D3.1.1. Designs of the new visualizations.

Programme	ITEA3		
Challenge	Smart Engineering		
Project number	17038		
Project name	Visual diagnosis for DevOps software development		
Project duration	1st October 2018 – 30st June 2022		
Project website			
Project WP	WP*3* - Visualizations		
Project Task	Task *3.1 – Visualization types*		
Deliverable type	X	Doc	Textual deliverable
		SW	Software deliverable
Version	1.0		
Delivered	09/12/2019		
Access	X	Public	
		Abstracts are public	
		Confidential	

Version history

Version	Date	Author	Notes
0.1	25.11.2019	Kari Systä (TAU)	Initial draft
0.2	02.12.2019	Kari Systä (TAU)	First version with content in all sections
0.9	02.12.2019	Kari Systä (TAU)	For review
1.0	09.12.2019	Kari Systä (TAU)	Incorporated review comments by Paris Avgeriou

1 Introduction

This document (D3.1.1) gives a top-level design of the VISDOM visualizations. The purpose is to communicate the innovation of the VISDOM project and guide the forthcoming research and implementation work in WP3.

The detailed design will be done as a part of the implementation during the project. No separate design document has been planned on the ITEA FPP.

2 EKG

2.1 Background

The FPP document describes EKG as follows:

> *"The **EKG** of a software project will show the rhythm of pace of a project. Software projects typically include various repeating tasks and iterations, such as sprints or delivery cycles. Such phenomena can be shown as repeating visual patterns, similar to the rises and falls of an EKG curve. Problems will be quick and easy to spot as variations in the patterns and especially difference between planned and real state. This is a novel idea that not been realized so far, although some similar ideas have been presented by Lehtonen"*

This EKG visualization is based on a similar idea with the concept of EKG in medicine. In Medicine EKG uses 12 different inputs to generate a repeating visualization shown in Figure 1. Medical staff can inspect this diagram to find various hearth-related problems. An expert can differentiate normal and pathological patterns from each other.

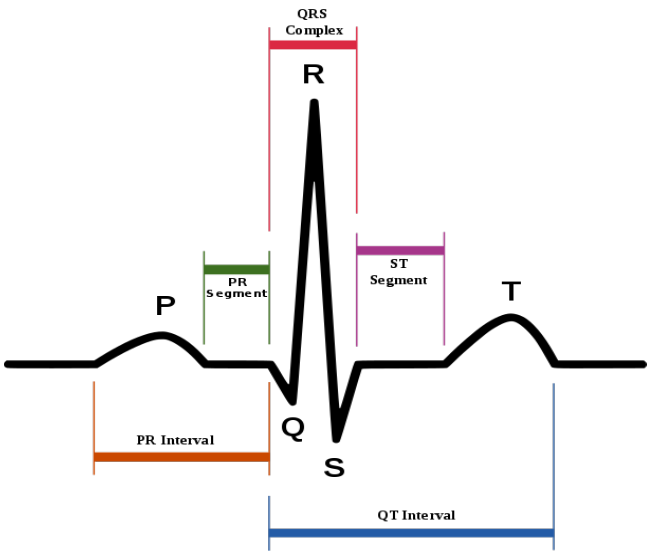

Figure 1. EKG in medicine (wikipedia commons)

2.2 Description of the idea

EKG in VISDOM is based on a similar idea: a set of input data is used to generate visualizations that different stakeholders based on their interests and expertise can analyse. The data sources and visualization will be explored in the research of VISDOM WP3, demonstrated in WP4 and exploited/disseminated in WP5.

The core idea of software engineering EKG can be described with the following attributes

- Visualization consists of data from several sources, e.g., issue tracking, version management, testing tool, CI/CD pipeline.
- Visualization is composed of repeating shapes representing cyclic activities like frequent deployments.
- Visual appearance of normal and unusual situation is different

- The visualized deviations represent some interesting phenomenon to some stakeholders.

In the next section we describe some earlier examples of this idea. None of these have been developed beyond research prototypes.

2.3 Prior examples

In our earlier research we have discovered two candidates for EKG-like visualizations.

