

# Bi-weekly Random Bits from the Internet

2015-11-08

(MCCALL MEAT)

Fail at Scale: Reliability in the Face of Rapid Change

P2, Ben Maurer, ACM Queue Volume 13, Issue 8

President Obama & Marilynne Robinson: A Conversation in  
Iowa

P13, President Barack Obama and Marilynne Robinson, The New York Review of Books, November 5 & 17, 2015 Issue

Multithreaded Toolkits: A Failed Dream?

P32, Graham Hamilton, Java.net, October 19, 2004

# Fail at Scale: Reliability in the Face of Rapid Change

Ben Maurer, *ACM Queue* Volume 13, Issue 8

Failure is part of engineering any large-scale system. One of Facebook's cultural values is embracing failure. This can be seen in the posters hung around the walls of our Menlo Park headquarters: "What Would You Do If You Weren't Afraid?" and "Fortune Favors the Bold."

To keep Facebook reliable in the face of rapid change we study common patterns in failures and build abstractions to address them. These abstractions ensure that best practices are applied across our entire infrastructure. To guide our work in building reliability abstractions we must understand our failures. We do this by building tools to diagnose issues and by creating a culture of reviewing incidents in a way that pushes us to make improvements that prevent future failures.

## Why Do Failures Happen?

While every failure has a unique story, many failures boil down to a small number of fundamental root causes.

### Individual machine failures

Often an individual machine will run into an isolated failure that doesn't affect the rest of the infrastructure. For example, maybe a machine's hard drive has failed, or a service on a particular machine has experienced a bug in code, such as memory corruption or a deadlock.

The key to avoiding individual machine failure is automation. Automation works best by combining known failure patterns (such as a hard drive with S.M.A.R.T. errors) with a search for symptoms of an unknown problem (for example, by swapping out servers with unusually slow response times). When automation finds symptoms of an unknown problem, manual investigation can help develop better tools to detect and fix future problems.

