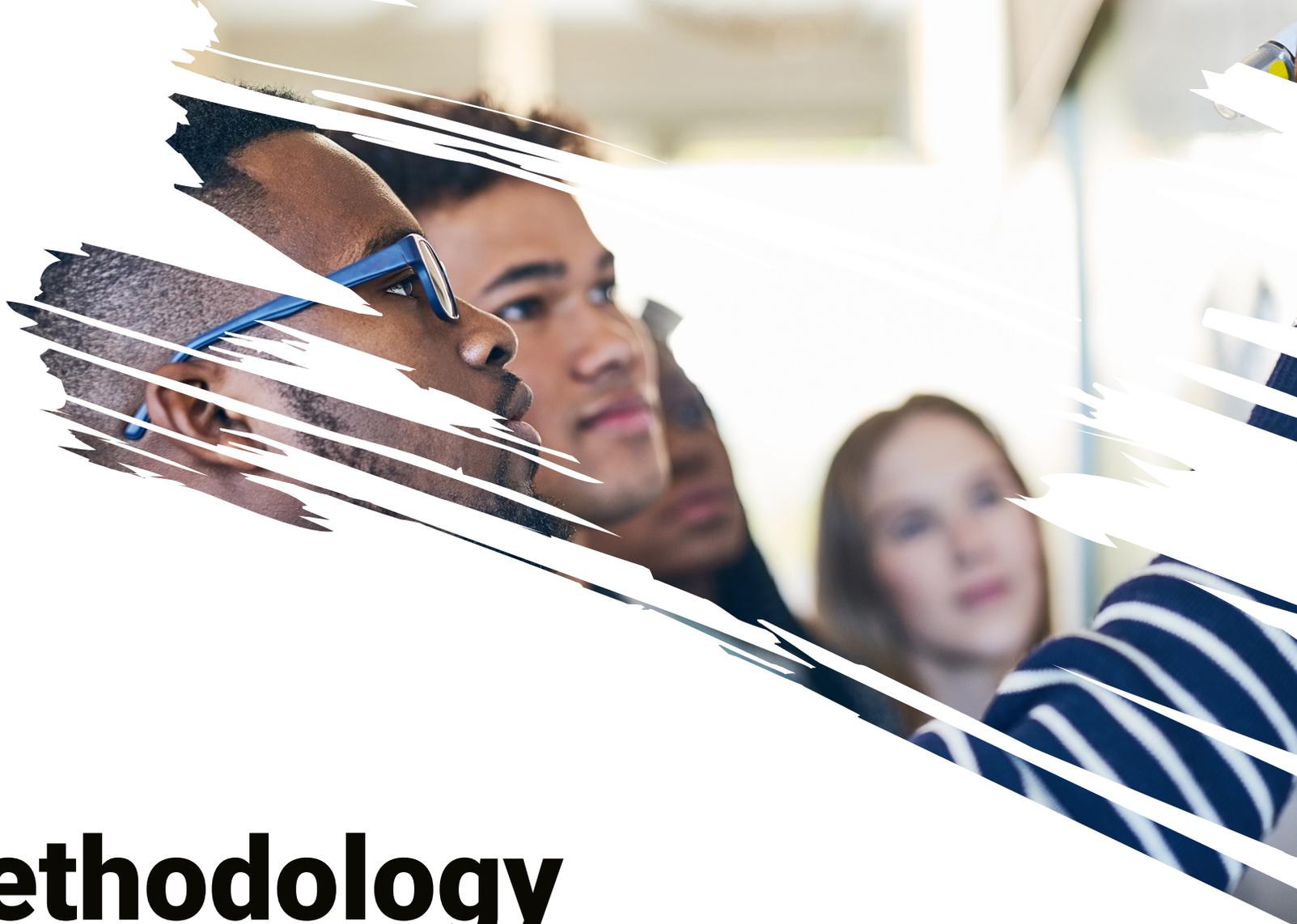
My teams and I had the special challenge of renewing our aging B2B platform technology. [...] His consultation took the form of a three-day workshop with our entire development team and led to highly successful results.

Marcus proved to be extremely competent and professional. [...]

In conclusion, I would like to emphasize that working with Marcus was a very positive experience. His ability to understand complex technical challenges and find pragmatic solutions has left a lasting impression on us. [...]

Andreas Rasic

CTO / CIO @ S&S Internet Systeme GmbH



**BACK
ENDHANCE**

Methodology

If you can't explain it simply, you
don't understand it well enough.

~ Albert Einstein



Timeline

1.

Business Constraints

All software must be designed with the business model and organizational constraints in mind. These findings form **the basis for the following formats.**

2.

Technology Map

Which technology is used to solve which problem in the software.

3.

Architecture Workshop

Based on arc42, we draw the static and dynamic blocks of the architecture.

Final Report

4.



Interviews

Individual interviews with all relevant key personnel, from department heads to tech leads and product owners, provide an insight into the processes and strategies of the development department.

5.



Code Review

The implementation of the solutions is checked for maintainability and security.