In a student project of Tampere University, the students used Agilefant[1] to track and report their work and progress. In a master thesis [2] some simple visualizations have been generated from that data. An example of the generated visualizations is shown in Figure 2. The students were expected to follow the Scrum methodology, and the teachers analysed the student projects from these visualizations. In this example the student group learned how to implement time-boxing in Scrum during the project: the completion of the work end of the sprint was typically not very visible in the first sprints (although the end date of the sprint was reached, the work continued), but the behaviour improved during the course. This is the case also in the example group depicted in Figure 2. In this visualization the learning process can be discovered visually: all tasks are finished, and the sprints are properly completed. For instance, in the case of one example group, recognized shapes as shown in left part of Figure 3. This led to the EKG idea depicted in the right hand side of Figure 3. The irregularity of the shape could be an indication of problems.

Figure 2. Example visualization of a project. The horizontal lines represent "tickets" and circles events on the ticket.

[1] https://www.agilefant.com

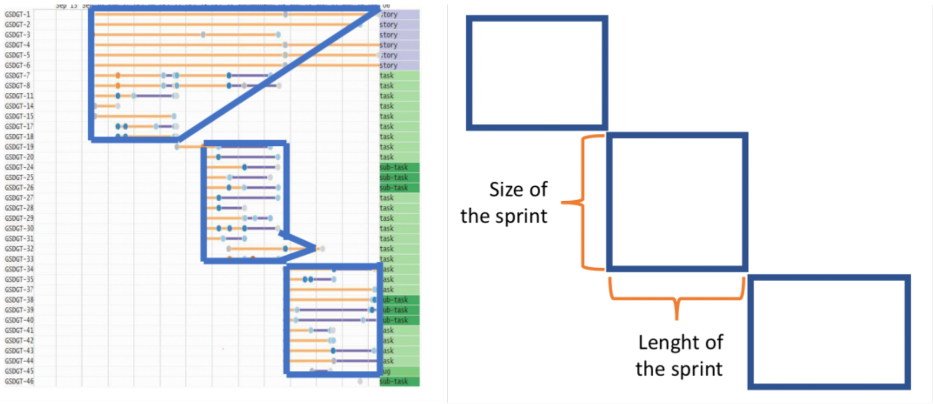

Figure 3. Initial sketching of the first EKG.

A similar idea has been proposed by Lehtonen [1]. He gives an example of a ”reference visualization” (see Figure 4), that similarly to above comes from feature development information. In this visualization, a horizontal line starts when the development of a feature starts and ends when the feature is deployed. In a well-working organization, this process should then result in the visualization shown in Figure 4; a set of new features are developed, and then deployed together.

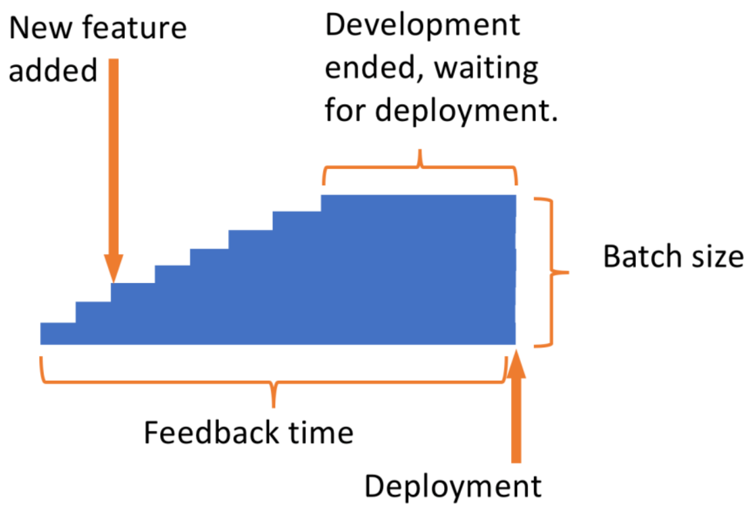

Figure 4. Reference visualization as proposed by Lehtonen [1]. The feedback time is the minimum time before the development can receive used feedback.

Similarly to the above, the deviation in size or shape of repeated instances of this visualization can be used like EKG. In his work Lehtonen shows examples of ideal and real visualizations. See Figure 5 for an example.

Figure 5. Example of ideal and real CD pipelines according to Lehtonen [1].

