MICROPROFILE™

Eclipse MicroProfile:

Accelerating Cloud-Native Application Development with Java Microservices

Take your company to the new world of enterprise Java with mission-critical, cloud-native microservices.

A publication brought to you by ECLIPSE FOUNDATION

EXECUTIVE SUMMARY

The pace of innovation has changed. Companies, big and small, are scrambling to implement digital strategies that will enable them to take new products and services to market faster and more efficiently than ever.

In the digital era, customers have become increasingly virtual, mobile, and social — and their expectations are soaring. The proliferation of open source software and emerging technologies such as the Internet of Things (IoT), artificial intelligence (AI), cloud computing, and big data have presented tremendous opportunities and challenges for businesses. Organizations have to become lean and nimble to take advantage of all of these technology trends.

Digital transformation projects have put heavy demands squarely on the shoulders of IT and application developers to execute on evolving business strategies rapidly.

Not your parents’ application development

Developing and tweaking monolithic applications using proprietary software is a thing of the past. Developers are discovering operational efficiencies and cost savings through newer architectures, frameworks, and the adoption of DevOps methodologies.

Application development delays caused by unwieldy software code and vendor-dependent release cycles are no longer acceptable in fast-paced businesses that are looking for a competitive edge. Speed, efficiency, and security of the applications have become key to ensuring an optimal user experience.

Leveraging Java EE expertise for microservices

For more than two decades, Java has prevailed as the programming language of choice among developers.

Features such as stability, portability, and scalability played a big part in securing Java’s dominant position. However, the standards-based approach to the development of Java EE, the de facto platform for large enterprises, had not kept pace with the evolving digital market demands. Developers needed tools and specifications to create dynamic and scalable cloud-native applications and microservices.

	Sep 2018	Sep 2017	Change	Programming Language	Ratings	Change
●	1	1		Java	17.436%	+4.75%
●	2	2		C	15.447%	+8.06%
●	3	5	△	Python	7.653%	+4.67%
●	4	3	▽	C++	7.394%	+1.83%
●	5	8	△	Visual Basic .NET	5.308%	+3.33%
●	6	4	▽	C#	3.295%	-1.48%
●	7	6	▽	PHP	2.775%	+0.57%
●	8	7	▽	JavaScript	2.131%	+0.11%
●	9	-	▲	SQL	2.062%	+2.06%
●	10	18	▲	Objective-C	1.509%	+0.00%
●	11	12	△	Delphi/Object Pascal	1.292%	-0.49%
●	12	10	▽	Ruby	1.292%	-0.64%
●	13	16	△	MATLAB	1.276%	-0.35%
●	14	15	△	Assembly language	1.232%	-0.41%
●	15	13	▽	Swift	1.223%	-0.54%
●	16	17	△	Go	1.081%	-0.49%
●	17	9	▼	Perl	1.073%	-0.88%
●	18	11	▼	R	1.016%	-0.80%
●	19	19		PL/SQL	0.850%	-0.63%
●	20	14	▼	Visual Basic	0.682%	-1.07%

Source: TIOBE Index for September 2018

In answer to the slowdown in Java EE innovation, many vendors, Java champions, Java User Groups, and corporations announced the Eclipse MicroProfile project in June 2016.

Eclipse MicroProfile is a collection of community-driven open source specifications that define an enterprise Java microservices platform — and it plays a vital role in the development of enterprise Java microservices. The open source approach of MicroProfile removes reliance on any single vendor's release cycles and makes the development of microservices a natural extension of a Java EE developer's existing skill set.

> "Eclipse MicroProfile was started as a reaction to the slow pace of Java EE development."
>
> Ivar Grimstad
> Principal Consultant at
> Cybercom Group

Third-party trademarks are the property of their respective owners. Copyright © 2018 Eclipse Foundation, Inc. All Rights Reserved. Made available under the Eclipse Public License 2.0 (EPL-2.0).

MicroProfile and Jakarta EE

Although the Java EE platform remains valuable for many developers and businesses, the community also recognized the need for a cloud-native Java that was ready to participate at the MicroProfile pace. To answer this need, in September 2017, leading software vendors collaborated to move Java EE technologies to the Eclipse Foundation where they will now continue to evolve under the Jakarta EE brand.