1. Business Constraints

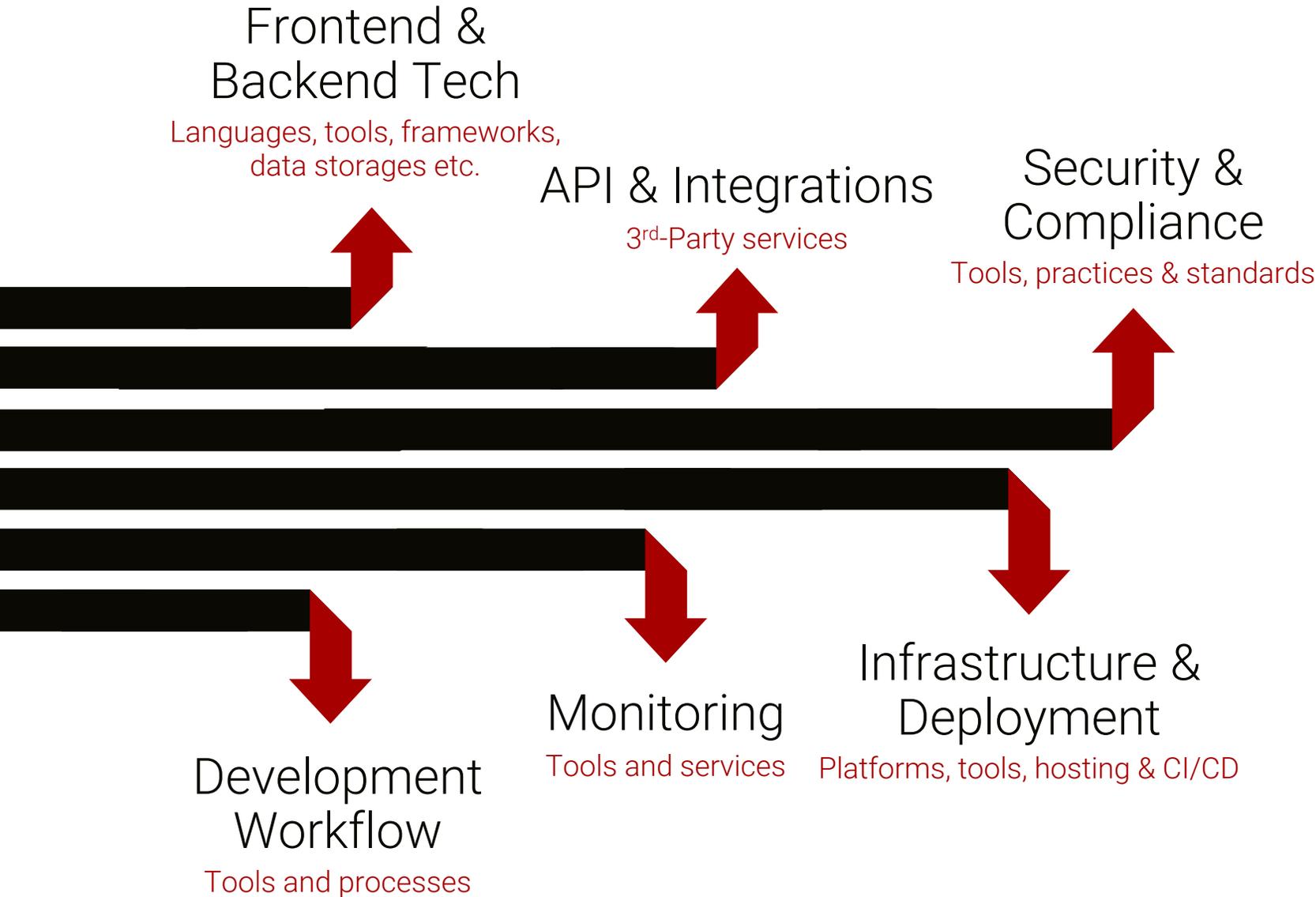
The architecture and technology decisions of a software are shaped by the framework of the business and the organisation.

In this session, this context is analysed with the relevant business owners (e.g. CEO, Head of Sales, PO, etc.).

The most important requirements for the quality of the software are identified based on **ISO 25010**.

Based on these findings, the focus is placed on the following formats.

2. Technology Map



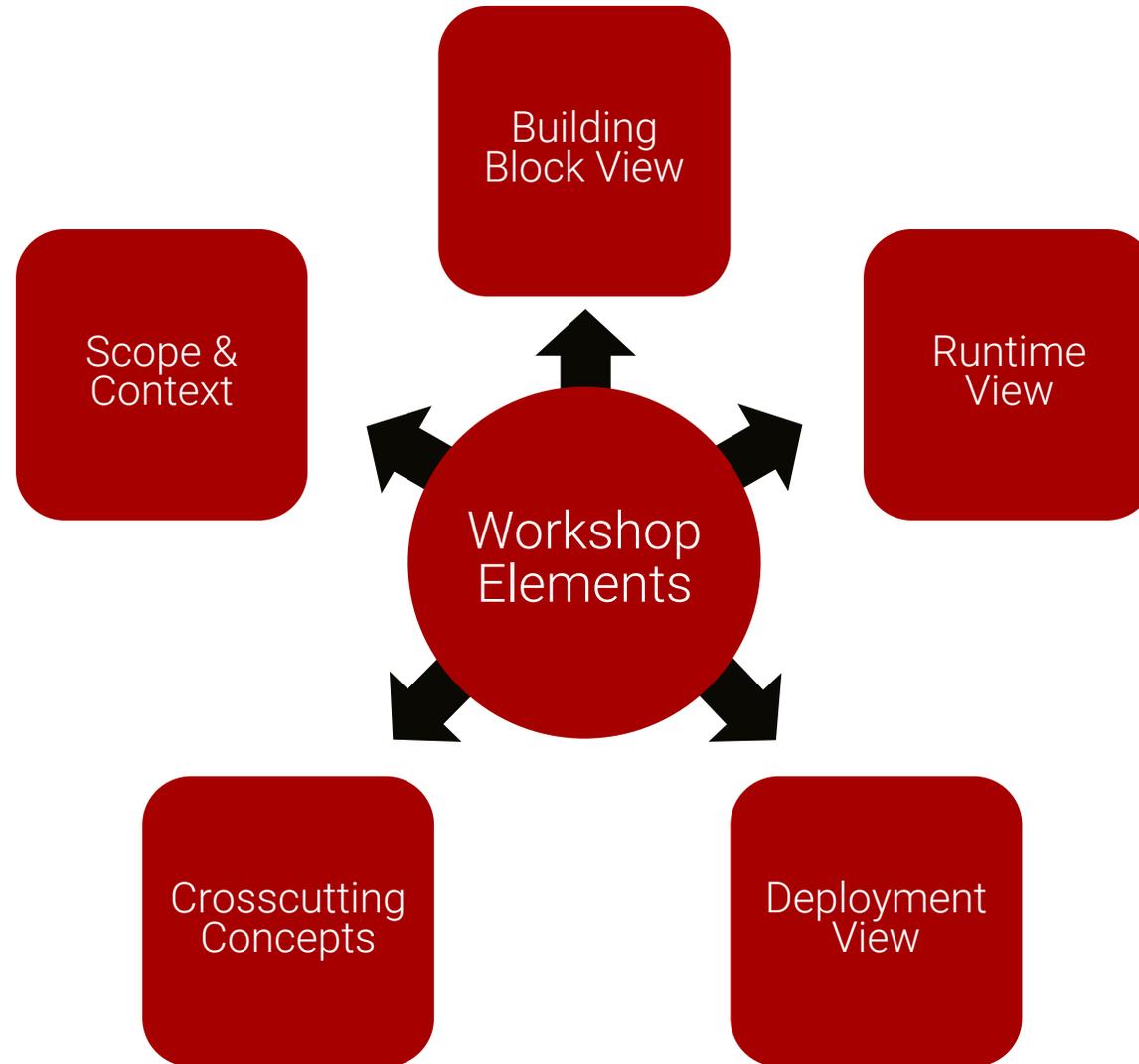
The **Tech Stack Canvas** is used to identify all the technologies used. These findings are essential for assessing the performance, security and scaling requirements