Sometimes repetitive implementations of *radar charts* share some of the features of EKG – a sudden change in the shape can be a symptom of problems. For instance, the Q-Rapids project [3], uses radar charts in their dashboards, as shown in Figure 6.

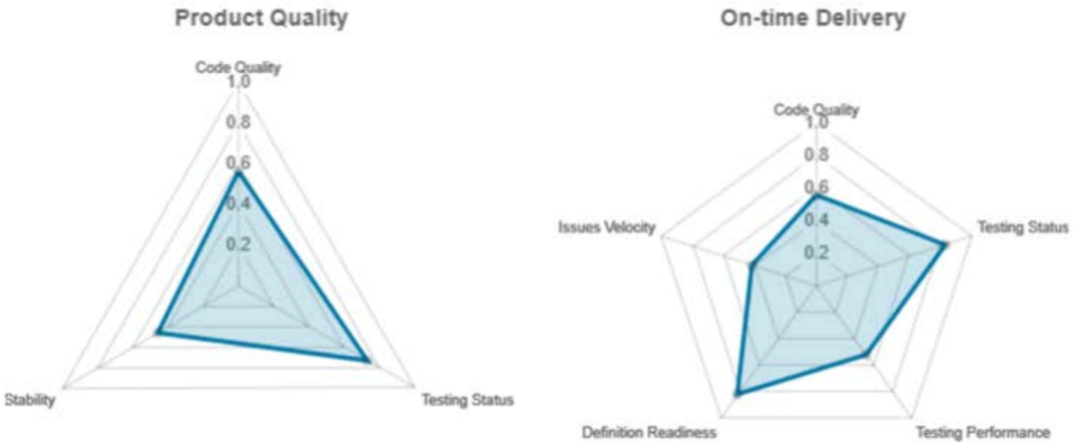

Figure 6. Radar charts in Q-Rapids dashboard [3]

2.4 Innovation and next steps

Although some initial research exists, as shown above, the idea is still immature and not really tested. In the VISDOM project we will explore this EKG concept more by creating Proof of Concept implementations and use them in real projects in order to assess and refine them. Especially, we will implement EKG visualizations that fit the needs of VISDOM use cases and are specific to DevOps. Specifically, we should add information about various aspects of DevOps, including quality, automatic testing, hosting (e.g. load and scaling), usage etc.

As for all visualizations, we aim at EKG-visualizations that utilize several data sources (tools) and thus reach for beyond any current state of the art.

The visualizations that are targeted show important aspects for the stakeholders, and feedback will be collected from selected stakeholders in the companies.

3 Pulse

3.1 Background

The FPP document describes Pulse as follows:

> *“**Pulse(s)** are used to visualize the different paces of different parts of the project (process). Visualizing the pulses, i.e., paces, of sub processes within a project can help notice bottlenecks or other local problems that can harm the project later.”*

3.2 Description of the idea

In medicine the pulse is often shown as a sequence of EKG shapes as shown in Figure 7.

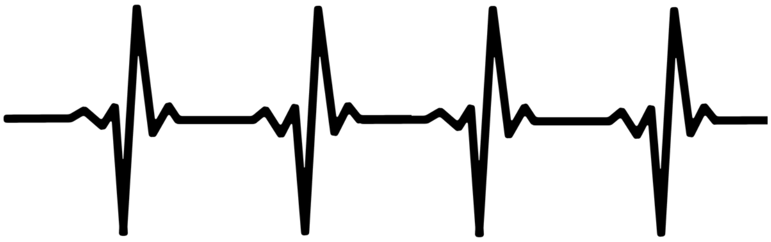

Figure 7. Pulse in medicine.

In the VISDOM project we assume the pulse is rather an additional visualization than replacement of EKG - as used by Lehtonen. (See the red vertical bars in Figure 8.) Pulse concentrates in the rhythm of while EKG visualizes the other indicators.

3.3 Prior examples

Figure 8 shows some visualizations presented in [1]. The upper part shows a target (reference) behavior in the project in question. This part of the development consists of four major releases, that are about the same size. The height of the shapes is relational to number of the new features in the release. The development continues as long as the slope continues. When the slope ends the horizontal part indicates waiting time for deployment. During that waiting time, the development team can start with next bunch of features, as shown by the vertical overlapping of the shapes.

