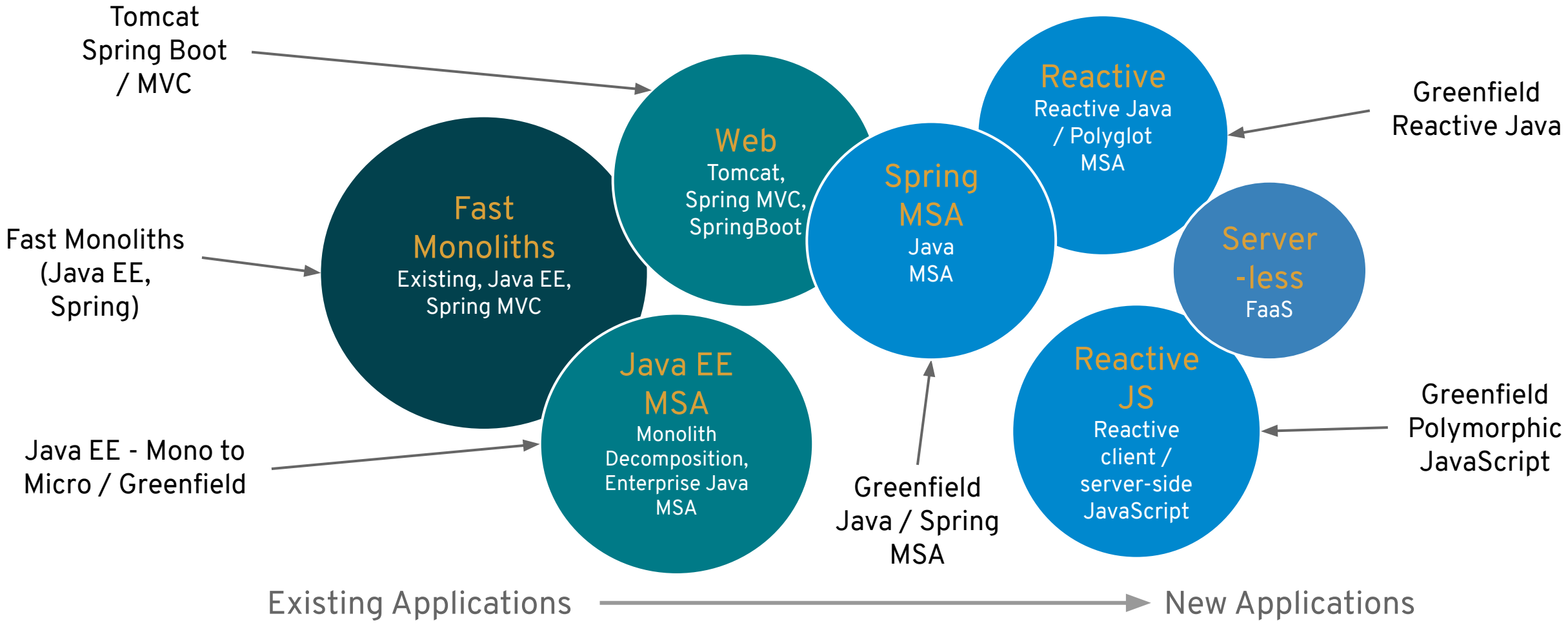
Migrating Your Middleware Applications to a Modern Platform

OpenShift Container Platform

Scott Seighman
Solutions Architect
sseighma@redhat.com

The Spectrum of Enterprise Applications



Migration & Modernization Approaches

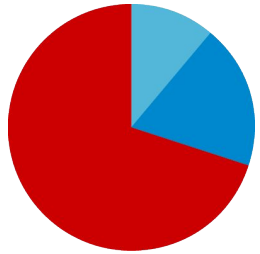
Modernizing Existing Apps

- Reuse existing functionality and data as much as possible
- Move existing workloads to a modern deployment platform
- Apply new processes, products, and technology to existing apps

Developing New Applications

- API-centric polyglot microservices architecture
- Autonomous development teams
- Agile development, continuous deployment, DevOps culture
- Containerized & orchestrated cloud deployments

Application Modernization



Existing Apps

How much work required to rewrite?

Review
Analyze
Prioritize

- Lift & Shift
- Connect & Extend
- Re-write
- Repurchase
- Retire
- Retain as is

Smaller or frozen apps are candidates here

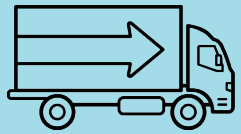
Highly scaled and high rate of change apps are candidates

Not a target

Patterns in Modernizing Workloads

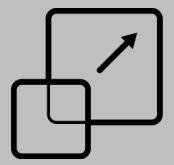
LIFT & SHIFT

- **Containerize** existing workloads
- Deploy them on **PaaS/CaaS**
- Keep external integrations and data on legacy
- Legacy applications have to be well written and suited



CONNECT & EXTEND

- **Legacy remains intact**
- New layer - new capabilities
- Deploy on **PaaS/CaaS**
- **New integration points** between legacy and new layers **(Need for Agile Integration)**



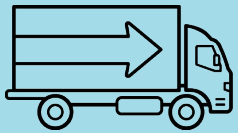
RE-WRITE

- **Legacy is totally replaced**
- New interfaces and data
- Use **PaaS/CaaS** to run
- Some data and features can be re-wrapped, but mostly are retired.