3. Architecture Workshop

To understand the architecture of a software, the delimitations of the modules, their interactions and the integrations must be considered from different perspectives.

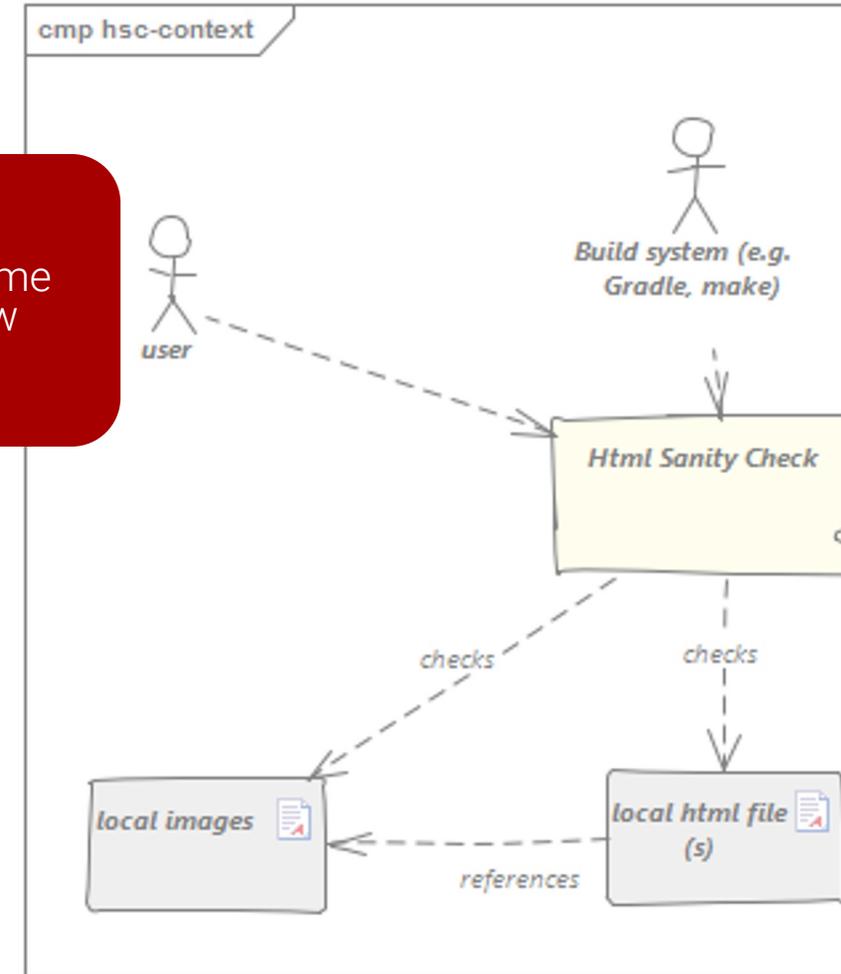
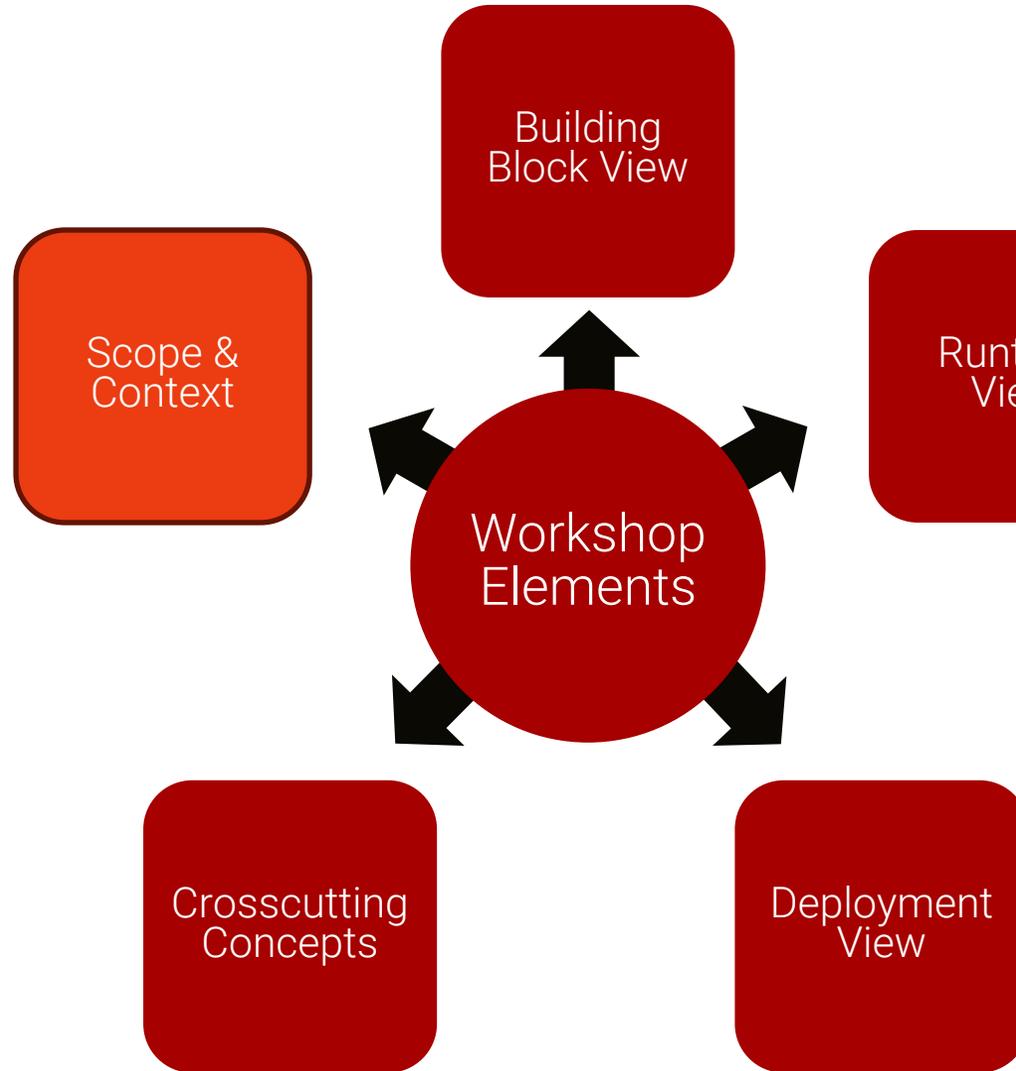
Based on the **arc42** documentation format, the static and dynamic elements of the software are modelled with the development team.