The smaller shapes are intermediate releases that fix bugs or response to urgent customer requirements. These are smaller both in terms of size and time. They, however, double the pulse (see red bars).

The lower part shows one potential real situation where the size of shapes (and of deployments) vary. In that case, the feedback time for some features is very long, and the pulse of deployments is very irregular.

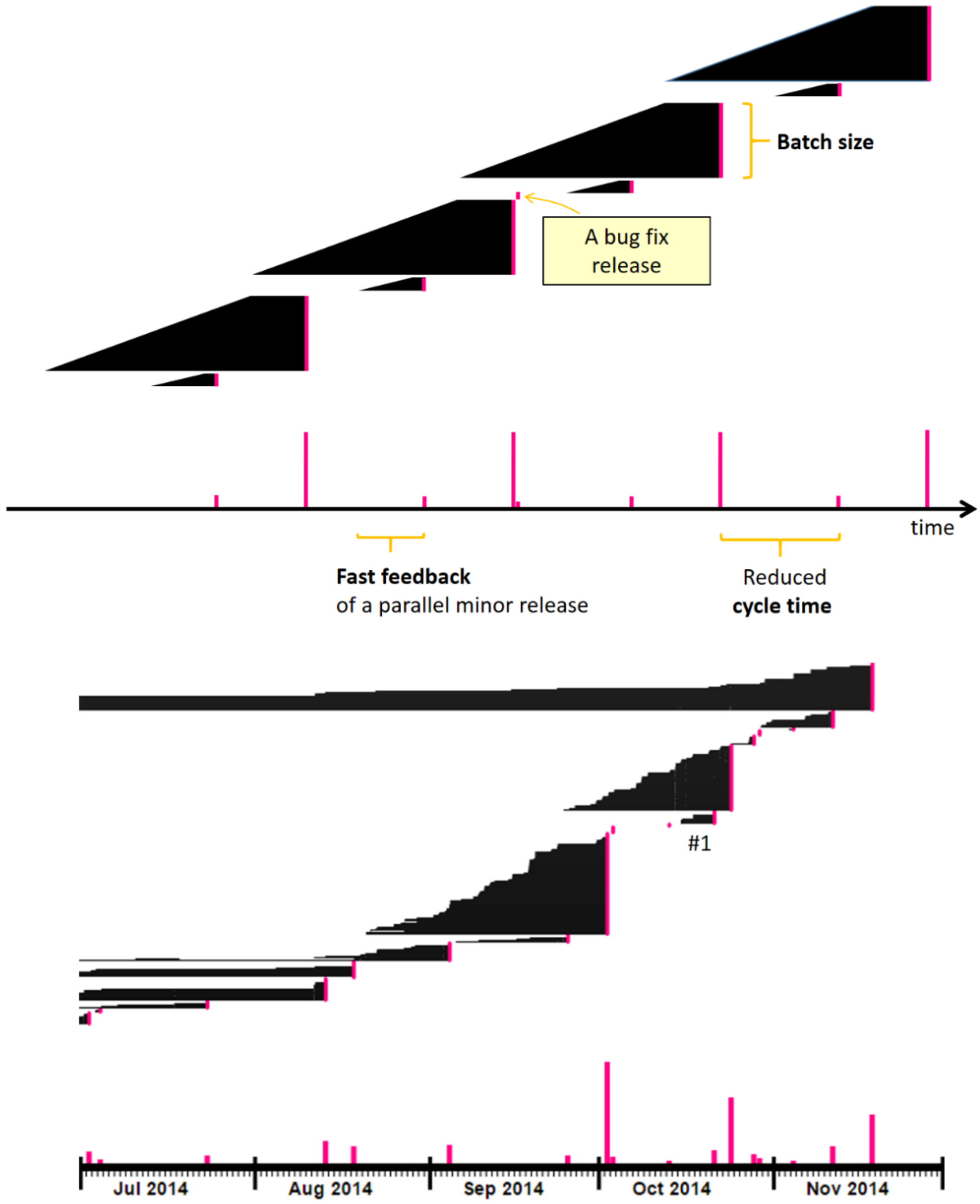

Figure 8. Pulse (red bars) added to a visualization. Reference behaviour is shown in above and a real data below. [1]

3.4 Innovation and next steps

Pulse is not necessarily a very novel way to visualize, but is can be combined with other visualizations in a novel way. Furthermore, our project will search for new types (sources) of repeated pulses.