With the transition of Java EE to Jakarta EE nearly complete, developers are able to seamlessly mix and match Java EE (Jakarta EE) and MicroProfile APIs in the same application. Since MicroProfile evolves incrementally and can deliver new capabilities at up to 12x the pace of traditional Java EE, with less risk and smaller feature sets, developers can adopt new innovations more quickly.

Many enterprise companies who built monolithic applications during the pre-cloud, pre-container era are now looking for frameworks and specifications that would enable them to move into the more efficient world of microservices.

MicroProfile components build upon the core programming model of Java EE making microservices development much more of a natural experience for Java EE and Jakarta EE developers. Therefore, when moving to or incorporating MicroProfile, the knowledge of Java EE is not lost. MicroProfile defines a specification for the development of microservices while allowing vendors the flexibility to use the specs based on their product requirements.

Customers using MicroProfile benefit from being part of an ecosystem that collaborates on specifications while competing on open source implementations of those specifications. This helps them avoid vendor lock-in, reduce risk profile, and gain a competitive advantage.

> "My customers are looking to use the skillsets of the people that they already have. They have the business domain expertise and are good technologists, however, they don't have a toolkit to move into the world of cloud, containers, and microservices. Now, they would like to see the software and framework in MicroProfile and Jakarta EE evolve in a manner which is familiar enough for their existing engineers to get up to speed and build applications quickly."

