Editor: **Cesare Pautasso**
University of Lugano
c.pautasso@ieee.org

Editor: **Olaf Zimmermann**
University of Applied Sciences
of Eastern Switzerland, Rapperswil
ozimmerm@hsr.ch

Why They Just Don't Get It

Communicating about Architecture with Business Stakeholders

Jochem Schulenklopper and Eelco Rommes

The infamous gap between business and IT manifests itself every time people communicate across it. As professional services consultants, Jochem Schulenklopper and Eelco Rommes have witnessed it many times during meetings and workshops. Visual notations that might be intuitive for IT specialists but impervious for business stakeholders can't bridge the gap. So how can these communities avoid talking past each other and reach a common understanding? Read this insightful story to find out. —*Cesare Pautasso and Olaf Zimmermann, department editors*

HAVING BEEN EXPOSED to IT architecture descriptions of many kinds and flavors, we noticed that many IT architects don't really speak their business stakeholders' language. (In this article, we address the rather broad group of IT architects because they're particularly lucky to have businesspeople in their stakeholder group. Nevertheless, architects of all kinds and backgrounds will discover valuable lessons here.) Common architecture languages favor correctness and completeness over expressiveness and clarity. The models and descriptions created using these languages fulfill the architect's needs well but can be understood only by those with a similar mindset and background. Other stakeholders are often dazzled by the abstract symbols and terminology.

Here, we present visual-communication practices that have served us well to bridge the gap with business stakeholders. These practices stem from disciplines such as psychology, graphic design, communication science, and cartooning. They're intended to aid all architecture stakeholders in understanding, analysis, and discussion. (For a crash course in visual communication, see the related sidebar.)

What's the Problem?

Over the past five years, we've trained about 300 novice and experienced IT architects in all kinds of organizations—from enterprise architects in the airline industry to infrastructure architects in municipalities. One problem that keeps popping up is how to communicate effectively with "the business." "These businesspeople just don't get it," our trainees tell us. "What's wrong with them?"

0740-7459/16/$33.00 © 2016 IEEE

A CRASH COURSE IN VISUAL COMMUNICATION

An effective architecture visualization provides insight and overview in a way that's hard to achieve using only words. Creating such a picture requires knowledge of basic visual-communication principles from fields such as psychology,[1] communication science, design thinking,[2] and even cartooning[3]—subject areas that are unknown territory to many IT professionals. Here are six lessons we've learned from creating architecture visuals and teaching others how to do it.

First, catch the eye's attention with attributes of preattentive processing. Our eyes and brains are wired to perceive basic visual attributes of objects without conscious effort. Our attention then focuses on those objects that are different, even in minor ways. Architects can use this to focus the audience's attention by varying shape, line width, size, and color, for example. If a visualization has six red boxes and a single green one, you know which one will draw the audience's attention first.

Second, know your gestalt principles. Gestalt theory tells us that people organize visual elements into bigger groups.[4] They then perceive these *gestalts* as a new coherent unit. People unconsciously use some principles in grouping things. Applying these gestalt principles consciously is a powerful way to help viewers better understand a visualization. By using subtle clues regarding which things belong together, you can add more information to a visual without overcrowding it. The example architecture visualizations in the main article show some of these principles in practice. Placing applications near each other, giving them a container, using similar colors—gestalts are everywhere.

Third, apply color theory. Colors can evoke strong emotional responses and carry subtle yet rich messages. (For example, see the books of Faber Birren, who wrote extensively on color theory and psychology.) A manager might see a red application as a bigger concern than a green one. A smart, communication-wise architect applies color effectively in diagrams and images.

Fourth, use icons, logos, and images. Many architecture visuals are filled with jargon and abstract symbols. For example, ArchiMate features more than 60 abstract symbols, sometimes with variations in meaning transcending the comprehension of most stakeholders (architects included). (For a summary of the ArchiMate language notation, see http://pubs.opengroup.org/architecture/archimate2-doc/appendixA.html.) Imagery, simple icons, and metaphors (preferably from the organization's vocabulary) are much easier to understand. Make your message easy to understand by enabling the power of visual thinking.