The next step is to analyze existing Pulse-type of visualizations in the state of the art.

4 Blood-pressure

4.1 Background

The FPP document describes blood pressure as follows:

> *“**Blood pressure** indicates the stress level of a project - for example, an unusually high level of activities in proportion to how much the project is progressing or a sudden increase in the number of contributors would show as high blood pressure. A rise in the project’s blood pressure will be a trigger to make changes and react early enough.”*

4.2 Description of the idea

The anticipated visualizations will be rather simple and well-known charts, but the novelty is in the data it shows. We intend to find visualizations that show under- and over-loading situations of the DevOps team and/or the used infrastructure. Possible examples include:

- The workload of the team is beyond a sustainable level. Possible indicators for this include increased overtime, increased error rate, changed style and length of comments.
- The development pipe-line is overloaded and should be optimized or more capacity should be added. Such conditions slow down the development and increases the stress level of developers.
- Overloading of the hosting environment, that many of the cloud monitoring tools are used to track.

4.3 Prior examples

As also noted in our state of the art document [4], the cloud monitoring tools can be used to monitor the different load-level aspects of the computing infrastructure. An example of New Relic [6] visualization is shown in Figure 9.

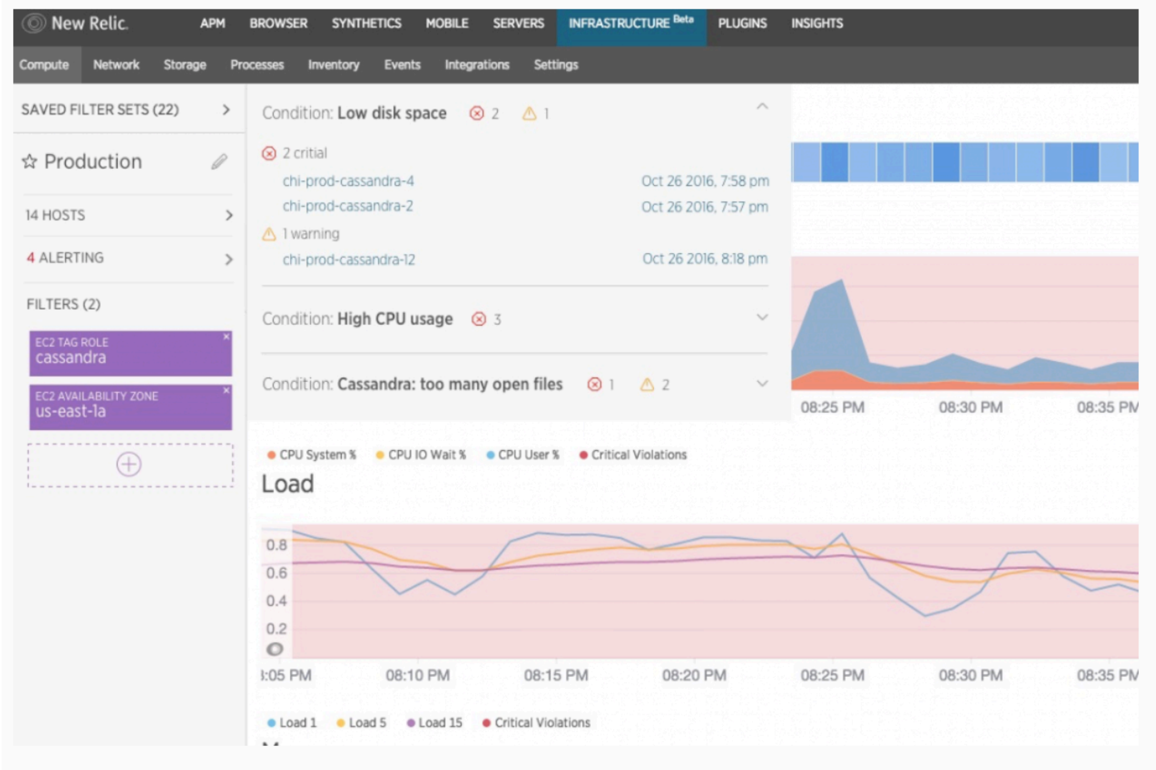

Figure 9. New Relic visualization showing load of computing infrastructure [7].

