- **Embracing falling boundaries**—Whether you like it or not, boundaries are falling down around you. We choose to embrace this by examining concepts like perimeterless enterprise, development environments in the cloud, and co-location by telepresence.

- **Applying proven practices to areas that somehow missed them**—We are not really sure why, but many in our industry have missed ideas like capturing client side javaScript errors, continuous delivery for mobile, database migrations for NoSQL, and frameworks for CSS.

- **Lightweight options for analytics**—Data science and analytics are not just for people with a PhD in the field. We highlight collaborative analytics and data science, where all developers understand the basics and work closely with experts when necessary.

- **Infrastructure as code**—Continuous delivery and DevOps have elevated our thinking about infrastructure. The implications of thinking about infrastructure as code and the need for new tools are still evolving.

ThoughtWorkers are passionate about technology. We build it, research it, test it, open source it, write about it, and constantly aim to improve it – for everyone. Our mission is to champion software excellence and revolutionize IT. We create and share the ThoughtWorks Technology Radar in support of that mission. The ThoughtWorks Technology Advisory Board, a group of senior technology leaders in ThoughtWorks, creates the radar. They meet regularly to discuss the global technology strategy for ThoughtWorks and the technology trends that significantly impact our industry.

The radar captures the output of the Technology Advisory Board's discussions in a format that provides value to a wide range of stakeholders, from CIOs to developers. The content is intended as a concise summary. We encourage you to explore these technologies for more detail. The radar is graphical in nature, grouping items into techniques, tools, platforms, and languages & frameworks. When radar items could appear in multiple quadrants, we chose the one that seemed most appropriate. We further group these items in four rings to reflect our current position on them. The rings are:

- **Adopt:** We feel strongly that the industry should be adopting these items. We use them when appropriate on our projects.
- **Trial:** Worth pursuing. It is important to understand how to build up this capability. Enterprises should try this technology on a project that can handle the risk.
- **Assess:** Worth exploring with the goal of understanding how it will affect your enterprise.
- **Hold:** Proceed with caution.

Items that are new or have had significant changes since the last radar are represented as triangles, (▲) while items that have not moved are represented as circles (●). The detailed graphs for each quadrant show the movement that items have taken. We are interested in far more items than we can reasonably fit into a document this size, so we fade many items from the last radar to make room for the new items. Fading an item does not mean that we no longer care about it.

For more background on the radar, see http://martinfowler.com/articles/radar-faq.html

In previous radars we have talked about embedded servlet containers, and these are now widely adopted on our projects. Tools such as SimpleWeb and Webbit take the simple, embedded approach further and o f er raw HTTP server functionality without implementing the Java Servlet speci f cation. At the same time, Tomcat, the most popular Java application server, is increasingly used in embedded setups and Microsoft provides self-hosted servers for the .NET framework, lending further weight to this trend.

D3 continues to gain traction as a library for creating rich

Managing dependencies in distributed systems can become complicated, and is a problem more people are facing with the move to f ner-grained micro services. Hystrix is a library for