Patterns in Modernizing Workloads

LIFT & SHIFT

- **Containerize** existing workloads
- Deploy them on a **PaaS**
- Keep external integrations and data on legacy
- Legacy applications have to be well written and suited



FOCUS FOR THIS SECTION

CONNECT & EXTEND

- **Legacy remains intact**
- New layer - new capabilities
- Deploy on **PaaS**
- **New integration points** between legacy and new layers **(Need for Agile Integration)**

RE-WRITE

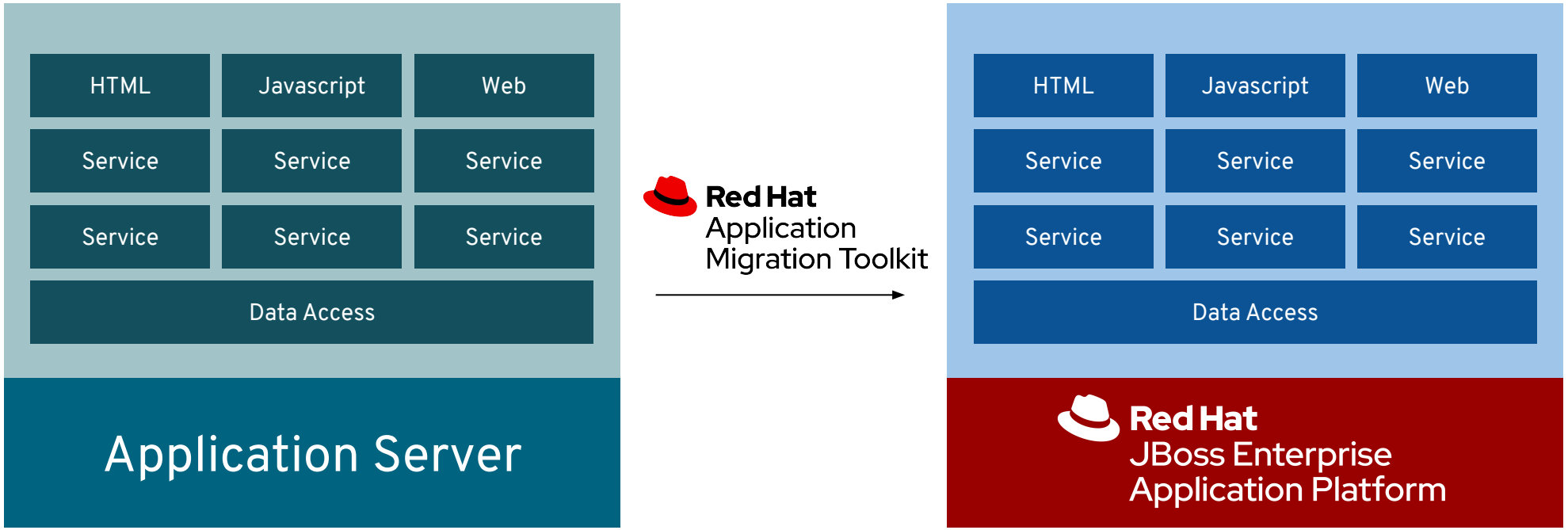
- **Legacy is totally replaced**
- New interfaces and data
- Use **PaaS** to run
- Some data and features can be re-wrapped, but mostly are retired.

Red Hat

Questions to Consider …

- What is your overall business objective for app modernization?
  - Biggest perceived risks?
- How much are you spending on app maintenance?
  - Lack of automation/IT standardization often the culprit
- How long does it take to get changes into production?
  - And what is your success/fail ratio?
- Current skill set?
  - Do you need training on newer technology before modernization?
  - Consider future availability of skills in the workforce
- Regulatory/compliance requirements?
  - Regional CCSP workloads
  - Data Sovereignty