Similar approaches could be used in our VISDOM.

However, monitoring of the stress-level of the team is a less explored topic.

4.4 Innovation and next steps

As we foresee that the blood-pressure visualization will be quite simple, the focus will be in finding and selecting of the relevant indicators and discovering of the ways to collect that data. Thus, the first step will be a literature study of stress indicators in software engineering teams.

5 X-Ray

5.1 Background

The FPP document describes the X-ray as follows:

> *"**X-ray** is needed to actually discover the "broken bones" of the project - code, architectural artefacts, and processes. While the aforementioned visualizations will give stakeholders instant feedback on the state of the project, in order to locate the cause of a problem when one arises, a detailed yet understandable view of the software and processes is required."*

5.2 Description of the idea

The common denominator of the X-Ray visualizations is the view of the internal structure (skeleton) beyond all the details. The underlying root cause that cause significant problems in a DevOps process are meant to be visualized. A typical example of such root cause are technical debt issues, that are causing extra maintenance effort (slowing down the development pace) and result in errors and bugs.

5.3 Prior examples

Different reverse-engineering methods and tools are typical examples of X-Ray visualizations. Our FPP document [5] lists four approaches in the state-of-the art section: visualization of software structure. software evolution, software behaviour and development process. Especially, the software structure is something that can be visualized with the X-Ray approach, and our non-funded partner Softagram conducts business on this area. In addition, we foresee opportunities to additional visualizations following the X-Ray metaphor:

- Showing skeleton of the communication structure of the DevOps Team
- Showing skeleton of the used cloud infrastructure of the deployed software. This may include the structures defined by the developers (infrastructure as code) and autoscaling implemented by the infrastructure.
- Showing usage and load.
- Showing quality aspects of the software, e.g., on top of an architecture skeleton. Some of the visualizations in TICS Framework by VISDOM partner TIOBE belong to this category. An example from TICS is shown in Figure 10.

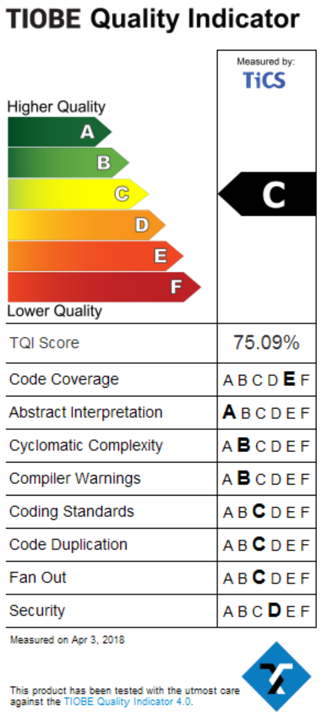

Figure 10. Quality skeleton, as generated by TIOBE TICS [8].

5.4 Innovation and next steps

The main innovations of VISDOM are planned to be based on DevOps approach, use of several tools as data sources, and provide relevant information to various stakeholders. Thus, the research goal is to develop and try new DevOps X-Ray visualizations that include code and operations. At the same time companies work on their own goals, for instance Qentinel works quality visualization of DevOps projects.

6 References

[1] Timo Lehtonen, Metrics and Visualizations for Managing Value Creation in Continuous Software Engineering, Doctoral Thesis, Tampere University of Technology.
[2] Lassi Väisänen, Visualizing software project's project management data, Master Thesis, Tampere University of Technology, 2018 (In Finnish).
[3] https://www.q-rapids.eu
[4] VISDOM Deliverable D1.1.1 Public state of the art document, December 2019
[5] VISDOM Full Project Proposal, v2.22, October 2019.
[6] https://newrelic.com
[7] New Relic Adds Infrastructure and Mobile App Tracking to Its Monitoring Service, https://thenewstack.io/new-relic-adds-infrastructure-mobile-app-tracking/
[8] https://www.tiobe.com