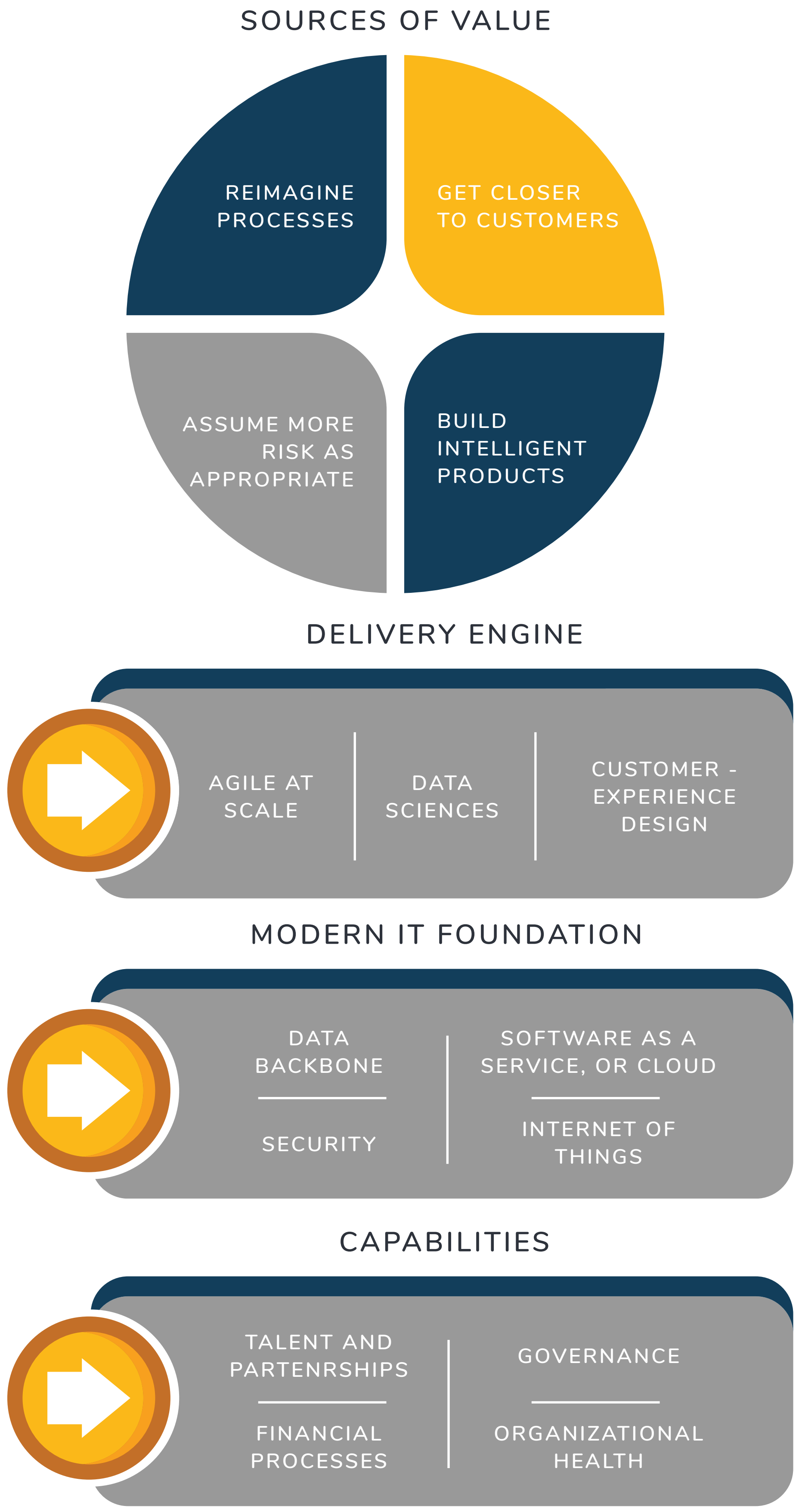
Martijn Verburg
CEO, jClarity

DIGITAL TRANSFORMATION IS NO LONGER JUST A BUZZ TERM

When it comes to digital experiences, customers' expectations are high — and climbing.

Key drivers of digital transformation include:

- Growth and adoption of cloud and containerization technologies that provide on-demand IT resources in a cost-efficient manner
- The proliferation of mobile devices and applications that enable "anytime, anywhere" connectivity for customers and employees
- Big Data and improved analytics that helps with the predictability of business trends and requirements
- Technological advances such as IoT, AI, virtual reality, augmented reality and others that drive operational efficiencies
- Savvy customers and empowered employees who demand personalized interactions on devices and applications of their choice



Source: McKinsey & Company

WITH NEW CHALLENGES COME NEW OPPORTUNITIES

IT teams are on the hook to do the heavy lifting to deliver user-friendly applications at a faster pace while supporting and maintaining traditional workloads. Many technologies, languages, and frameworks have become popular within organizations to help IT tackle these new demands.

Microservices, a variant of the Service Oriented Architecture (SOA), has evolved as a new approach for application development.

Similar to Lego® blocks, microservices are modular, decoupled applications that can be independently deployed, modified, scaled and integrated with other systems and services. They can be updated in hours or days instead of the weeks or months it would traditionally take to overhaul an entire platform.

Applications that use microservices, delivered on open source software, have become a viable alternative to the current application development models.

> "The beauty of microservices is that it does not restrict you to a single programming language. You can build one microservice using MicroProfile, another one with Spring, and another with Node.JS."
>
> *Ivar Grimstad,*
> *Principal Consultant at Cybercom Group*

Open source has become table stakes in application development

Open source software and systems play a significant part in most of today's successful commercial enterprises. Companies of all sizes are using open source tools and projects to develop, grow, and market their products and services.

In fact, recent studies show that adoption of this technology has been growing rapidly, stating, "65 percent of companies are using and contributing to open source." Further, 53 percent of developer teams indicated that their organization had an open source software program or had plans to establish one.

When it comes to open source, traditional competitors often become collaborators who work together to grow ecosystems within which they develop core capabilities and share intellectual property. This collaboration allows them to free up scarce organizational resources to focus on delivering differentiated products and services faster, thereby igniting revenue growth.

From a Java ecosystem perspective, having Jakarta EE and MicroProfile under an open source foundation such as the Eclipse Foundation enables companies to mitigate business risk by accelerating the market adoption of technologies, specifications, and standards.

"Many enterprises are stuck with software vendors whom they cannot influence. They cannot submit bug patches, requests for features, or help steer the direction of the technology. With MicroProfile, you absolutely can. It is one of the few open source communities where an enterprise, who is a user of the technology, can have their voice heard and can make technical and non-technical contributions."

Martijn Verburg,
CEO, jClarity

Significant benefits of open source software include:

- Free of cost to the community who have the rights to use, modify and distribute
- Flexibility due to customization possibilities
- More accessible than proprietary software
- Improved reliability due to the community approach to quicker bug fixes
- Focus on interoperability
- Secure development environment, as a result of more eyes on the source code
- Transparency that makes "who, what, where, when, and why" a matter of public record

"Both MicroProfile and Jakarta EE are open source community collaboration efforts. The intent is to enable choice through multiple implementations and thereby reduce risk for organizations."

John Clingan,
Senior Principal Product Manager at Red Hat

Understanding microservices

According to Martin Fowler, respected software developer, author, and speaker, "the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API."

Under the microservices model, applications are designed as a collection of unique services built around business capabilities that can be deployed, modified, and managed without impact to the application as a whole. Microservices use a distributed architecture where service components are accessed using remote access protocols such as Representational State Transfer (REST).