Lift-and-Shift Monolith to the Cloud



Red Hat

Red Hat

Red Hat Application Migration Toolkit

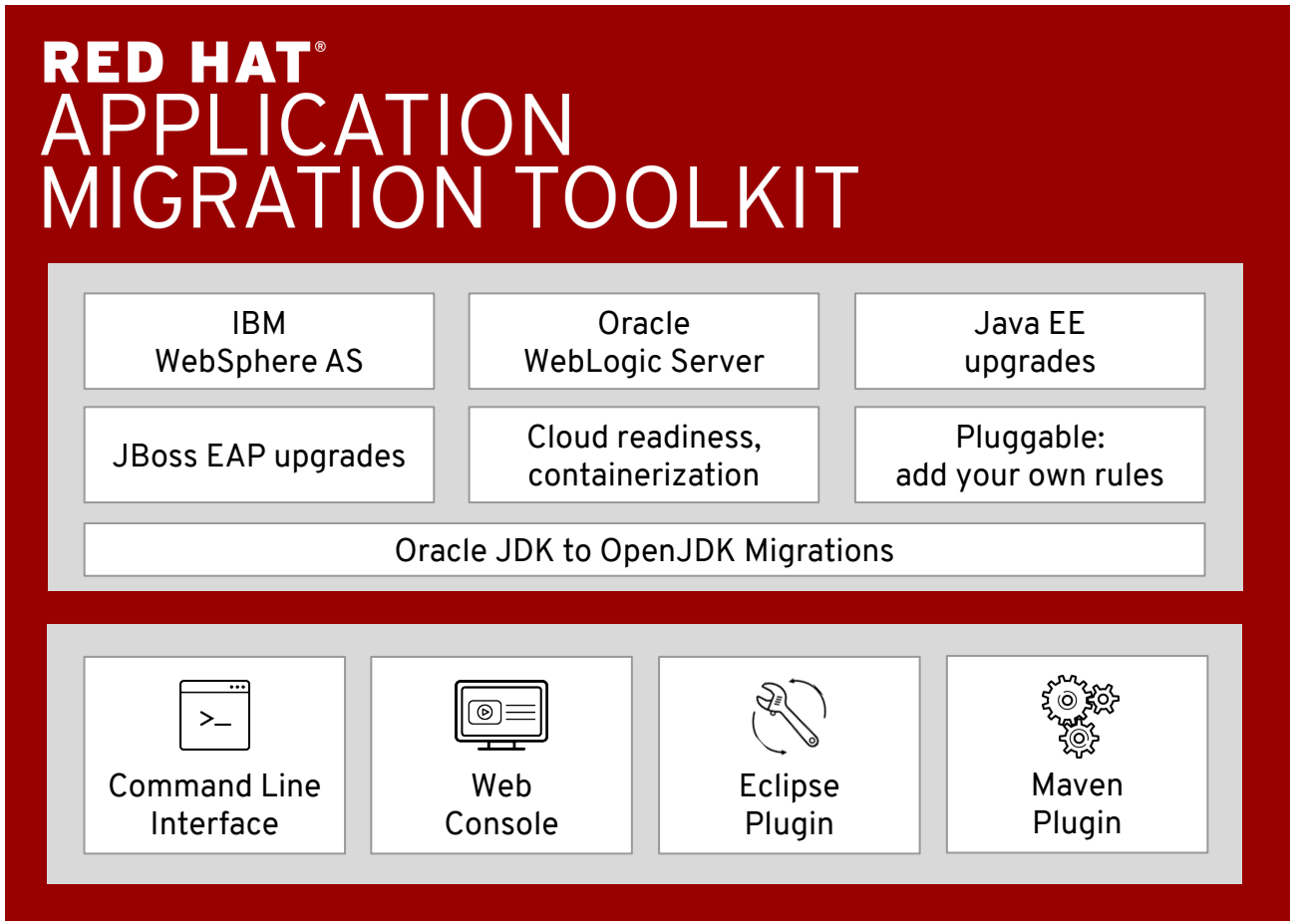
- Red Hat Application Migration Toolkit (RHAMT) is an assembly of open source tools that enables large-scale application migrations and modernizations
- The tooling consists of multiple individual components that provide support for each phase of a migration process
- The migrations supported include application platform upgrades, migrations to a cloud-native deployment environment, and also migrations from several commercial products to the Red Hat JBoss Enterprise Application Platform

Red Hat Application Migration Toolkit

- RHAMT is a rule-based migration tool that analyzes the APIs, technologies, and architectures used by the applications you plan to migrate

    1. Uses rules to extract files from archives
    2. Decompile files
    3. Scans and classifies file types
    4. Analyzes XML and other file content
    5. Analyzes the application code
    6. Builds the reports

Red Hat Application Migration Toolkit



Catalyze large scale Java application modernizations and migrations

- Automate analysis
- Support effort estimation
- Accelerate code migration
- Bring workloads to OpenShift
- Free & Open Source
- Assembly of tools for each phase of a transformation process

Homepage - Documentation

Red Hat

Red Hat Application Migration Toolkit

- RHAMT provides a number of different distributions to meet your needs, and all include detailed reports that highlight migration issues with effort estimation
  - CLI
  - Web Console
  - Eclipse Plugin

RHAMT CLI Example

```
$ bin/rhamt-cli --sourceMode --input /path/to/source_folder/
--output /path/to/output_folder/ --target eap:7
```

The options are straightforward:

- **`–sourceMode`** indicates the input files are source files instead of compiled binaries
- **`–input`** path to the file or directory containing the files to be analyzed
- **`–output`** path to the directory to contain the reports
- **`–target`** technology to migrate to; used to determine the rules for the analysis

Once the analysis finishes, a message will be seen in the console indicating the path to the report.