Fifth, apply design thinking and graphic design. An eye for design is not only an innate talent but also a skill you can learn. Some excellent books summarize the experience of decades of visual design, in bite-sized chunks you can apply immediately. For examples, see the "Further Reading" sidebar.

Finally, sketch to facilitate and record thinking. Sketching helps you not only communicate concepts but also build a common understanding for all participants. We often instruct our colleagues and customers to sketch their message or architecture, instead of using words or a professional diagram tool. Some sketches are even worthwhile to keep around—not just as a temporary or intermediate result—and we're not afraid to add them to the documentation.

References

1. S. Ting, "5 Psychology Studies That Tell Us How People Perceive Visual Information," Piktochart, 31 Oct. 2014; http://piktochart.com/5-psychology-studies-that-tell-us-how-people-perceive-visual-information.
2. D. Norman, *The Psychology of Everyday Things*, Basic Books, 2013.
3. S. McCloud, *Understanding Comics*, William Morrow, 1994.
4. K. Koffka, *Principles of Gestalt Psychology*, Lund Humphries, London, 1935.

Our usual answer is that businesspeople aren't the problem. The architects are. The architect's tools for communicating with stakeholders are blunt and often unsuitable.

How Architects (Try to) Communicate

As birds come in flocks and sheep come in herds, so IT architects come in discussions. We architects love to communicate. We've even invented new languages and tools for it. But what happens when we use these languages to communicate with nonarchitects?

Imagine you're an IT architect, walking into a meeting room packed with managers. You don't stress; you've done your homework. You connect to the projector, bring up the ArchiMate (www.opengroup.org/subjectareas/enterprise/archimate) model of the company's architecture, and start talking happily. But just when you reach the really juicy bits, someone interrupts. Usually, it's a senior manager who asks with some level of politeness what on earth you're talking about. Only now do you notice that half the audience looks irritated or puzzled—or both—and the rest are fiddling with their phones. Slightly panicking, you try to talk them through the model. "This stick man is called an actor. He's a user of an application, but he might also be another system. Ellipses are services provided by the application. They're used as a step in the business process shown here."

More frustrated faces. More fiddling with phones. You could have known: meetings with pointy-haired bosses are useless. They just don't get it.

But why don't they get it? Because you tried to explain a language instead of an architecture.

Effective communication means adapting the message to the audience, speaking their language, and selecting only the relevant information and presenting it accessibly. To make matters more interesting, architects communicate with a diverse set of stakeholders: end users, developers, operators, financial professionals, and business managers, to name a few. Each of them speaks a slightly different language, has his or her own interests, and has a different level of understanding of the system. Gregor Hohpe calls this "riding the elevator": architects "[move] up and down between the board room and the engine room of a large enterprise."[1]

FURTHER READING

Much has been written about graphic design, sketching, and visualizing information. We've found the following books and blogs especially helpful.

BOOKS

- *Universal Principles of Design* (Rockport Publishers, 2003), by William Lidwell, Kritina Holden, and Jill Butler, contains a library of graphical principles and their application in numerous examples.
- *The Sketchnote Handbook* (Peachpit Press, 2012), by Mike Rohde, is the resource on sketchnoting: taking notes in the form of drawings.
- *The Visual Display of Quantitative Information* (Graphics Press, 2001), by Edward Tufte, is the classic work on visualizing data.

BLOGS

- *Information Is Beautiful* (www.informationisbeautiful.net), by David McCandless, distills the world's data into beautiful and useful infographics and diagrams.
- *FlowingData* (http://flowingdata.com), by Nathan Yau, explores how statisticians and designers use visualization to understand data.
- *Visualising Data* (www.visualisingdata.com), by Andy Kirk, provides theoretical and practical insight about visualization with examples, references, and resources.