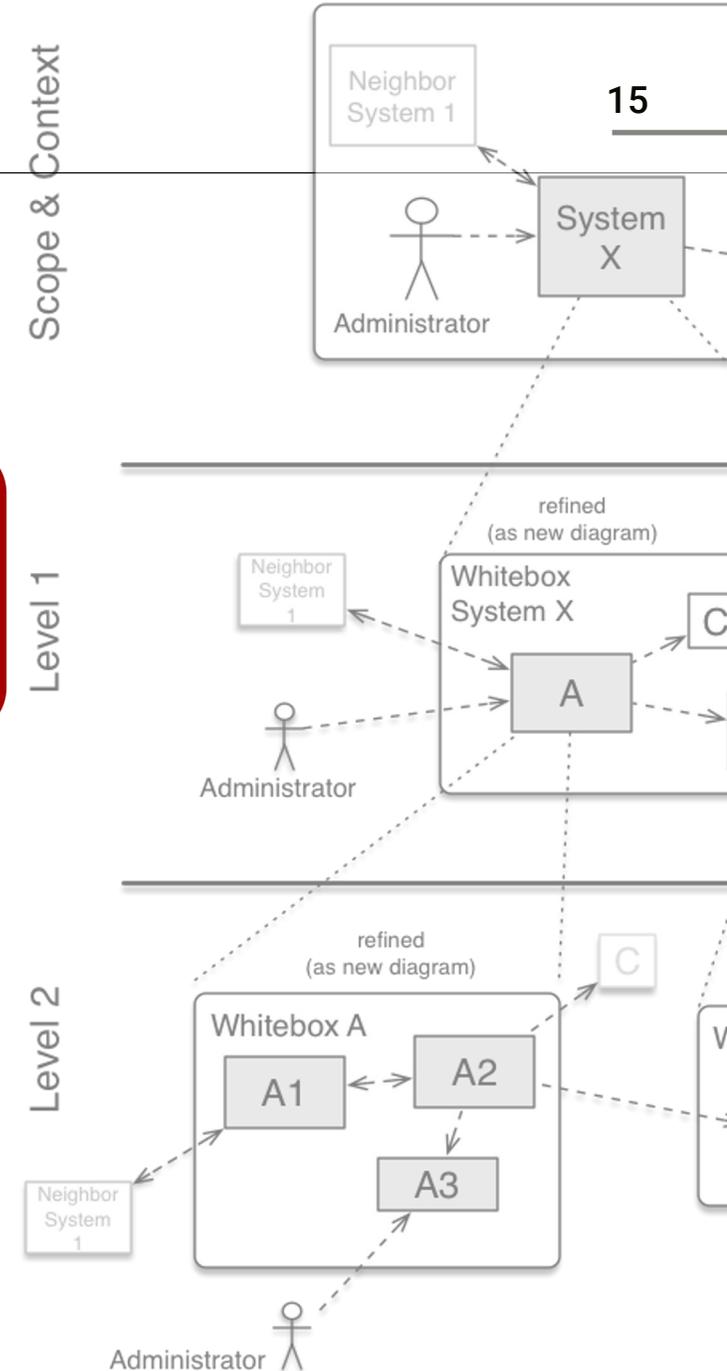
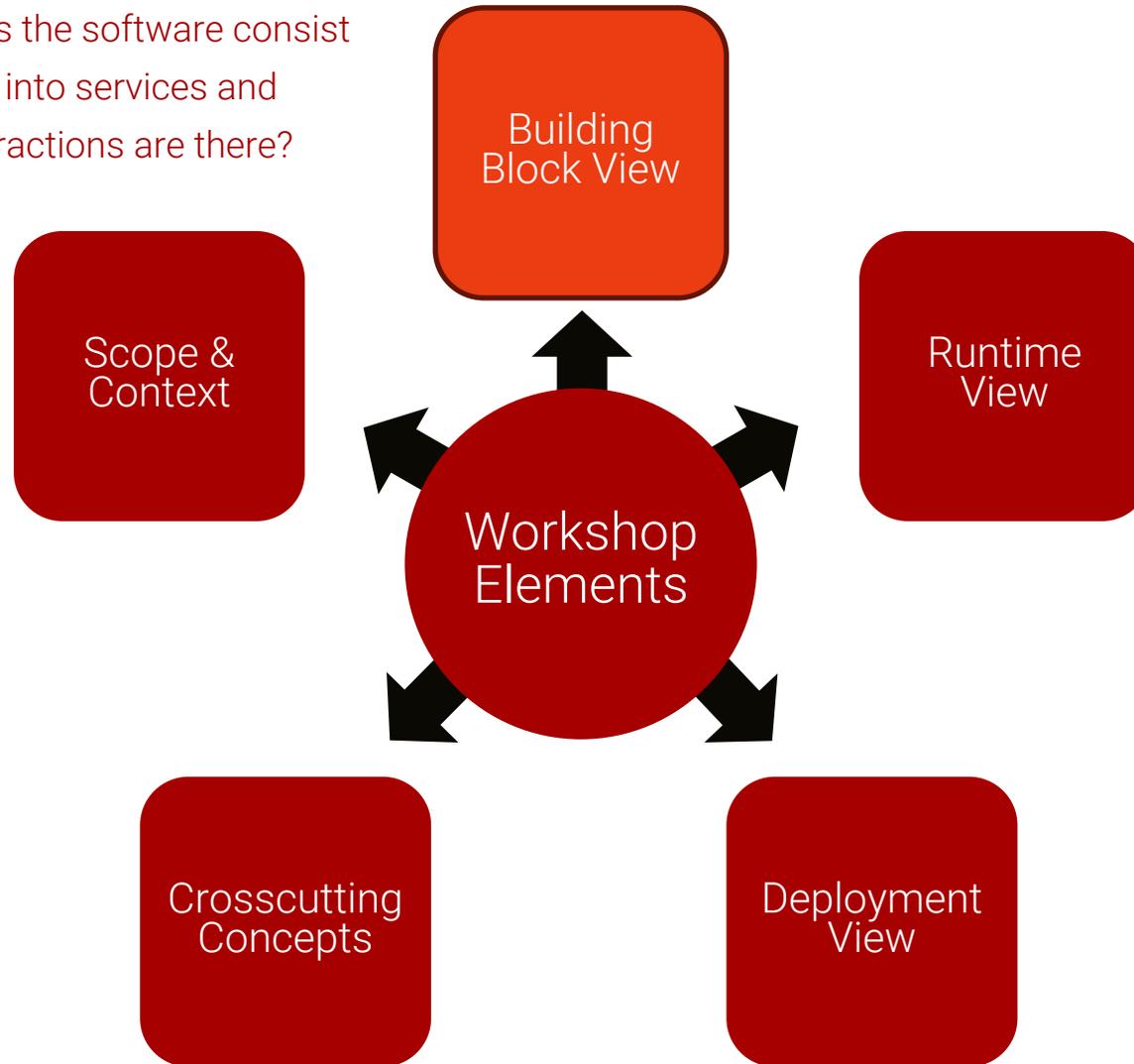
Workshop Elements