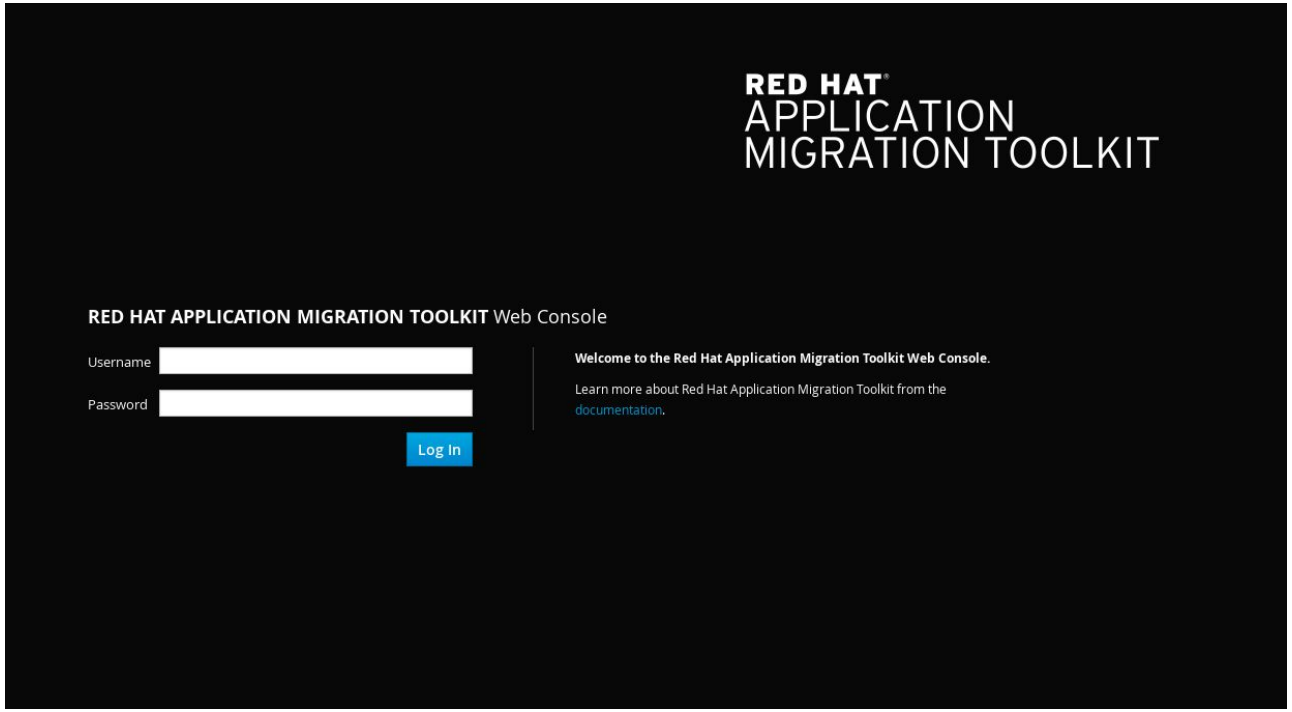
```
Report created: /path/to/output_folder/index.html
Access it at this URL: file:///path/to/output_folder/index.html
```