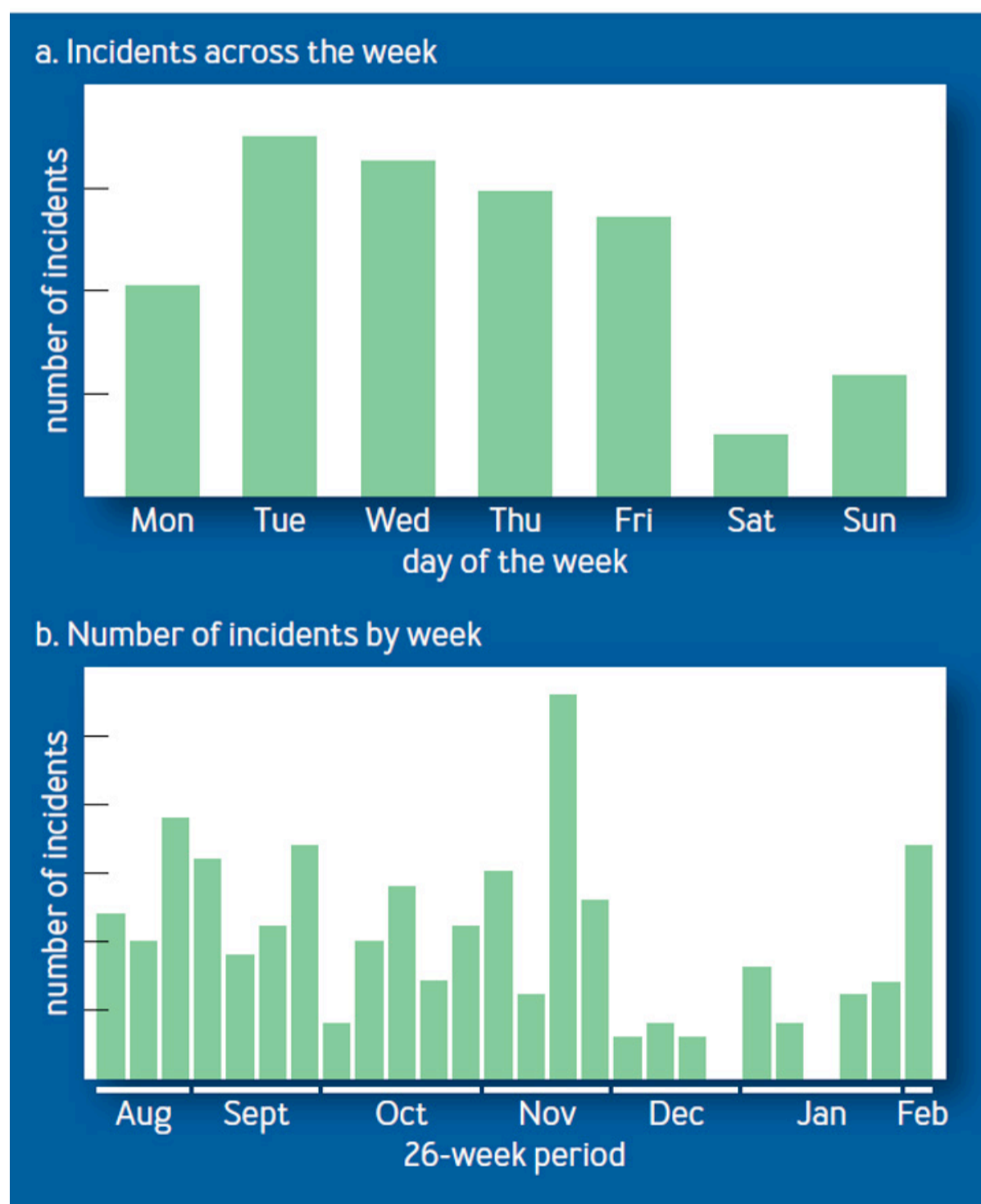
### Legitimate workload changes

Sometimes Facebook users change their behavior in a way that poses challenges for our infrastructure. During major world events, for example, unique types of work-

loads may stress our infrastructure in unusual ways. When Barack Obama won the 2008 U.S. Presidential election, his Facebook page experienced record levels of activity. Climactic plays in major sporting events such as the Super Bowl or World Cup result in an extremely high number of posts. Load testing, including “dark launches” where a feature is activated but not visible to the user, helps ensure that new features are able to handle load.

Statistics gathered during such events often provide a unique perspective on a system’s design. Oftentimes, major events cause changes in user behavior (for example, by creating focused activity around a particular object). Data about these changes often points to design decisions that will allow smoother operation in subsequent events.

FIGURE 1: SLA VIOLATION EVENTS



Human error

Given that Facebook encourages engineers to “Move Fast and Break Things”—an-

other one of the posters that adorns the offices—one might expect that many errors are caused by humans. Our data suggests that human error is a factor in our failures. Figure 1 includes data from an analysis of the timing of events severe enough to be considered an SLA (service-level agreement) violation. Each violation indicates an instance where our internal reliability goals were not met and caused an alert to be generated. Because our goals are strict most of these incidents are minor and not noticeable to users of the site. Figure 1a shows how incidents happened substantially less on Saturday and Sunday even though traffic to the site remains consistent throughout the week. Figure 1b shows a six-month period during which there were only two weeks with no incidents: the week of Christmas and the week when employees are expected to write peer reviews for each other.

These two data points seem to suggest that when Facebook employees are not actively making changes to infrastructure because they are busy with other things (weekends, holidays, or even performance reviews), the site experiences higher levels of reliability. We believe this is not a result of carelessness on the part of people making changes but rather evidence that our infrastructure is largely self-healing in the face of non-human causes of errors such as machine failure.

## Three Easy Ways to Cause an Incident

While failures have different root causes, we have found three common pathologies that amplify failures and cause them to become widespread. For each pathology, we have developed preventative measures that mitigate widespread failure.

### Rapidly deployed configuration changes

Configuration systems tend to be designed to replicate changes quickly on a global scale. Rapid configuration change is a powerful tool that can let engineers quickly manage the launch of new products or adjust settings. However, rapid configuration change means rapid failure when bad configurations are deployed. We use a number of practices to prevent configuration changes from causing failure.

- Make everybody use a common configuration system. Using a common configuration system ensures that procedures and tools apply to all types of configuration. At Facebook we have found that teams are sometimes tempted to handle configurations in a one-off way. Avoiding these temptations and managing configurations in a unified way has made the configuration system a leveraged way to make the site more reliable.

- Statically validate configuration changes. Many configuration systems allow

loosely typed configuration, such as JSON structures. These types of configurations make it easy for an engineer to mistype the name of a field, use a string where an integer was required, or make other simple errors. These kinds of straightforward errors are best caught using static validation. A structured format (for example, at Facebook we use Thrift)<sup>4</sup> can provide the most basic validation. It is not unreasonable, however, to write programmatic validation to validate more detailed requirements.

- Run a canary. First deploying your configuration to a small scope of your service can prevent a change from being disastrous. A canary can take multiple forms. The most obvious is an A/B test, such as launching a new configuration to only 1 percent of users. Multiple A/B tests can be run concurrently, and you can use data over time to track metrics.