In contrast, monolithic applications are not modular and are built as single systems. Changes made to any part of the system require the application to be rebuilt and re-deployed as the whole application — proving a leg up for the microservice architecture.

Why adopt a microservices architecture?

1 Strong Module Boundaries

Strong Module Boundaries provide autonomy whereby services can be developed, tested and deployed without adversely impacting the rest of the application

2 Improved scalability

Self-contained services help achieve improved scalability and flexibility to meet the needs of the business

3 Better Manageability

Better Manageability derived from the decoupling of services from the application enables simplified change control and maintenance ensuring reliable performance

4 Technology Diversity

Technology Diversity removes single platform reliance allowing microservices to mix multiple languages, development frameworks, and data-storage technologies

SO, WHAT IS ECLIPSE MICROPROFILE?

Eclipse MicroProfile is a collection of community-driven open source specifications that define an enterprise Java microservices platform. MicroProfile is designed to run enterprise applications with reliability, availability, scalability, and performance. It helps businesses achieve improved agility, faster time-to-market, higher development productivity, easier maintenance, continuous integration, and continuous deployment.

Most open source projects are code-centric. A shared code base is used to collaborate around the implementation of software components and applications.

> “MicroProfile started something unique in the industry. We collaborate around open source specifications, and people can go off and implement projects based on those specs. Customers don’t get locked into a single implementation.”
>
> John Clingan,
> *Senior Principal Product Manager at Red Hat*

Momentum around MicroProfile continues to grow. As of publication, the list of MicroProfile implementations and open source projects includes:

- Red Hat - Thorntail
- Red Hat - Red Hat Application Runtimes
- IBM - WebSphere Liberty
- IBM - Open Liberty
- Payara Foundation - Payara Micro
- Payara Foundation - Payara Server
- Tomitribe - TomEE
- Oracle - Helidon
- Fujitsu - Launcher
- SmallRye
- Hammock

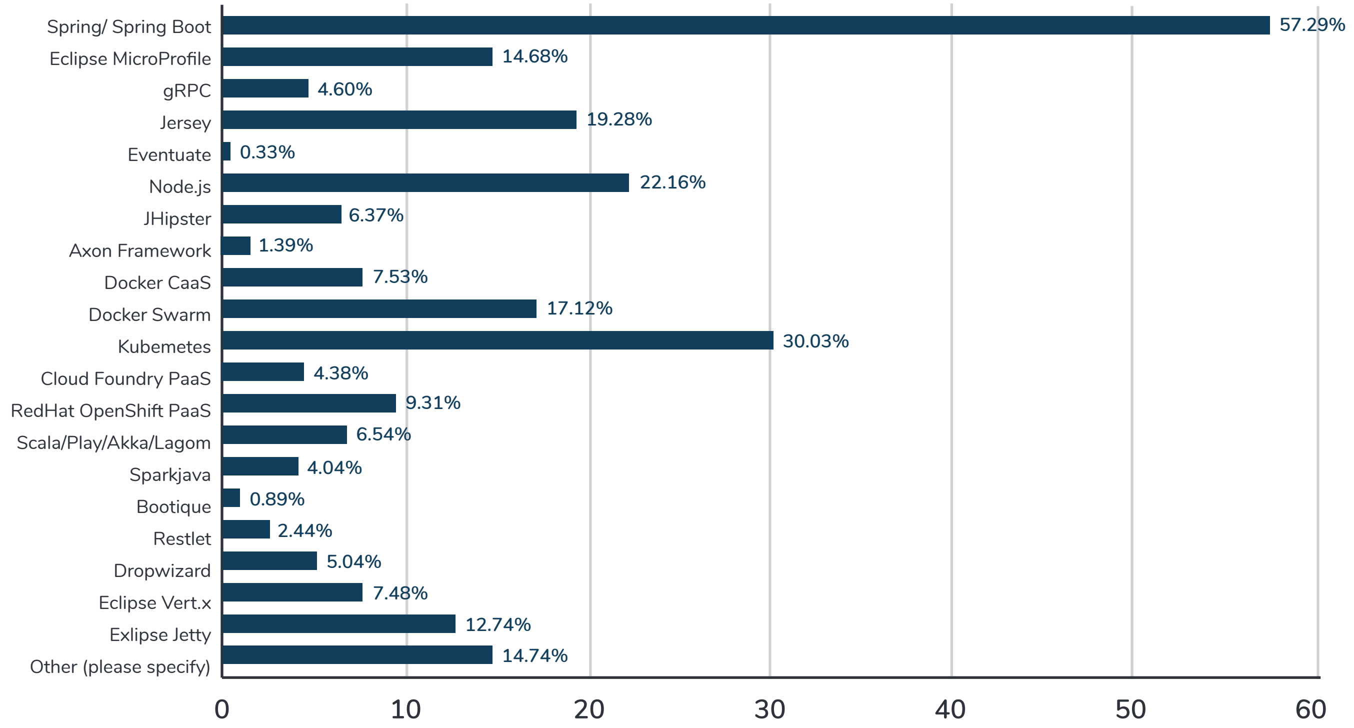
View the complete listing of MicroProfile implementations.