RHAMT Web Console

Start the Web Console

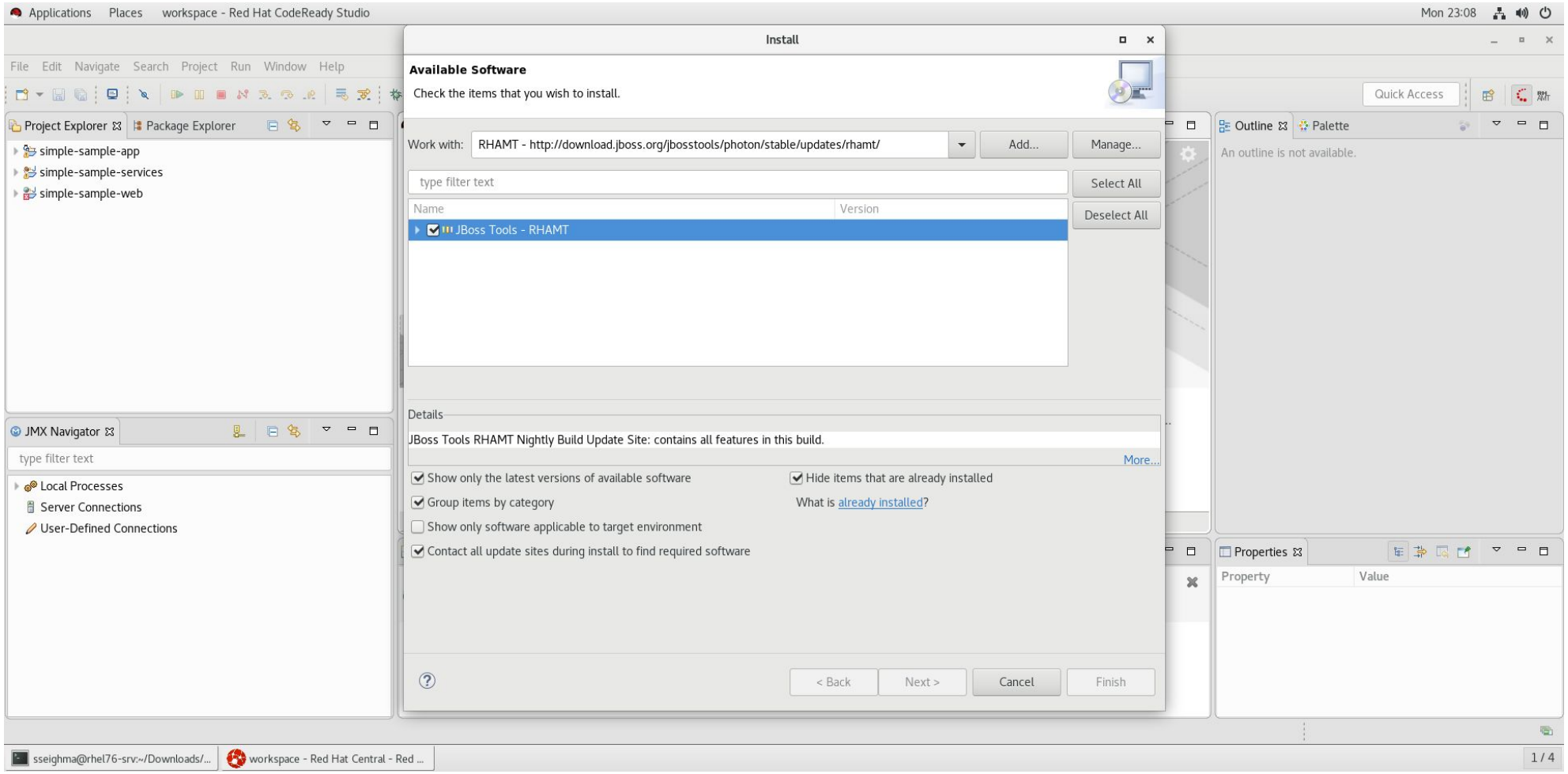
```
$ RHAMT_HOME/run_rhamt.sh
```

Open the console in a browser: http://localhost:8080/rhamt-web



Red Hat

RHAMT Eclipse Plugin

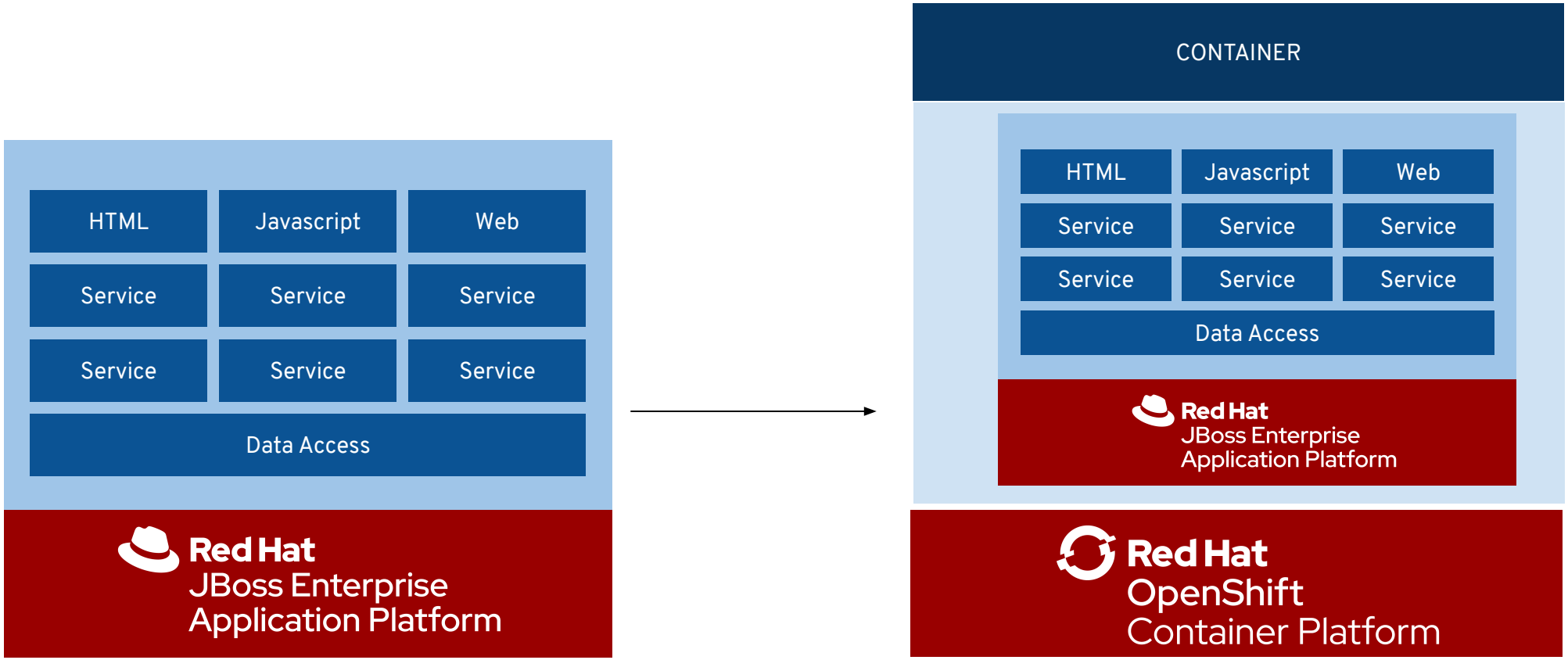


RHAMT Oracle JDK to OpenJDK Migration

- **OracleJDK to OpenJDK Migration Path**
  - This is accomplished by specifying oraclejdk as the source technology and openjdk as the target technology, resulting in a report that highlights issues when performing this migration
- Analyze an existing Java application portfolio for required changes when moving from Oracle JDK to OpenJDK
- Source code of the application is not needed, binary format is sufficient
- App Server runtime independence that works for Tomcat, WebSphere, WebLogic, or Spring applications