For reliability purposes, however, A/B tests do not satisfy all of our needs. A change that is deployed to a small number of users, but causes implicated servers to crash or run out of memory, will obviously create impact that goes beyond the limited users in the test. A/B tests are also time consuming. Engineers often wish to push out minor changes without the use of an A/B test. For this reason, Facebook infrastructure automatically tests out new configurations on a small set of servers. For example, if we wish to deploy a new A/B test to 1 percent of users, we will first deploy the test to 1 percent of the users that hit a small number of servers. We monitor these servers for a short amount of time to ensure that they do not crash or have other highly visible problems. This mechanism provides a basic “sanity check” on all changes to ensure that they do not cause widespread failure.

- Hold on to good configurations. Facebook’s configuration system is designed to retain good configurations in the face of failures when updating those configurations. Developers tend naturally to create configuration systems that will crash when they receive updated configurations that are invalid. We prefer systems that retain old configurations in these types of situations and raise alerts to the system operator that the configuration failed to update. Running with a stale configuration is generally preferable to returning errors to users.

- Make it easy to revert. Sometimes, despite all best efforts, a bad configuration is deployed. Quickly finding and reverting the change is key to resolving this type of issue. Our configuration system is backed by version control, making it easy to revert changes.

## Hard dependencies on core services

Developers tend to assume that core services—such as configuration management, service discovery, or storage systems—never fail. Even brief failures in these core services, however, can turn into large-scale incidents.

- Cache data from core services. Hard dependencies on these types of services are often not necessary. The data these services return can be cached in a way that allows for the majority of services to continue operating during a brief outage of one of these systems.
- Provide hardened APIs to use core services. Core services are best complemented by common libraries that follow best practices when using these core services. For example, the libraries might provide good APIs for managing the cache or good failure handling.
- Run fire drills. You might think you are able to survive an outage of a core service, but you never know until you try. For these types of outages we have had to develop systems for fire drills ranging from fault injection applied to a single server to manually triggered outages of entire data centers.

## Increased latency and resource exhaustion

Some failures result in services having increased latency to clients. This increase in latency could be small (for example, think of a human configuration error that results in increased CPU usage that is still within the service's capacity), or it could be nearly infinite (a service where the threads serving responses have deadlocked). While small amounts of additional latency can be easily handled by Facebook's infrastructure, large amounts of latency lead to cascading failures. Almost all services have a limit to the number of outstanding requests. This limit could be due to a limited number of threads in a thread-per-request service, or it could be due to limited memory in an event-based service. If a service experiences large amounts of extra latency, then the services that call it will exhaust their resources. This failure can be propagated through many layers of services, causing widespread failure.

Resource exhaustion is a particularly damaging mode of failure because it allows the failure of a service used by a subset of requests to cause the failure of all requests. For example, imagine that a service calls a new experimental service that is only launched to 1% of users. Normally requests to this experimental service take 1 millisecond, but due to a failure in the new service the requests take 1 second. Requests for the 1% of users using this new service might consume so many threads that requests for the other 99% of users are unable to run.

We have found a number of techniques that can avoid this type of buildup with a low false positive rate.

- **Controlled Delay.** In analyzing past incidents involving latency, we found that many of our worst incidents involved large numbers of requests sitting in queues awaiting processing. The services in question had a resource limit (such as a number of active threads or memory) and would buffer requests in order to keep usage below the limit. Because the services were unable to keep up with the rate of incoming requests, the queue would grow larger and larger until it hit an application-defined limit. To address this situation, we wanted to limit the size of the queue without impacting reliability during normal operations. We studied the research on bufferbloat as our problems seemed similar—the need to queue for reliability without causing excessive latency during congestion. We experimented with a variant of the CoDel (controlled delay) algorithm:

```
onNewRequest(req, queue):  
  
    if (queue.lastEmptyTime() < (now - N seconds)) {  
        timeout = M ms  
    } else {  
        timeout = N seconds;  
    }  
    queue.enqueue(req, timeout)
```