> "Why I believe Jakarta EE and MicroProfile will be a success is because of the players involved. The traditional big players in the application server market (IBM, Red Hat, Oracle, and others) are all supporting it."
>
> Ivar Grimstad,
> *Principal Consultant at Cybercom Group*

MicroProfile plays a vital role in the evolution of enterprise Java because of its fast pace for community-driven innovation, synergy with Jakarta EE, and the potential ways it can influence it. Developers familiar with enterprise Java frameworks will find in MicroProfile a natural progression of enterprise Java into the world of cloud-native application development.

According to the *Jakarta EE Developer Survey*, Java is consistently either first or second in every reputable programming language ranking, with almost 99 percent of developers using it at the workplace — and Fortune 500 companies have no plans to abandon its use.

MicroProfile offers a migration path from what they have today to microservices architecture and the cloud-native capability of their next-generation applications. While developers are using multiple frameworks, the community-led Eclipse MicroProfile is increasing its presence with almost 15 percent adoption at only one year old.



> "If you are a Java EE shop, you should look at Eclipse MicroProfile. It is lighter and contains fewer new things that you can get lost in. Your learning curve is less steep. It's ideal for greenfield microservices."
>
> Ivar Grimstad,
> *Principal Consultant at Cybercom Group*

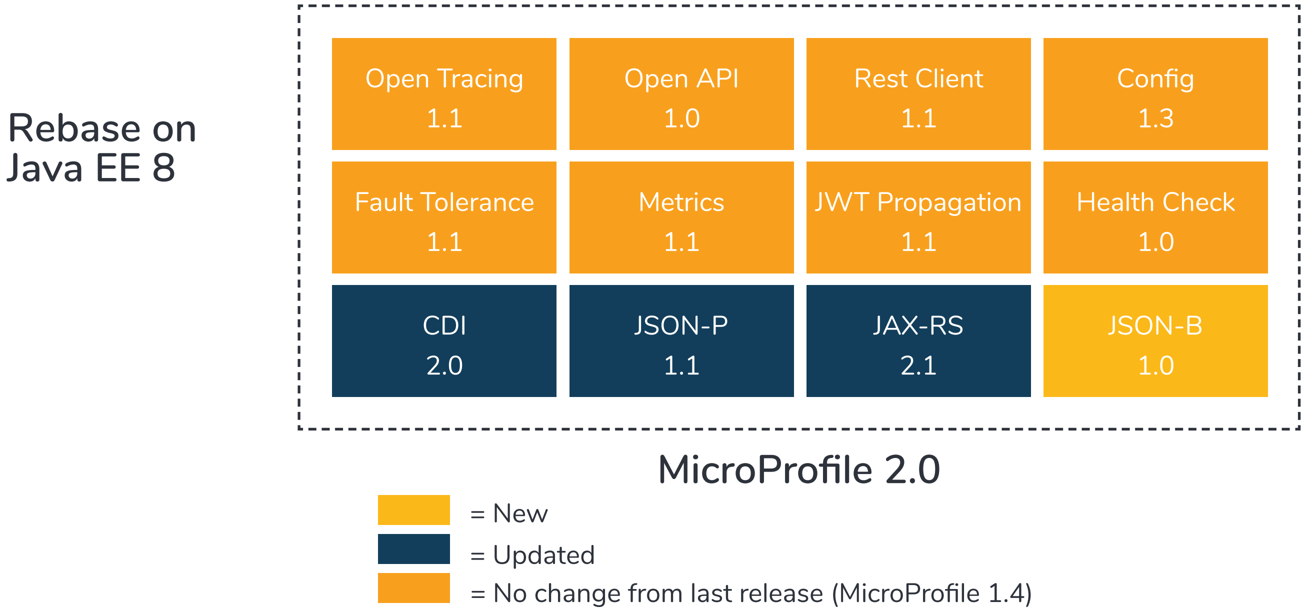
"Eclipse MicroProfile is at the vanguard of cloud-native Java innovation, with many exciting developments quickly moving from incubation to specification to multiple implementations. These new technologies form part of an overall evolution of an industry standard enterprise Java stack for a cloud-native world. Most important for enterprises, it is the commitment to innovation and vendor neutrality that makes for a winning combination for enterprise application developers."