RHAMT Demo

Lift-and-Shift Monolith to the Cloud



Red Hat

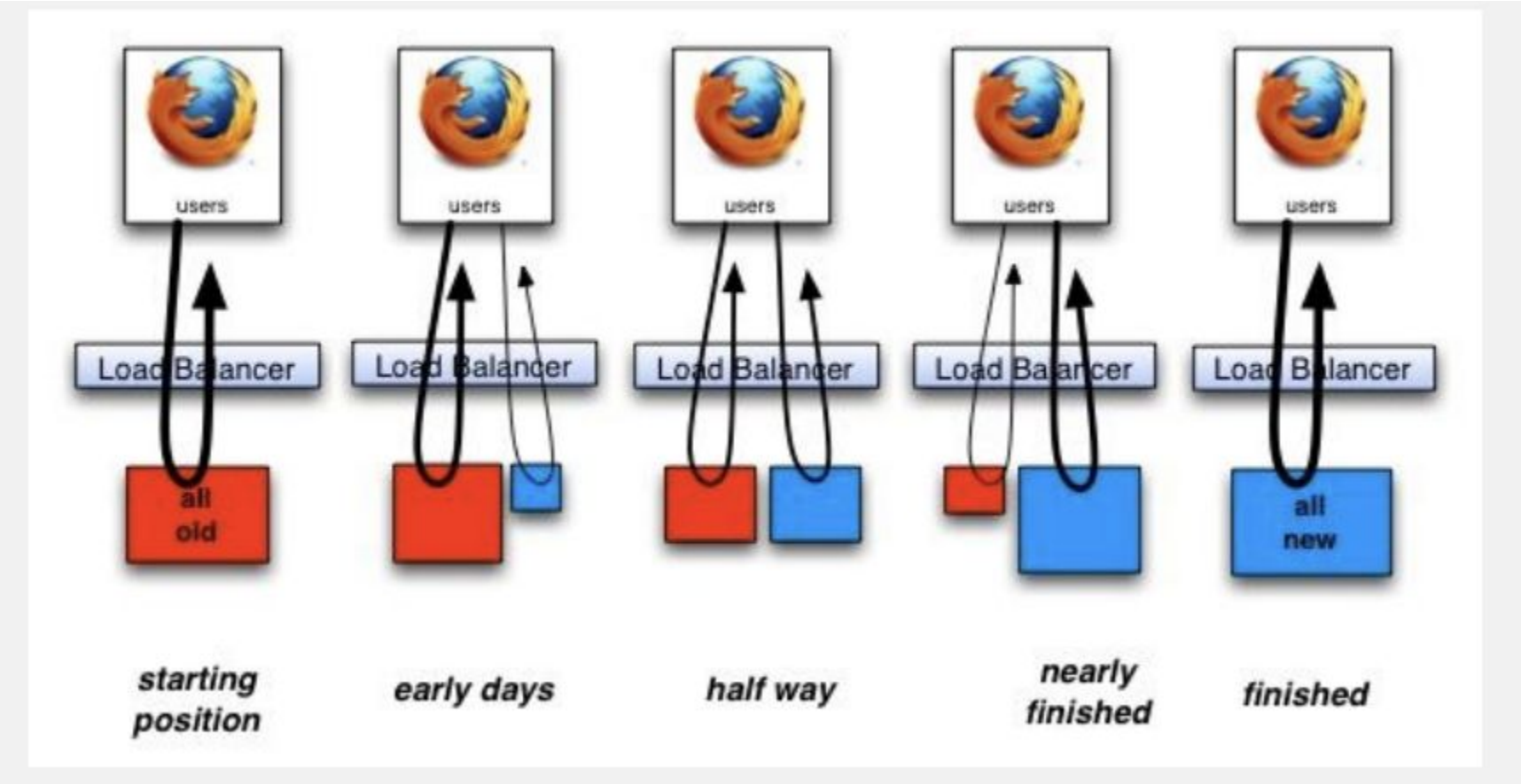
Majestic Monolith

https://m.signalvnoise.com/the-majestic-monolith-29166d022228

Majestic (Fast-Moving) Monolith

- Large organizations have a tremendous amount of resources **invested in existing** monolith applications
- Looking for a **sane way** to capture the benefits of containers and orchestration **without having to complete rewrite**
- **OpenShift** provides the platform for their existing investment with the benefit of a **path forward** for microservice based apps in the future

Strangling the Monolith



Frameworks & Runtimes for Kubernetes and OpenShift

Build new and modernize existing applications, using cloud computing models and DevOps practices to deliver applications and services quickly and frequently.

- Established and emerging runtimes, frameworks, and languages
- Leverage your developers' enterprise Java expertise with minimal to no learning curve to microservices
- Prescriptive/guided development via missions and boosters

Guided Choice of Runtimes & Languages

ENTERPRISE JAVA

Red Hat JBoss Enterprise Application Platform

SPRING APPS

JAVA MICROSERVICES

JAVASCRIPT FLEXIBILITY

REACTIVE SYSTEMS

TOMCAT SIMPLICITY

RED HAT® JBOSS® WEB SERVER

Advantages of a Modern (Container-Based) Architecture

- Be flexible
- Think bigger
- Work smarter
- Achieve (actual) standardization
- Write once, run anywhere
- Deliver exceptional application quality
- Use your favorite tools and languages

Summary

- Application development teams are evolving their process, platform and architecture to meet modern business challenges
- There are multiple technical solutions for app modernization depending on resources, regulations and risk
- Some organizations move faster than others
- Red Hat JBoss EAP and OpenShift Container Platforms + RHOAR provides a trusted solution for today's business-critical apps and a supported path to modern application architectures

Resources

- Path to Cloud Native Application
- Understanding Cloud-Native Apps
- Application Modernization
- Challenges on Integration
- Red Hat Process Automation
- IDC Business Value of OpenShift Whitepaper
- Agile Integration - A Blueprint for enterprise architecture

RHAMT Resources

- Download
  - https://developers.redhat.com/products/rhamt/download/
- Getting Started
  - https://developers.redhat.com/products/rhamt/hello-world/#fndtn-rhel
- Documentation
  - https://developers.redhat.com/products/rhamt/docs-and-apis/
- RHAMT Forums
  - https://developer.jboss.org/en/windup

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat

Why Migrate to JBoss EAP?

Runtime [1][2] (framework)	Boot time server only	Boot time including app deployment	Memory usage without load	Memory usage under load	Measured[3] throughput
JBoss EAP (Java EE)	2 - 3 sec	**3 sec**	40 MB	**200 - 400 MB**	**23K req/sec**
JBoss EAP (Spring)	2 - 3 sec	7 sec	40 MB	500 - 700 MB	9K req/sec
JBoss WS/Tomcat (Spring)	**0 - 1 sec**	8 sec	40 MB	0.5 - 1.5 GB	8K req/sec
Fat JAR (Spring Boot)	N/A	**3 sec**	**30 MB**	0.5 - 2.0 GB	11K req/sec

Don’t believe it? Try it out yourself http://bit.ly/modern-java-runtimes

[1] The microservice is a simple REST application.
[2] All runtimes are using their default settings
[3] The performance test was conducted with ApacheBench using 500K request with 50 users and keep-alive enabled.