The information that's relevant changes with every audience and occasion. Sometimes it's budgets. Sometimes it's risk. Sometimes it's technical details. But it's never your modeling language's syntax. So, what's a savvy IT architect to do?

Bridging the Communication Gap

One common approach is to simply ignore the differences and use a language that we as architects are familiar with. We might draw UML on a whiteboard, make jokes in Klingon, and answer questions with "42." Implicitly, this means we expect the audience to learn our language, watch *Star Trek*, and read up on their Douglas Adams before speaking to us.

Another approach is to become a polyglot by learning all their languages. That takes time you might not have, and it's hard work. The question remains whether you'll be able to communicate your thoughts, analysis, and solutions in a language that's not grounded in IT. You'll probably feel uncomfortable, not the trusted advisor you'd like to be.

A third way is to create a common language together with your stakeholders that you use to communicate about IT architecture and their concerns. If the topic is important and sufficiently complex, this is your best option. This doesn't mean you invent a completely new language that covers all business concerns from scratch. Rather, borrow from languages you all already speak to cocreate a language that suits your and your stakeholders' specific concerns.

This is best done using an incremental approach in which you develop visualizations and their language in parallel. Typically, we use a series of workshops to show the visualization in intermediate stages and get feedback on its correctness, its effectiveness in making the architecture understandable, and whether it shows the right information at the right level. We bring large printouts to these workshops, distribute pens and markers, and stimulate the audience to enhance the visualization according to their insights and wishes.

One advantage of this approach is that the resulting language is suited to the audience. Not everyone who makes financial decisions has an economics degree. In a technology company such as Philips or Siemens, many engineers and physicists hold senior management positions; they'll bring metaphors, analogies, and metrics from their domain to the table.

Between workshops, we use the feedback to create a better version of the architecture description: clearer, more complete, and more effective.

Example: Information Flowing through a Landscape of Applications

Figure 1 shows an example of where this might lead to. You might wonder how this differs from other architectural models you've seen and created. This is still boxes and lines, right?

Actually, instead of being generic items, these boxes, lines, colors, and symbols are loaded with meaning for the intended audience. The people in this company know exactly why the large red box in the middle is large and red: this IT application is at the heart of their company, and all the money flows through it. That's also why it's in the middle. The amount of persistent data determines the application's size—that's why the box is so large.

Most organizations have similar connotations in their internal vocabulary. For example, in the Dutch Tax Administration, blue, red, and green symbolize distinct parts of the organization. Even the outbound envelopes have colors matching these business domains. It would be foolish not to take this aspect into account when creating a visual model for the organization. But you can also create such connotations while creating visualizations. (As a rule, though, critical information shouldn't depend on colors alone because color blindness is more common than you might think. For example, as many as 8 percent of men and 0.5 percent of women with Northern European ancestry have the common form of red-green color blindness.)

Typically, IT architects present images of the IT landscape as a collection of applications, connections, and databases. This is hardly interesting to business users. In financial terms, you're showing a company's assets and liabilities. This might be a necessity for the yearly report but doesn't really tell you how the stuff is being used, by whom, and when and where. For the business, these latter aspects are way more interesting than applications and technology.

One of our more successful visual patterns shows the actual use of an IT landscape by employing "metro" lines to illustrate flows of information or the execution of business processes. Anyone who has seen a metro or train map immediately interprets this correctly: the metro lines bring data or state from one place to another. A circle indicates a metro station: that's where the application is used to process information. This is another example of tapping into an existing language to retrieve elements that the audience already understands, instead of demanding them to learn a new syntax.

Example: Visualizing Quality Attributes Dynamically

In some internal projects, we're experimenting with letting business metrics determine an application's size in a visualization, discovering whether that helps effectively communicate the crux of the matter. Figure 2 shows groups of applications layered in blue, green, and orange. According to the "number of database records" metric, an application landscape might look like Figure 2a. The other landscape views differ according to some other metric—for example, each application's yearly maintenance budget (see Figure 2b). The applications' colors and relative position remain stable, but the applications' size tells a different story.