Mike Milinkovich,
Executive Director of the Eclipse Foundation

According to the Eclipse Foundation's global survey of 1,800 Java developers, 46 percent of organizations are already building microservices. Another 20 percent say they plan to build microservices in the next year. The rate is only going to increase as developers become more familiar with MicroProfile and more adept at building microservices. Nearly half of respondents (45 percent) said they are using Java to build microservices, with another 21 percent planning to join them in the coming year. The survey also finds that 15 percent of respondents have already adopted MicroProfile, which was first announced in June 2016.

The Java community's desire to see enterprise Java evolve to support cloud, containers, microservices, serverless, API management, and other digital transformation related capabilities is clear. MicroProfile has been rapidly fulfilling the need to extend the Java EE platform to help developers create portable cloud-native applications.

With six major releases in less than two years, the MicroProfile community is helping developers leverage innovations sooner.



Rebase on Java EE 8

🟧 Open Tracing 1.1	🟧 Open API 1.0	🟧 Rest Client 1.1	🟧 Config 1.3
🟧 Fault Tolerance 1.1	🟧 Metrics 1.1	🟧 JWT Propagation 1.1	🟧 Health Check 1.0
⬛ CDI 2.0	⬛ JSON-P 1.1	⬛ JAX-RS 2.1	🟨 JSON-B 1.0

MicroProfile 2.0

🟨 = New
⬛ = Updated
🟧 = No change from last release (MicroProfile 1.4)

"Many enterprise customers who have Java EE or Spring "legacy" applications are looking to migrate those applications on to a more modern stack. Now that they have MicroProfile and Jakarta EE to choose, they can have a smoother migration path and a technology stack that vendors like jClarity can easily support. Customers don't have to incur the expense of moving to an entirely different language ecosystem. It is certainly going to add longevity to people's businesses"

Martijn Verburg,
jClarity

BUSINESS BENEFITS: PROVIDE A FRAMEWORK FOR BUILDING OUT THE BUSINESS CASE

Developing the business case for Eclipse MicroProfile

The business case development and approval processes within organizations vary. However, the underlying costs-benefits analysis and principles mostly remain the same. The following factors will help you determine if your organization can benefit from adopting a microservices architecture in general and specifically a community-driven open source specification for enterprise Java microservices such as Eclipse MicroProfile.

Is your business a good fit for enterprise Java microservices?

To determine if enterprise Java microservices are a good fit for your company, ask yourself the following questions.

- Do you interact with your customers primarily through software and web interfaces?
- Are software licensing costs preventing your applications from achieving "Internet scale?"
- Is your IT development team slow to react to change requests from the business?
- Do you have scalability issues of your applications, which negatively impact the growth of your business?
- Do you have an existing team of enterprise Java developers?
- Do you foresee business growth and integration requirements where flexibility in application development is crucial?

If you answered "yes" to most of the questions above, your business could likely take advantage of a microservices architecture to drive operational efficiencies and improve time-to-market for your products.

> The great thing about MicroProfile is that multiple vendors are working on it to define the base profile — APIs and specifications. There are multiple implementations on top of which vendors can add their bells and whistles."

Kevin Sutter,
STSM, MicroProfile Architect at IBM

The value of an Eclipse MicroProfile implementation

Over and above the general benefits of microservices listed above, the value proposition that MicroProfile brings include:

Community collaboration

MicroProfile is an open source project run by the community. No single vendor controls the evolution and maturation of the specification.

Freedom of choice

Many vendors have implemented MicroProfile as part of their software stacks. Customers can choose the implementation that works best for their environment.

Faster evolution

Improved functionality is delivered in time-boxed releases, which allow developers and customers to start leveraging updates in their projects sooner rather than later.