The Umbrella

APPLICATION MODERNIZATION & MIGRATION

MODERNIZATION

- BETTER SOFTWARE ARCHITECTURE
- AGILE INTEGRATION
- STREAMLINE APPLICATION LIFECYCLE
- CONTINUOUS INNOVATION

MIGRATION

- APPLICATION SERVERS
- INTEGRATION PLATFORMS
- BUSINESS DECISION MANAGEMENT
- APPLICATION INFRASTRUCTURE

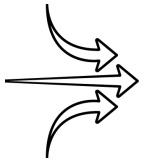
Customer value beyond cost - Digital transformation

RE-BALANCE MAINTENANCE AND INNOVATION

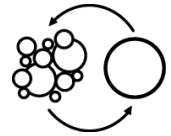
DECREASE COMPLEXITY, INCREASE EFFICIENCY

REDUCE / AVOID VENDOR LOCK-IN, INFLEXIBLE LICENSE MODELS

INCREASE SPEED & BECOME MORE PRODUCTIVE



REMOVE TECHNICAL DEBT & RISK



ADOPT AGILE METHODOLOGIES, DEVOPS

APPLICATION MODERNIZATION & MIGRATION

MODERNIZATION

- BETTER SOFTWARE ARCHITECTURE
  - Modularize
  - Adopt standards
  - Reuse instead of reinvent
  - Clean technical debt
- AGILE INTEGRATION
  - Decouple, integrate, govern APIs, services, applications & clouds
- STREAMLINE APPLICATION LIFECYCLE
  - Accelerate ideas to production
  - Automation & self-service
  - Containerization
  - CI / CD
- CONTINUOUS INNOVATION
  - Agile methodology
  - DevOps principles
  - Collaboration

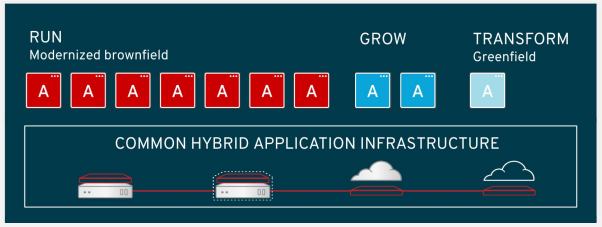
MIGRATION

- APPLICATION SERVERS
  - IBM WebSphere, Oracle WebLogic, Glassfish, Community versions ... ... to EAP / JWS / JDG
- INTEGRATION PLATFORMS
  - **Functional** (TIBCO, WebMethods, JCAPS, Cordys ... to JBoss Fuse)
  - **Data** (JDV)
  - **Messaging** (AMQ)
- BUSINESS DECISION MANAGEMENT
  - ILOG, TIBCO, ActiveMatrix, Pega, Bonita, Oracle & IBM Suites
  - Red Hat JBoss BPMS & BRMS
- APPLICATION INFRASTRUCTURE
  - **OS** (Windows/AIX/Unix to RHEL)
  - **Virtualization** (RHEV)
  - **Containers** (OpenShift)

Red Hat Application Migration & Modernization Program

Red Hat provides the most comprehensive technologies, tools and services to support you

TODAY and TOMORROW



COMBINE TRANSFORMATION

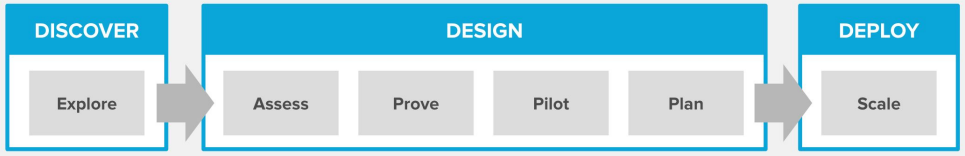
CORE MIGRATION EXISTING & NEW WORKLOADS	APPLICATION SERVERS	ESB & INTEGRATION PLATFORMS	BPM & DECISION MANAGEMENT	APPLICATION INFRASTRUCTURE
MODERNIZATION INITIATIVES ENABLING BUSINESS VELOCITY	BETTER SOFTWARE ARCHITECTURE	AGILE INTEGRATION	STREAMLINE APPLICATION LIFECYCLE	CONTINUOUS INNOVATION

Migration → Modernization

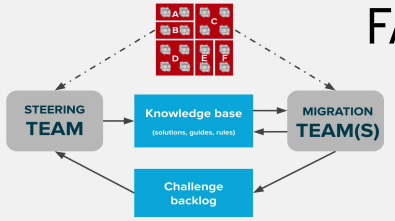
Making old apps new again ← Modern app development

BENEFITS

APPROACH



FACTORY



Red Hat