Draw the big picture. Which actors are there? How do they interact with the software? Which systems does the software integrate with?



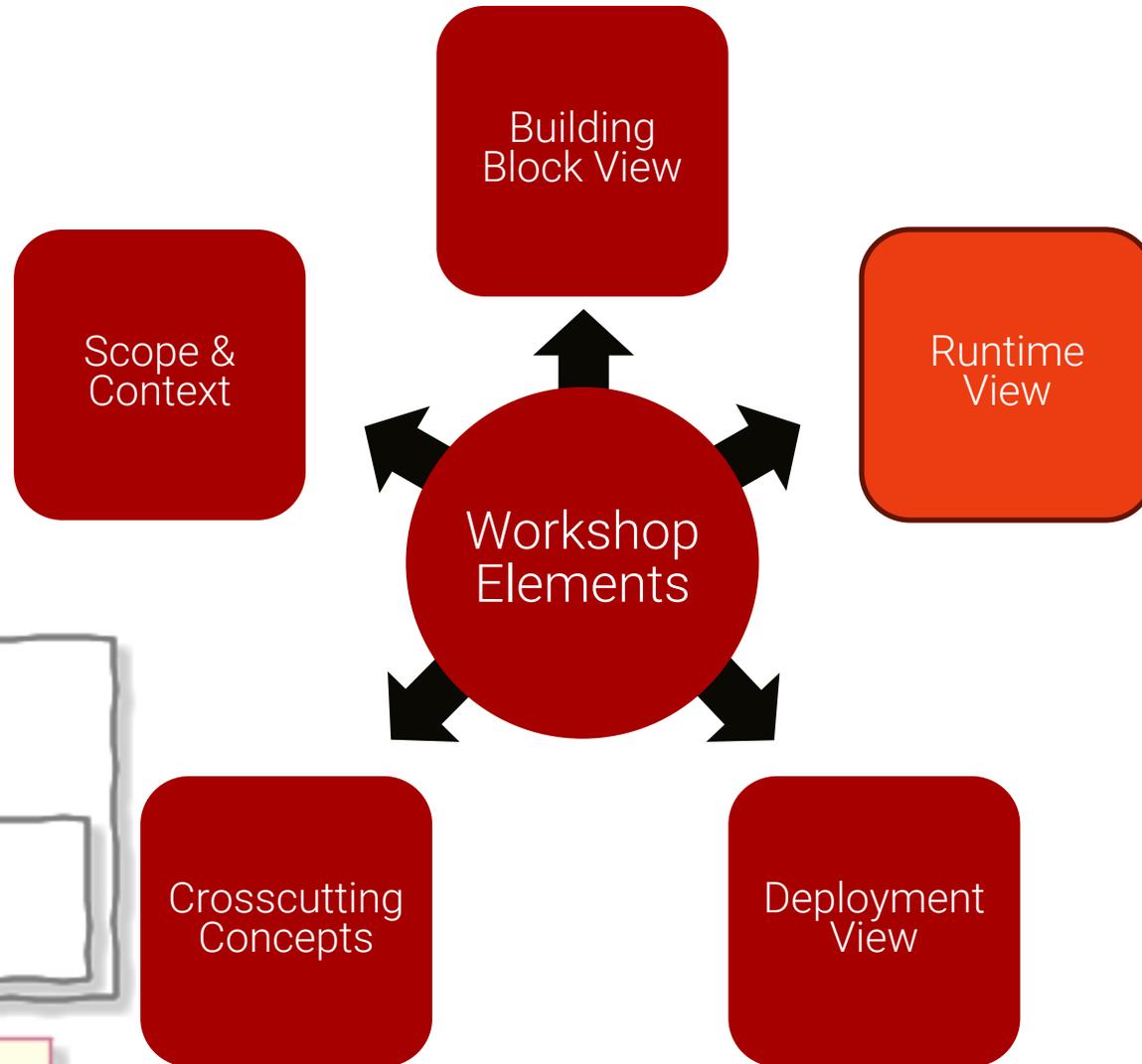
Workshop Elements