Some other metrics might reveal even more interesting views of that landscape: the number of users (see Figure 2c) or the number of availability incidents (see Figure 2d). This gives stakeholders a way to compare and reason about the landscape. Here are some examples:

- The application crucially positioned in the center of the landscape has the lowest availability. Why?
- The applications serving the most users don't receive proportional budgets. Is that aligned with our plans?
- The application managing the most data has the highest availability, but that plagued application in the center provides the only pathway to that data. How problematic is that?

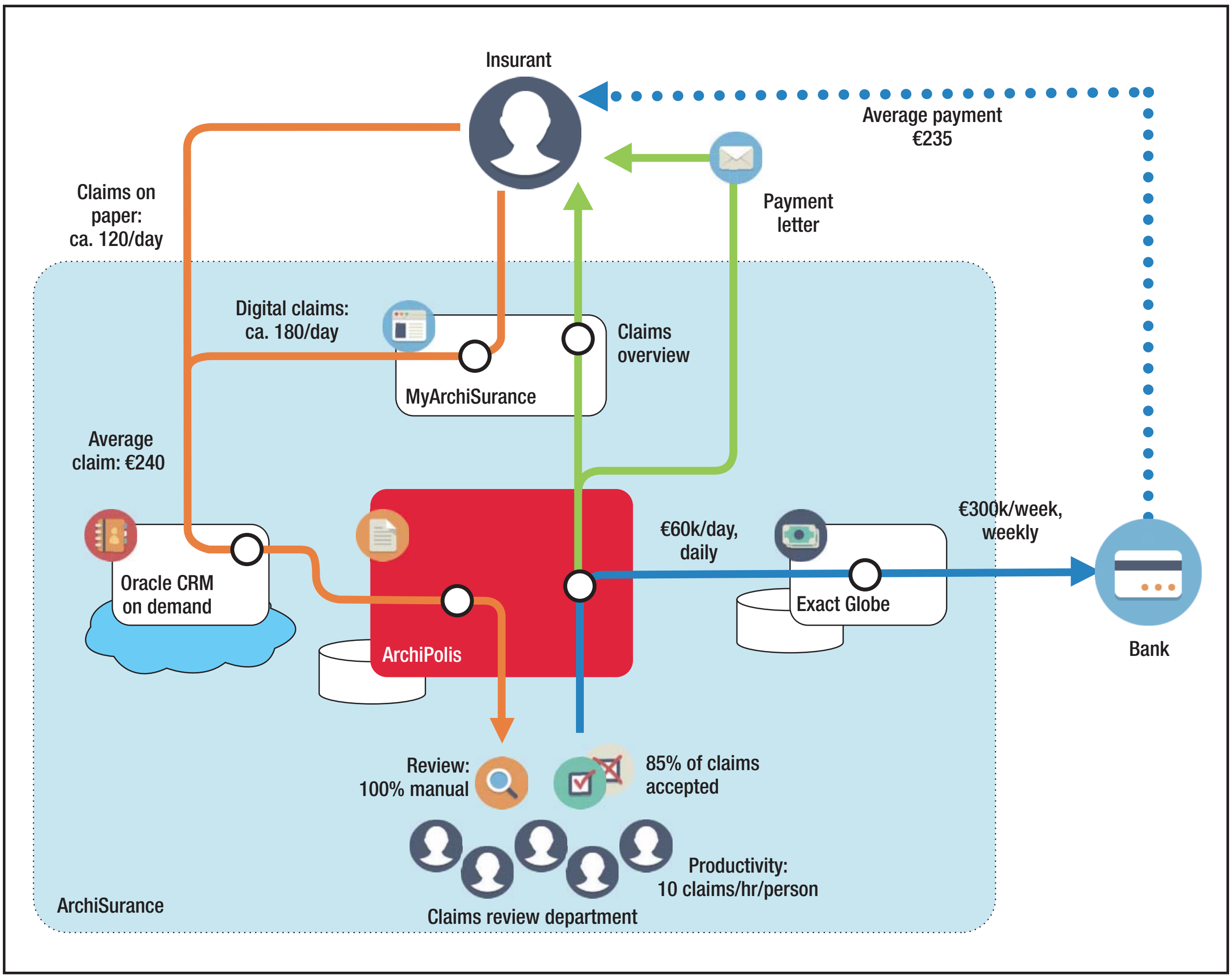

FIGURE 1. Part of the IT landscape of an (imagined) insurance company, inspired by the ArchiMate example at www.archimate.nl/en/about_archimate/example.html. Instead of being generic items, these boxes, lines, colors, and symbols are loaded with meaning for the intended audience.

In our experiments, the IT landscape on display isn't a static image. A UI lets stakeholders select different business-oriented metrics. In response, the landscape view dynamically changes as applications grow or shrink according to the selected metric. Of course, there are as many business metrics as there are decision makers. Some are universal; others are domain or even stakeholder specific. The examples shown here are intended to spark creativity. Every architect must find the right language for his or her stakeholders.

Example: Visualizing How IT Availability Impacts Business Processes

Figure 3 is adapted from our work for an application-hosting provider that manages mission-critical IT landscapes for businesses. Business and IT personnel share this view to answer, how do IT defects impact the business? The view shows the main business processes and the underlying IT applications but leaves out less relevant aspects such as integrations and application technology. The business processes drive the layout, showing the main processes as value streams through the organization. IT applications are ordered according to their place in these processes.

The hosting provider has a range of monitoring tools that gather real-time information on the applications'