No need for re-training

MicroProfile builds upon Java constructs and is a natural progression for existing enterprise Java developers. There is a little-to-no learning curve.

Portability

Users can develop applications that can run on multiple MicroProfile implementations.

Interoperability

MicroProfile ensures interworking between implementations through interoperability tests. Interoperability is explicitly defined for MicroProfile JWT security specification.

Multiple ways to use APIs

MicroProfile APIs provide easy-to-use interfaces — context dependency injection-based (CDI), programmatic, command-line, and file-based.

Resources at your fingertips

Various resources that help with the adoption of MicroProfile are made available to the community. Some of these are listed on the next page:

> "There is no need for a big-bang approach to moving from monolithic applications to microservices. You can carve out slices, such as business domains, and create microservices using the new technologies."

Ivar Grimstad,
Principal Consultant at Cybercom Group

Resources:

- A thorough set of artifacts for each API that include Test Compatibility Kit (TCK), Javadoc, PDF document for download, API Maven artifact coordinates, Git tags and downloads (specification and source code)
- MicroProfile Google Group to facilitate community communication and discussion
- MicroProfile Sandbox to provide an environment for incubating ideas and code examples that eventually turn into a separate repository towards a new specification
- A wizard-driven starter application that generates MicroProfile-related assets you need to quickly get started with your microservices projects
- A set of MicroProfile code samples for your developers to reuse

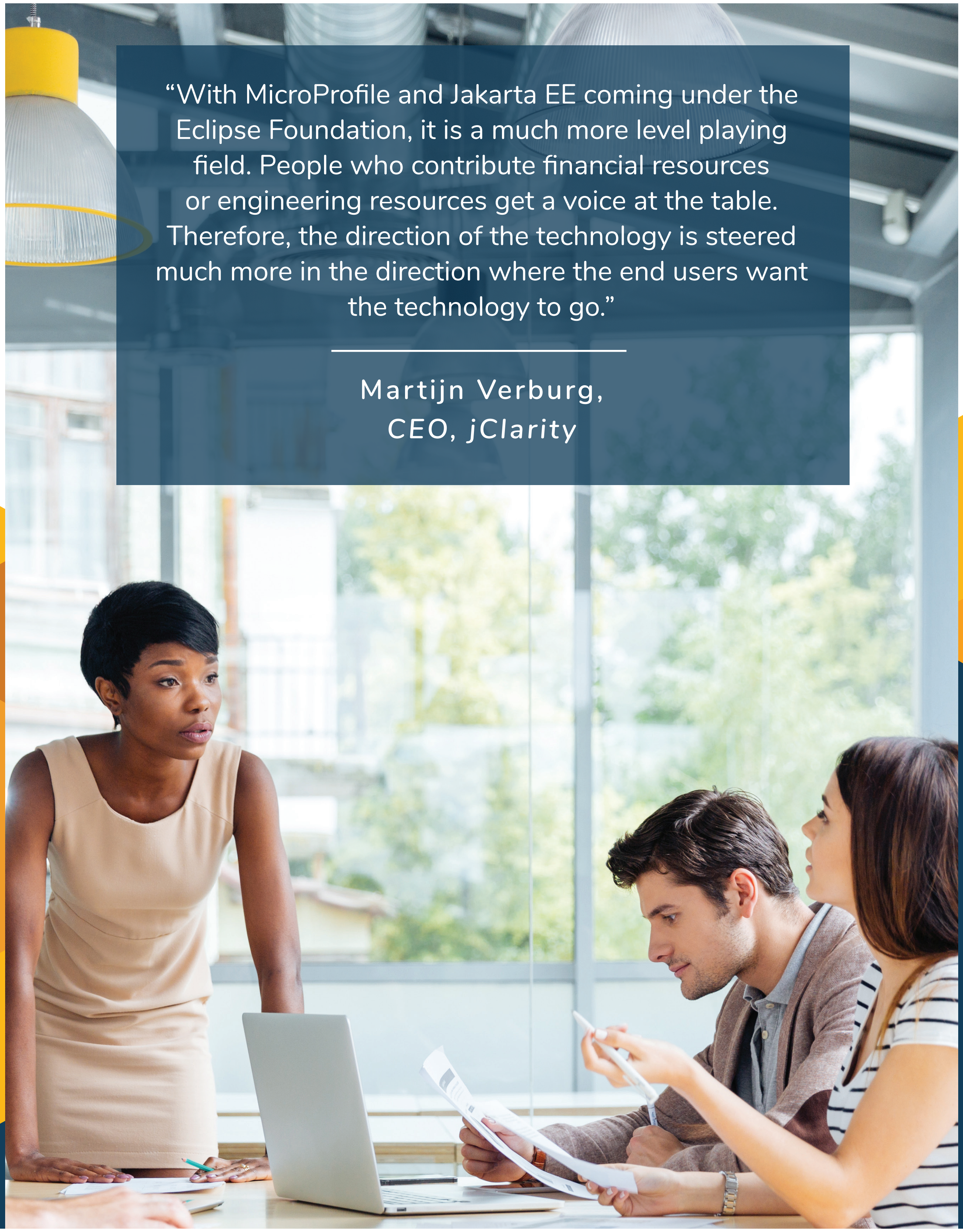
While determining if microservices are a good fit for your organization, it is essential to understand some of the caveats.