Which domains does the software consist of? How is it divided into services and modules? What interactions are there?

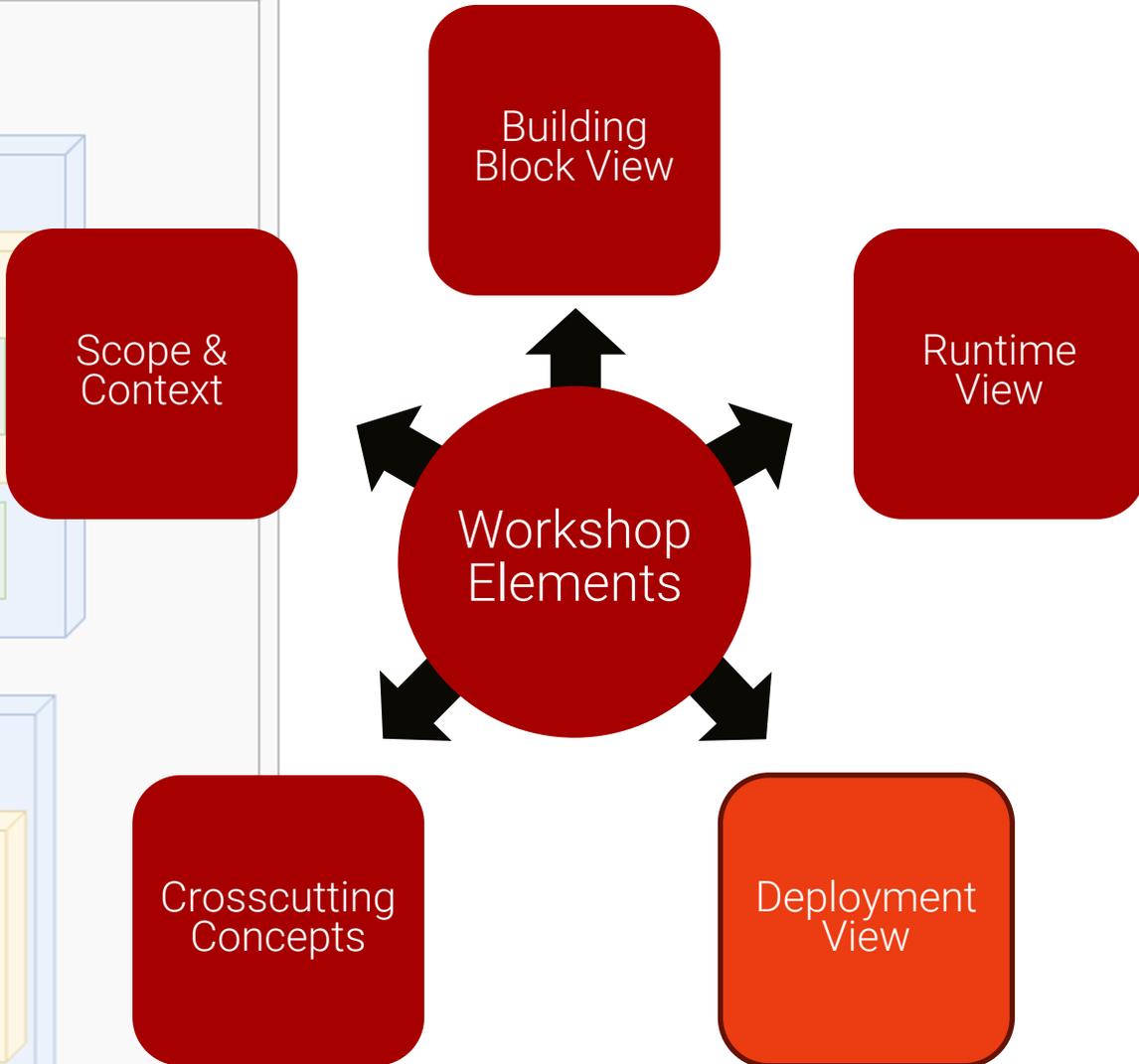
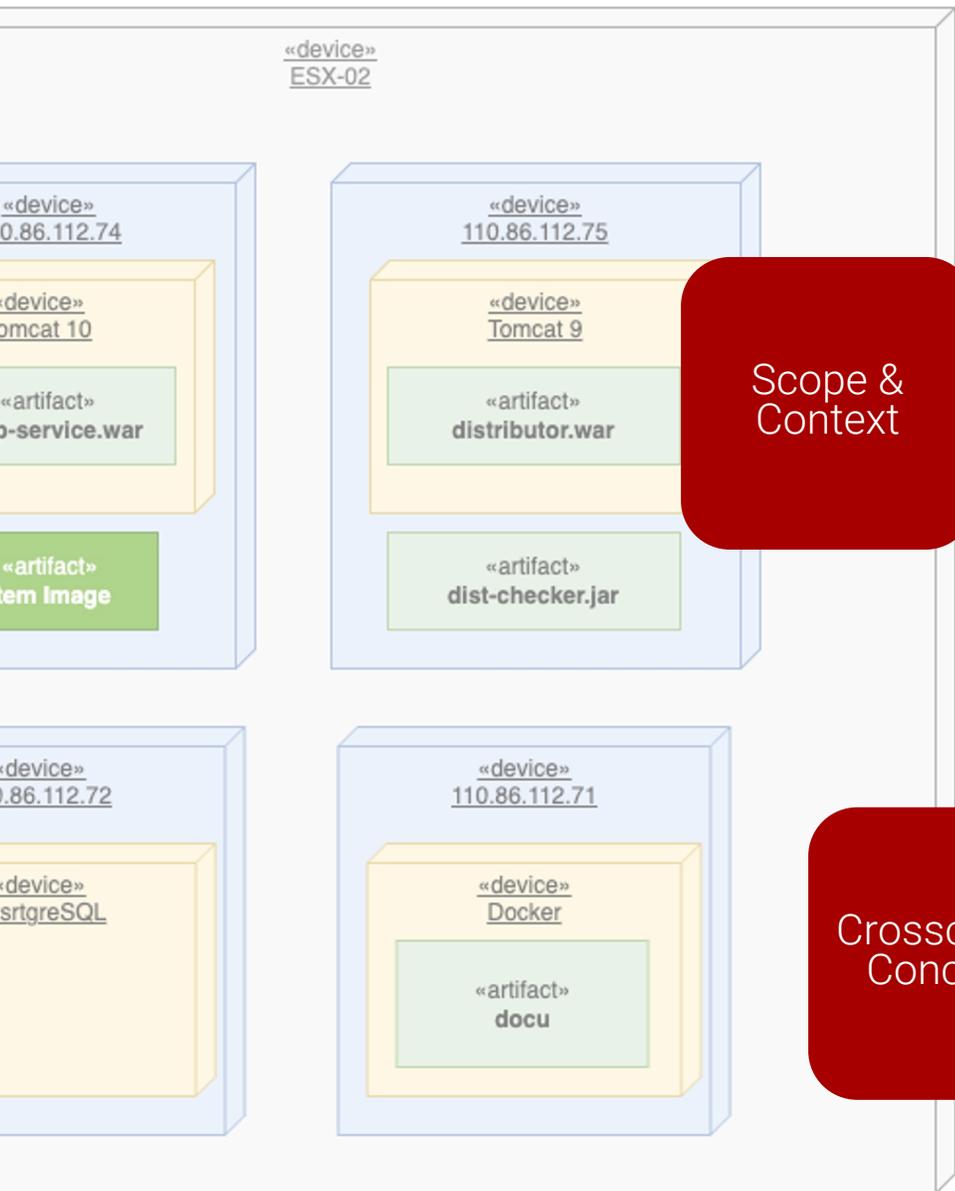