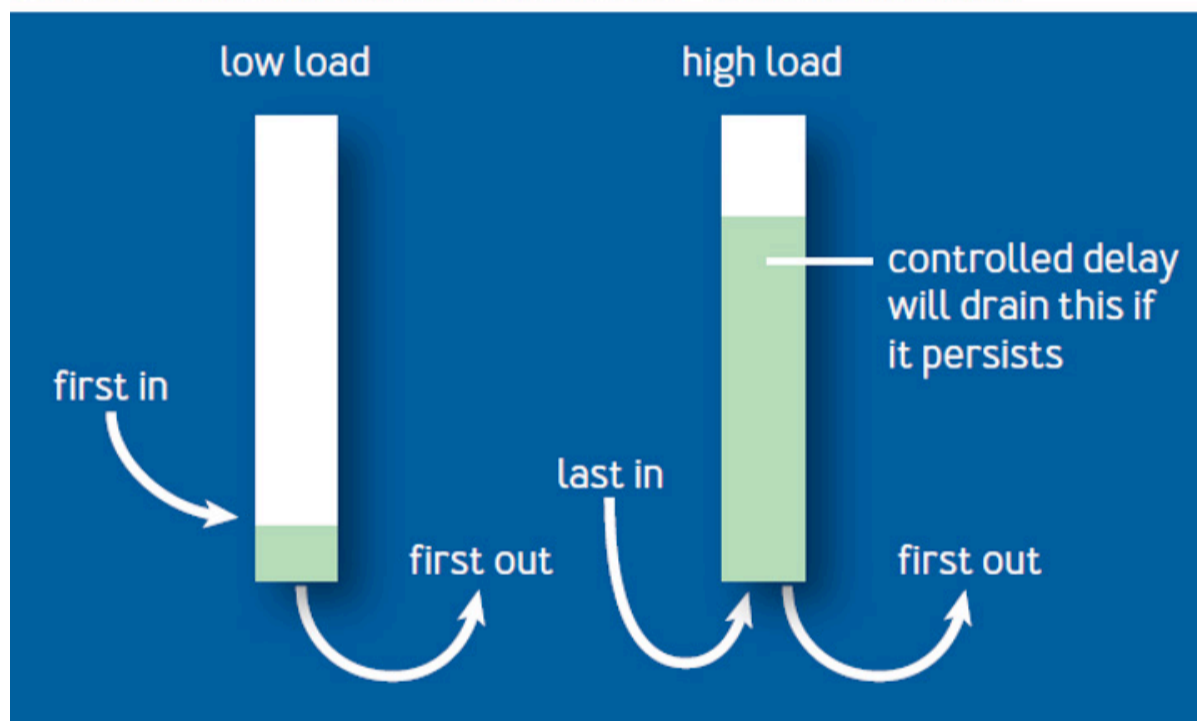
In this algorithm, if the queue has not been empty for the last  $N$  milliseconds, then the amount of time spent in the queue is limited to  $M$  milliseconds. If the service has been able to empty the queue within the last  $N$  milliseconds, then the time spent in the queue is limited to  $N$  milliseconds. This algorithm prevents a standing queue (because the `lastEmptyTime` will be in the distant past, causing an  $M$ -ms queuing timeout) while allowing short bursts of queuing for reliability purposes. While it might seem counterintuitive to have requests with such short timeouts, this process allows requests to be quickly discarded rather than build up when the system is not able to keep up with the rate of incoming requests. A short timeout ensures that the server always accepts just a little bit more work than it can actually handle so it never goes idle.

An attractive property of this algorithm is that the values of  $M$  and  $N$  tend not to need tuning. Other methods of solving the problem of standing queues, such as setting a limit on the number of items in the queue or setting a timeout for the queue, have required tuning on a per-service basis. We have found that a value of 5 milliseconds for  $M$  and 100 ms for  $N$  tends to work well across a wide set of use cases.

Facebook's open source Wangle library provides an implementation of this algorithm which is used by our Thrift framework.

- Adaptive LIFO (last-in, first-out). Most services process queues in FIFO (first-in first-out) order. During periods of high queuing, however, the first-in request has often been sitting around for so long that the user may have aborted the action that generated the request. Processing the first-in request first expends resources on a request that is less likely to benefit a user than a request that has just arrived. Our services process requests using adaptive LIFO. During normal operating conditions, requests are processed in FIFO order, but when a queue is starting to form, the server switches to LIFO mode. Adaptive LIFO and CoDel play nicely together, as shown in figure 2. CoDel sets short timeouts, preventing long queues from building up, and adaptive LIFO places new requests at the front of the queue, maximizing the chance that they will meet the deadline set by CoDel. HHVM, Facebook's PHP runtime, includes an implementation of the Adaptive LIFO algorithm.

FIGURE 2: LIFO (LEFT) AND ADAPTIVE LIFO WITH CODEL (RIGHT)



- Concurrency Control. Both CoDel and adaptive LIFO operate on the server side. The server is often the best place to implement latency-preventing measures—a server tends to serve a large number of clients and often has more information than its clients possess. Some failures are so severe, however, that server-side controls are not able to kick in. For this reason, we have implemented a stopgap measure in clients. Each client keeps track of the number of outstanding outbound requests on a per-service basis. When new requests are sent, if the number of outstanding requests to that service exceeds a configurable number, the request is immediately marked as an error. This mechanism prevents a single service from monopolizing all its client's resources.