Depending on the nature of your business, a monolithic application that works well might be all your business needs.

Development teams can often underestimate the complexity of implementing a microservices architecture. It should be well understood that the implementation of microservices is not about deploying multiple smaller monolithic applications. The switch to a microservices architecture is not a one-time change — it is a continuous process that requires the adoption of new technologies and frameworks.

While the concept of using "the right tool for the right task" works most of the time, too many niche tools and frameworks can leave you with a maze of complexity and skill-set gaps that slow your business down.

> "With MicroProfile and Jakarta EE coming under the Eclipse Foundation, it is a much more level playing field. People who contribute financial resources or engineering resources get a voice at the table. Therefore, the direction of the technology is steered much more in the direction where the end users want the technology to go."
>
> Martijn Verburg,
> CEO, jClarity



The Eclipse Foundation's vision is to be the leading community for individuals and organizations to collaborate on commercially-friendly open source software.

> “We need organizations such as the Eclipse Foundation for the governance they provide. It is very worthwhile to promote and fund them.”
>
> Kevin Sutter,
> STSM, MicroProfile
> Architect at IBM

THERE ARE SEVERAL WAYS TO LEARN MORE ABOUT MICROPROFILE AND TO GET INVOLVED IN DRIVING THE EVOLUTION OF ENTERPRISE JAVA MICROSERVICES:

To learn more, visit the MicroProfile website which includes a blog, FAQs, and other useful information.

The Eclipse MicroProfile project page offers more technically oriented information on the project, its governance, specifications, and links to code repositories.

To join in the discussion, check out the MicroProfile discussion group.

To participate in MicroProfile calls, meetings and events, check out the project calendar.

Third-party trademarks are the property of their respective owners. Copyright © 2018 Eclipse Foundation, Inc. All Rights Reserved. Made available under the Eclipse Public License 2.0 (EPL-2.0).

ABOUT US

The Eclipse Foundation is a not-for-profit organization supported by more than 275 member organizations who value the Foundation's unique Working Group governance model, open innovation processes, and community-building events. The Foundation provides a global community of individuals and organizations with a mature, scalable, and business-friendly environment for open source software collaboration and innovation. It is home to MicroProfile, Jakarta EE, the Eclipse IDE and more than 350 open source projects including runtimes, tools, and frameworks for a wide range of technology domains such as the Internet of Things, automotive, geospatial, systems engineering and many others.

Why consider participating in the Eclipse Foundation's projects?

Organizations interested in participating in Eclipse MicroProfile, as well as Jakarta EE and other Eclipse open source projects, are expected to become members of the Foundation. Membership demonstrates your commitment to the success of MicroProfile and the objectives of providing a well governed, vendor-neutral solution in the marketplace. Membership in the Foundation also has a number of additional broader benefits, including:

Ability to join and participate in the Eclipse Foundation Working Groups, industry collaborations that develop new open source software platforms

Take advantage of marketing programs that target the Eclipse developer community, including exposure for commercial products on Eclipse Marketplace Client

The right to participate in the Eclipse Board of Directors and guide the strategy of the Eclipse Foundation (based on established guidelines)

Receive discounts on sponsorship and registration fees for Eclipse Foundation events

The opportunity to network with other member organizations through Foundation-run events and one-on-one introductions by Foundation staff

The ability to use the Eclipse Foundation member logo on your site and have your logo listed on our members' page and in other promotional material

To find out more about joining the Eclipse Foundation, please visit eclipse.org/membership