Workshop Elements

```
«abstract»  
Checker  
  
...tor.getConfiguredHtmlFiles()  
...s =  
...actory.getAllCheckersByAnnotation  
  
...AllCheckers(file)  
...rs]  
...ingleCheck(file)  
  
«abstract»  
Checker
```

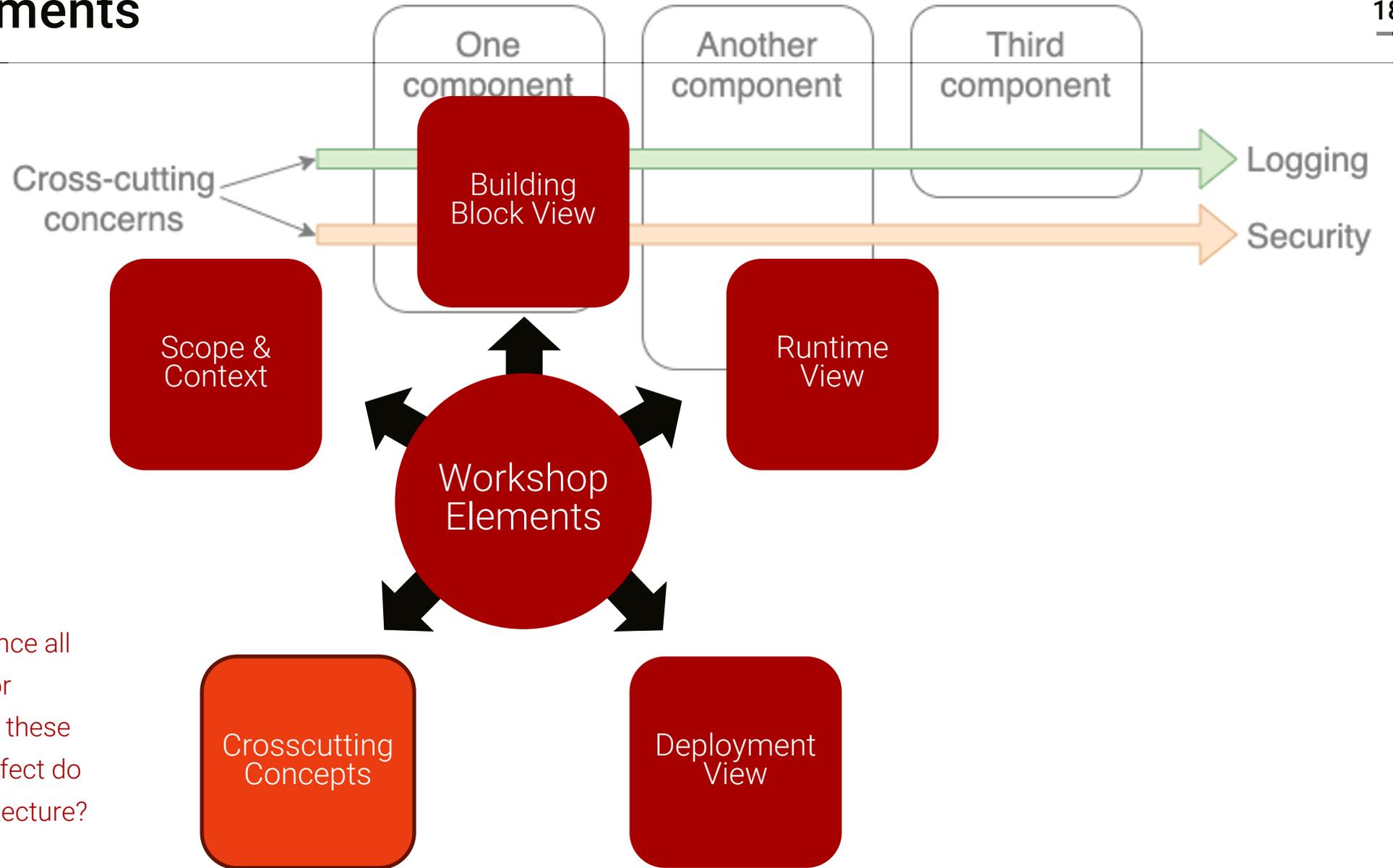


What is the runtime behaviour of the application? How do the most important algorithms work? How is the thread and memory management?



What infrastructure is used?
Where is which application deployed?
Where is the hardware located?
What backup strategy is in place?

Workshop Elements



Which concepts influence all parts of the software or organisation? How are these implemented? What effect do they have on the architecture?

+ 4. Interviews

Interviews are conducted with all key personnel to assess the maturity of the company.
For example, with:

1. ● CTO & Head of Development
2. ● Lead Developer
3. ● Teamlead
4. ● Scrum Master / Project Manager
5. ● Product Owner
6. ● Lead DevOps / Administrator
7. ● Support Lead
8. ● Quality Assurance Lead



CTO & Head of Development

- Team topologie
- Tech vision & strategy
- Organisational structure
- Salary structure
- Employee development strategy



Lead Developer

- Development process
- Migration & versioning strategy
- Security concepts
- Observability
- Automation



Teamlead

- Personal Development Goals
- Onboarding Process