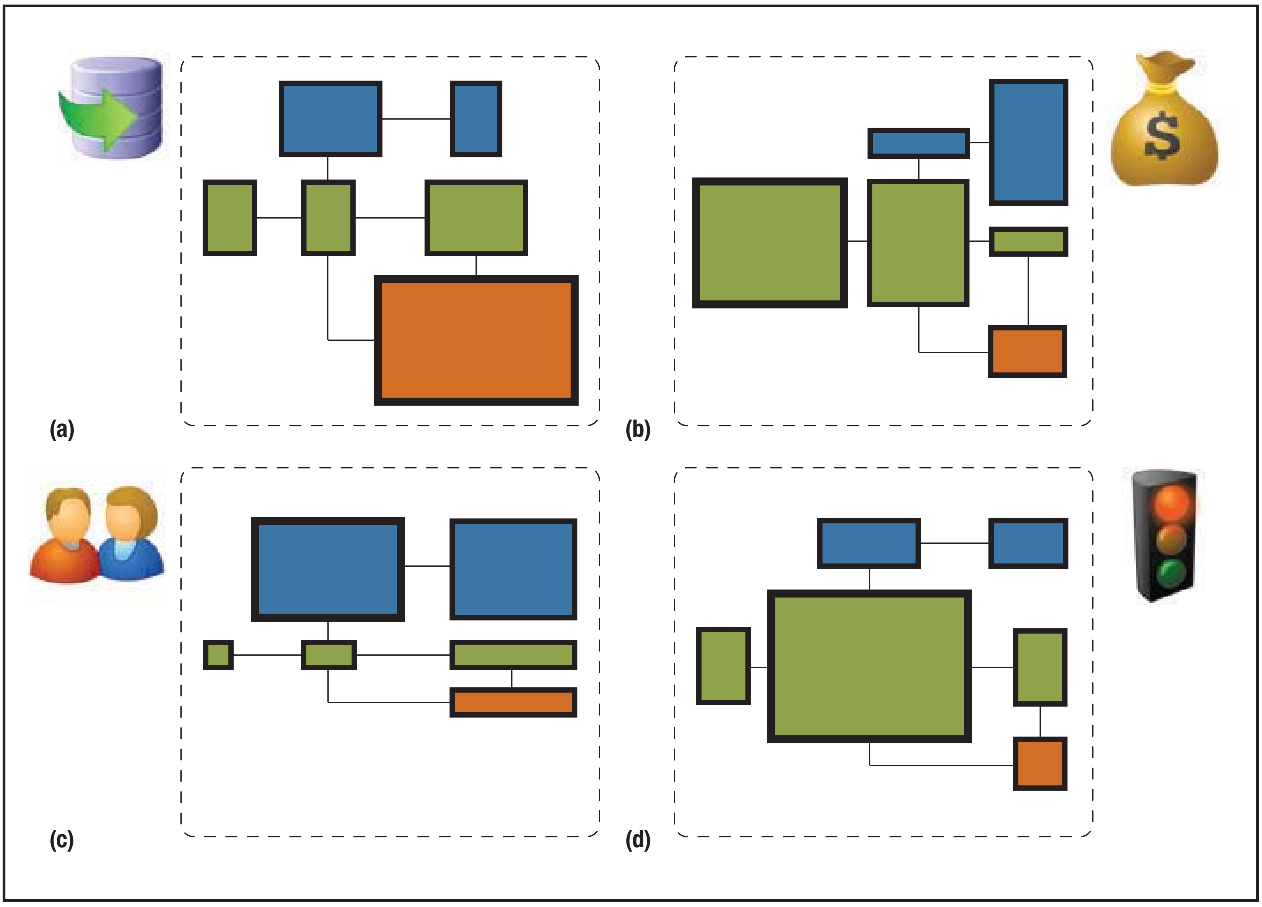

FIGURE 2. Business metrics determine the applications' size in an architecture visualization. (a) The number of database records. (b) The yearly maintenance budget. (c) The application's number of online users. (d) The number of availability issues per month.

performance and availability. Business process status—ranging from all okay, to hindered operations, to unavailable—is dynamically updated, on the basis of the underlying applications' availability.

This shared view means that problems are better understood and therefore less likely to cause stress. The IT department sees the impact of a failure and understands the need to solve problems quickly. Businesspeople see the colors of a process change before they notice any problems in their actual work. The knowledge that IT has the same view and is working on it builds trust.

Challenges with This Visualization Approach

This approach clearly has its drawbacks, especially from the traditional IT architect's perspective. The visualizations need hard work because you can't generate them from an architecture repository. They are incomplete and can be hard to maintain. Sometimes, the semantics are ambiguous, and they certainly don't adhere to any standard.

We accept these drawbacks because we get something in return that's much more valuable: visualizations that enable architects and the rest of the organization to communicate with each other about content, instead of having to explain or decipher the form. A shared language for describing the organization's IT enables both architects and businesspeople to reason about it from their own disciplines and backgrounds, and make decisions that affect the architecture. In hindsight, we corroborate that our examples fit Grady Booch's idea of "predicative and actionable [architecture] diagrams,"[2] although they're meant for not only colleague architects and developers but also business managers, end users, and other nonarchitects.