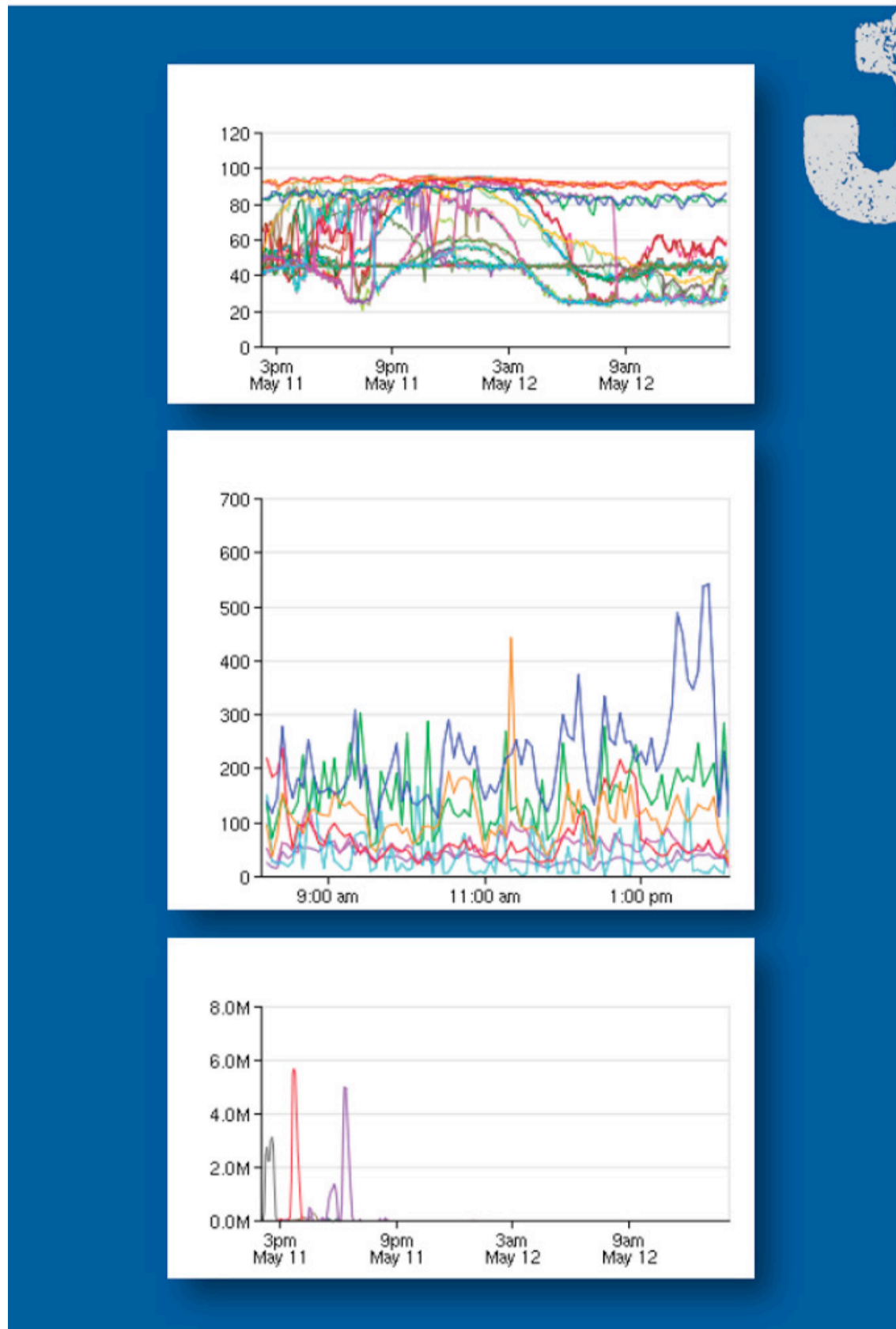
## Tools that Help Diagnose Failures



Despite the best preventative measures, some failures will always occur. During outages the right tools can quickly lead to the root cause, minimizing the duration of the failure.