- Hiring Process
- Training Strategy



Scrum Master / Projectmanager

- Meeting structure
- Processes
- Tooling



Product Owner

Roadmap

Product strategy

Requirements

Expectations



Lead DevOps / Administrator

Resilience

SLA & On-Call

Monitoring

Provisioning

Automations

Security



Support Lead

Automation

Tooling

Support Levels



Quality Assurance Lead

Automation

Processes

Release authorisation

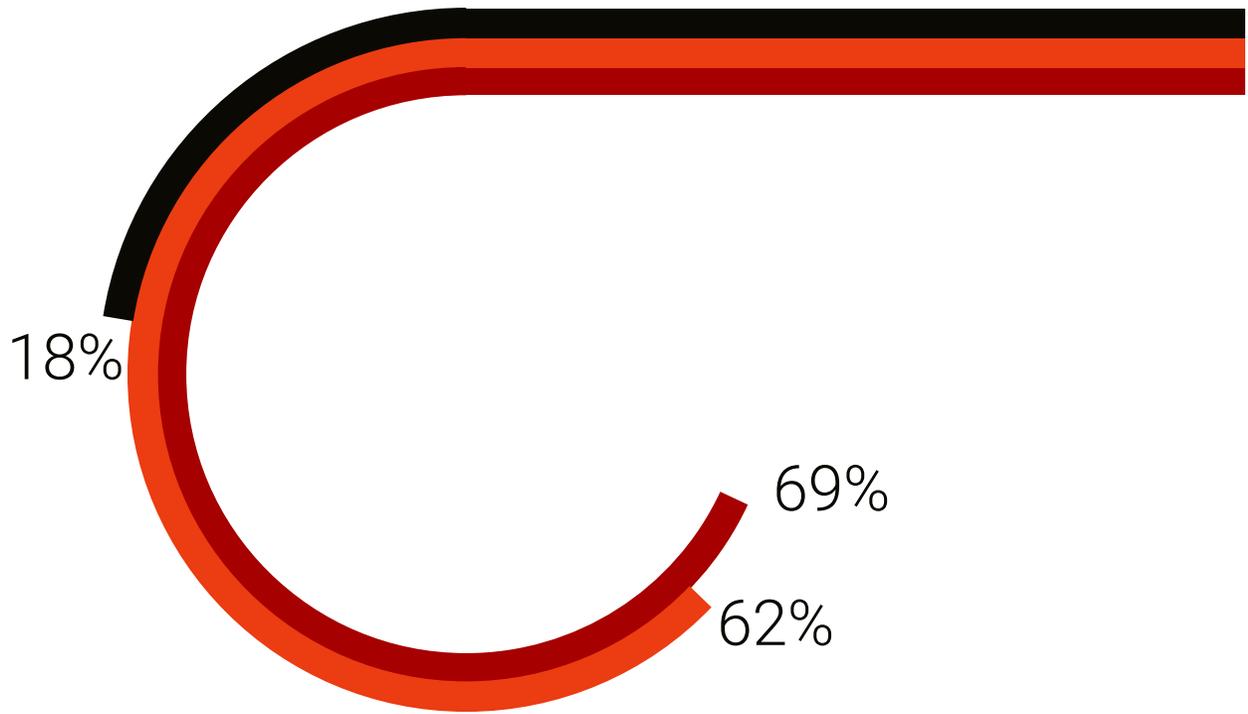
Feature testing



5. Code Review

Once a complete understanding of the concepts of the software and its architecture has been gained, the software is tested in detail. **Special focus is placed on the implementation of the central concepts and quality features.**

Among other things, the following aspects are analysed: Module distribution, build tool, CI / CD pipeline, database schema, dependencies, code quality, documentation, API maturity, authentication and authorisation, profiling, standardisation levels, ...



Valuation

All analysed aspects are summarised and weighted in a final report. A **risk assessment and a recommendation** for action are issued for each defect.

The report is divided into organisational, personnel, infrastructure, architecture and implementation aspects.

Contact Us

contact@backenhance.com

<https://backenhance.com>

+49 151 / 449 351 67