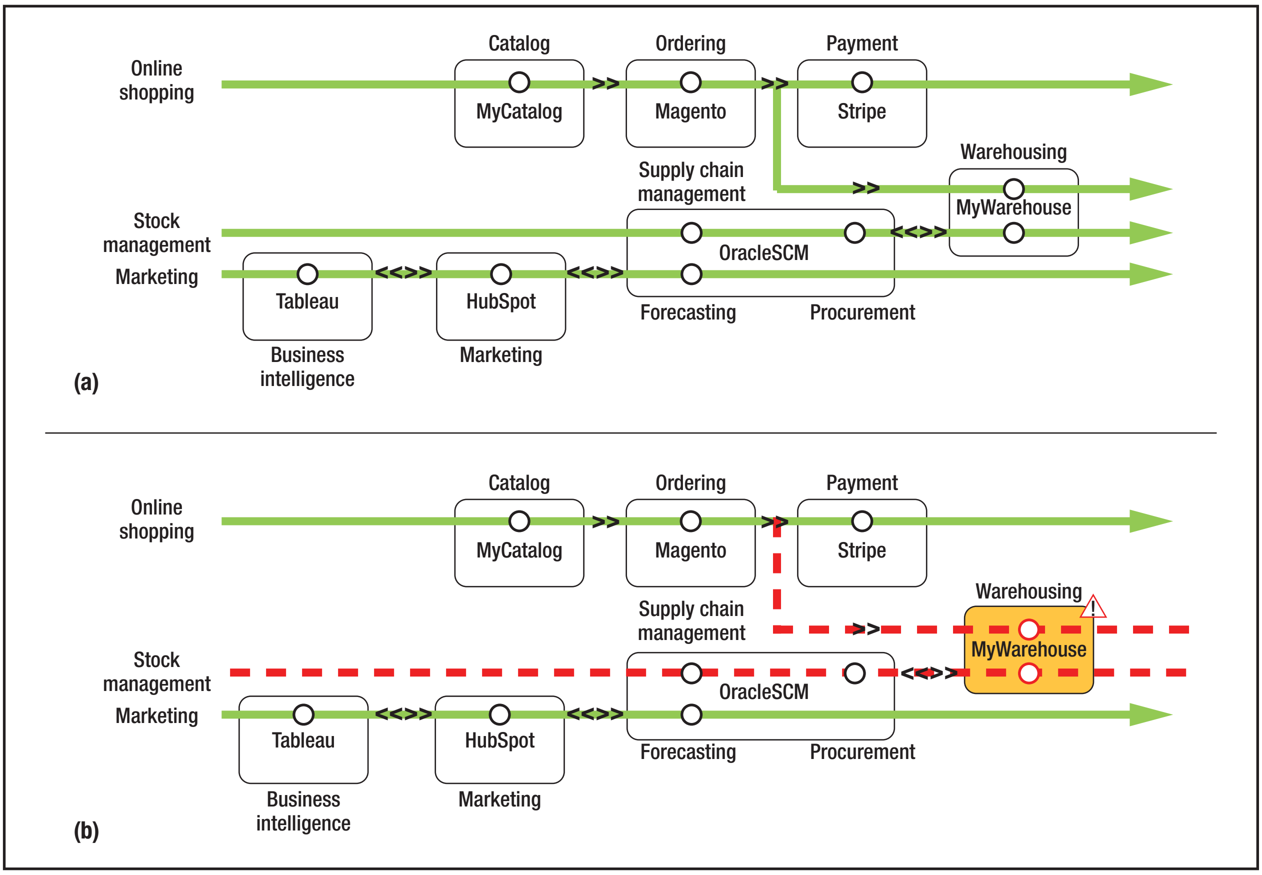

FIGURE 3. How IT system availability impacts a customer's business processes. (a) All is well. (b) The custom-built application MyWarehouse is unavailable, negatively affecting (some) business processes.

Further research and practice will increase our knowledge of how people with and without a technical background perceive and understand architecture visualizations. When we learn to speak the languages of our target audiences, they might, finally, get what we were trying to say all along. ⓢⓦ

References

1. G. Hohpe, "The Architect Elevator," 15 Jan. 2015; www.enterpriseintegrationpatterns.com/ramblings/79_elevator.html.
2. G. Booch, "Draw Me a Picture," *IEEE Software*, vol. 28, no. 1, 2011, pp. 6–7; doi:10.1109/MS.2011.4.

JOCHEM SCHULENKLOPPER is a software architect and consultant at Xebia. Contact him at jschulenklopper@xebia.com.

EELCO ROMMES is a senior advisor at M&I/Partners. Contact him at eelco.rommes@mxi.nl.