## High-Density Dashboards with Cubism

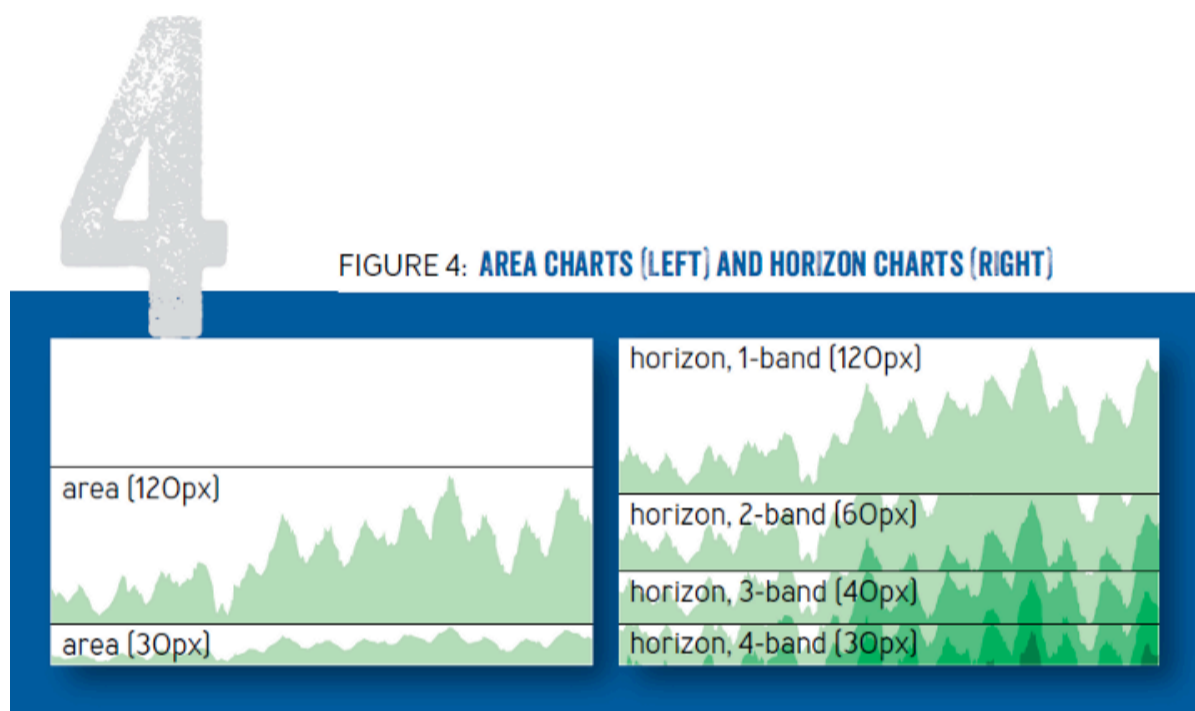
FIGURE 3: DIFFICULT-TO-READ CHARTS FROM A TYPICAL DASHBOARD



When handling an incident, it is important to have quick access to information. Good dashboards allow engineers quickly to assess the types of metrics that might be abnormal and then use this information to hypothesize a root cause. We found, however, that our dashboards grew so large that it was difficult to navigate them

quickly, and that charts shown on those dashboards had too many lines to read at a glance, as in figure 3.

To address this, we built our top-level dashboards using Cubism, a framework for creating horizon charts—line charts that use color to encode information more densely, allowing for easy comparison of multiple similar data series. For example, we use Cubism to compare metrics between different data centers. Our tooling around Cubism allows for easy keyboard navigation so engineers can view multiple metrics quickly. Figure 4 shows the same data set at various heights using area charts and horizon charts. In the area chart version, the 30-pixel version is hard to read. On the other hand, the horizon chart makes it extremely easy to find the peak value, even at a height of 30 pixels.



What just changed?

Since one of the top causes of failure is human error, one of the most effective ways of debugging failures is to look for what humans have changed recently. We collect information about recent changes ranging from configuration changes to deployments of new software in a tool called OpsStream. However, we have found over time that this data source has become extremely noisy. With thousands of engineers making changes, there are often too many to evaluate during an incident.

To solve this problem, our tools attempt to correlate failures with relevant changes. For example, when an exception is thrown, in addition to outputting the stack trace, we output any configuration settings read by that request that have had their values changed recently. Often, the cause of an issue that generates many stack traces is one of these configuration values. We can then quickly respond to the issue—for example, by reverting the configuration and involving the engineer who made the change.

## Learning from Failure

After failures happen, our incident-review process helps us learn from these incidents.

The goal of the incident-review process is not to assign blame. Nobody has been fired because an incident he or she caused came under review. The goal of the review is to understand what happened, remediate situations that allowed the incident to happen, and put safety mechanisms in place to reduce the impact of future incidents.

### A methodology for reviewing incidents

Facebook has developed a methodology called DERP (for detection, escalation, remediation, and prevention) to aid in productive incident reviews.

- Detection. How was the issue detected—alarms, dashboards, user reports?
- Escalation. Did the right people get involved quickly? Could these people have been brought in via alarms rather than manually?
- Remediation. What steps were taken to fix the issue? Can these steps be automated?
- Prevention. What improvements could remove the risk of this type of failure happening again? How could you have failed gracefully, or failed faster to reduce the impact of this failure?

DERP helps analyze every step of the incident at hand. With the aid of this analysis, even if you cannot prevent this type of incident from happening again, you will at least be able to recover faster the next time.

### Move Fast by Breaking Fewer Things

A “move-fast” mentality does not have to be at odds with reliability. To make these philosophies compatible,

Facebook’s infrastructure provides safety valves: our configuration system protects against rapid deployment of bad configurations; our core services provide clients with hardened APIs to protect against failure; and our core libraries prevent re-

source exhaustion in the face of latency. To deal with the inevitable issues that slip through the cracks, we build easy-to-use dashboards and tools to help find recent changes that might cause the issues under investigation. Most importantly, after an incident we use lessons learned to make our infrastructure more reliable.

# President Obama & Marilynne Robinson: A Conversation in Iowa

President Barack Obama and Marilynne Robinson, *The New York Review of Books*, November 5 & 17, 2015 Issue



The President: Marilynne, it's wonderful to see you. And as I said as we were driving over here, this is an experiment, because typically when I come to a place like Des Moines, I immediately am rushed over to some political event and I make a speech, or I have a town hall, or I go see some factory and have wonderful conversations with people. But it's very planned out and scripted. And typically, we're trying to drive a very particular message that day about education or about manufacturing.

But one of the things that I don't get a chance to do as often as I'd like is just to have a conversation with somebody who I enjoy and I'm interested in; to hear from them and have a conversation with them about some of the broader cultural forces that shape our democracy and shape our ideas, and shape how we feel about citizenship and the direction that the country should be going in.

And so we had this idea that why don't I just have a conversation with somebody I really like and see how it turns out. And you were first in the queue, because—

Marilynne Robinson: Thank you very much.

The President: Well, as you know—I've told you this—I love your books. Some listeners may not have read your work before, which is good, because hopefully they'll go out and buy your books after this conversation.

I first picked up *Gilead*, one of your most wonderful books, here in Iowa. Because I was campaigning at the time, and there's a lot of downtime when you're driving between towns and when you get home late from campaigning. And you and I, therefore, have an Iowa connection, because *Gilead* is actually set here in Iowa.

And I've told you this—one of my favorite characters in fiction is a pastor in *Gilead*, Iowa, named John Ames, who is gracious and courtly and a little bit confused about how to reconcile his faith with all the various travails that his family goes through. And I was just—I just fell in love with the character, fell in love with the book, and then you and I had a chance to meet when you got a fancy award at the White House. And then we had dinner and our conversations continued ever since.

So anyway, that's enough context. You just have completed a series of essays that are not fiction, and I had a chance to read one of them about fear and the role that fear may be playing in our politics and our democracy and our culture.\* And you looked at it through the prism of Christianity and sort of the Protestant traditions that helped shape us, so I thought maybe that would be a good place to start.

Why did you decide to write this book of essays? And why was fear an important topic, and how does it connect to some of the other work that you've been doing?

Robinson: Well, the essays are actually lectures. I give lectures at a fair rate, and then when I've given enough of them to make a book, I make a book.

The President: So you just kind of mash them all together?

Robinson: I do. That's what I do. But it rationalizes my lecturing, too. But fear was very much—is on my mind, because I think that the basis of democracy is the willingness to assume well about other people.

You have to assume that basically people want to do the right thing. I think that you can look around society and see that basically people do the right thing. But when people begin to make these conspiracy theories and so on, that make it seem as if what is apparently good is in fact sinister, they never accept the argument that is

made for a position that they don't agree with—you know?

The President: Yes.

Robinson: Because [of] the idea of the “sinister other.” And I mean, that's bad under all circumstances. But when it's brought home, when it becomes part of our own political conversation about ourselves, I think that that really is about as dangerous a development as there could be in terms of whether we continue to be a democracy.

The President: Well, now there's been that strain in our democracy and in American politics for a long time. And it pops up every so often. I think the argument right now would be that because people are feeling the stresses of globalization and rapid change, and we went through one of the worst financial crises since the Great Depression, and the political system seems gridlocked, that people may be particularly receptive to that brand of politics.

Robinson: But having looked at one another with optimism and tried to facilitate education and all these other things—which we've done more than most countries have done, given all our faults—that's what made it a viable democracy. And I think that we have created this incredibly inappropriate sort of in-group mentality when we really are from every end of the earth, just dealing with each other in good faith. And that's just a terrible darkening of the national outlook, I think.

The President: We've talked about this, though. I'm always trying to push a little more optimism. Sometimes you get—I think you get discouraged by it, and I tell you, well, we go through these moments.

Robinson: But when you say that to me, I say to you, you're a better person than I am.

The President: Well, but I want to pick up on the point you made about us coming from everywhere. You're a novelist but you're also—can I call you a theologian? Does that sound, like, too stuffy? You care a lot about Christian thought.

Robinson: I do, indeed.

The President: And that's part of the foundation of your writings, fiction and non-fiction. And one of the points that you've made in one of your most recent essays is that there was a time in which at least reformed Christianity in Europe was very much “the other.” And part of our system of government was based on us rejecting an exclusive, inclusive—or an exclusive and tightly controlled sense of who is part

of the community and who is not, in favor of a more expansive one.

Tell me a little bit about how your interest in Christianity converges with your concerns about democracy.

Robinson: Well, I believe that people are images of God. There's no alternative that is theologically respectable to treating people in terms of that understanding. What can I say? It seems to me as if democracy is the logical, the inevitable consequence of this kind of religious humanism at its highest level. And it [applies] to everyone. It's the human image. It's not any loyalty or tradition or anything else; it's being human that enlists the respect, the love of God being implied in it.

The President: But you've struggled with the fact that here in the United States, sometimes Christian interpretation seems to posit an "us versus them," and those are sometimes the loudest voices. But sometimes I think you also get frustrated with kind of the wishy-washy, more liberal versions where anything goes.

Robinson: Yes.

The President: How do you reconcile the idea of faith being really important to you and you caring a lot about taking faith seriously with the fact that, at least in our democracy and our civic discourse, it seems as if folks who take religion the most seriously sometimes are also those who are suspicious of those not like them?

Robinson: Well, I don't know how seriously they do take their Christianity, because if you take something seriously, you're ready to encounter difficulty, run the risk, whatever. I mean, when people are turning in on themselves—and God knows, arming themselves and so on—against the imagined other, they're not taking their Christianity seriously. I don't know—I mean, this has happened over and over again in the history of Christianity, there's no question about that, or other religions, as we know.

But Christianity is profoundly counterintuitive—"Love thy neighbor as thyself"—which I think properly understood means your neighbor is as worthy of love as you are, not that you're actually going to be capable of this sort of superhuman feat. But you're supposed to run against the grain. It's supposed to be difficult. It's supposed to be a challenge.

The President: Well, that's one of the things I love about your characters in your novels, it's not as if it's easy for them to be good Christians, right?



Robinson: Right.

The President: It's hard. And it's supposed to be hard. Now, you grew up in Idaho, in a pretty—it wasn't a big, cosmopolitan place.

Robinson: The word "cosmopolitan" was never applied.

The President: Which town in Idaho did you grow up in?

Robinson: [Coeur d'Alene] is where I really grew up.

The President: How big was the town when you were growing up?

Robinson: 13,500 people.

The President: All right. So that's a town.

Robinson: Yes, the second-largest city in the state at the time.

The President: And how do you think you ended up thinking about democracy, writing, faith the way you do? How did that experience of growing up in a pretty small place in Idaho, which might have led you in an entirely different direction—how did you end up here, Marilynne? What happened? Was it libraries?

Robinson: It was libraries, it was—people are so complicated. It's like every new person is a completely new roll of the dice, right?

The President: Right.

Robinson: I followed what was for me the path of least resistance, which meant reading a lot of books and writing, because it came naturally to me. My brother is excellent in many of these things, you know? And I think we reinforced each other, he and I, but it was perfectly accidental.

With all respect to that environment, many very smart people do not follow the path in life that people like my brother and I did. You learn from them even if you don't learn from them in a formal sense. But I always knew what I wanted to do in a sense—I mean, not be, but do. I didn't really have the concept of author until I was in high school. But I was writing.

The President: But you knew you wanted to read and write.

Robinson: Yes, that's what I wanted to do.

The President: Were your parents into books, or did they just kind of encourage you or tolerate your quirkiness?

Robinson: There was great tolerance in the house for quirkiness. No, it's a funny thing because on the one hand, I'm absolutely indebted to my origins, whatever they are, whatever that means. On the other hand, with all love and respect, my parents were not particularly bookish people.

The President: Well, that's why you have good sense along with sort of an overlay of books on top of good sense. What did your mom and dad do?

Robinson: My mother was a stay-at-home mother. My father was a sort of middle-management lumber company guy.

The President: But they encouraged it.

Robinson: You know what, they were the adults and we were the kids, you know what I mean? Sort of like two species. But if they noticed we were doing something—drawing or painting or whatever we were doing—then they would get us what we needed to do that, and silently go on with it. One of the things that I think is very liberating is that if I had lived any honest life, my parents would have been equally happy. I was under no pressure.

The President: Well, you told me about a certain attitude that your parents had that was—there was a certain set of homespun values of hard work and honesty and humility. And that sounded really familiar to me when I think about my grandparents who grew up in Kansas.

And that's part of what I see in your writing. And part of my connection to your books, I think, is an appreciation for—without romanticizing Middle America or small-town America—that sense of homespun virtues. And that comes out in your writing. And it sometimes seems really foreign to popular culture today, which is all about celebrity and being loud and bragging and—

Robinson: I mean, I really think that you have to go very far up in American culture to get beyond the point where people have good values. I mean, you really have that feeling sometimes that honesty is more intrinsic in some person that's doing very low-level work than it is in perhaps somebody that's trying to find his way into

some sensation—

The President: These big systems where everything is all about flash. But that's not how your parents saw the world, right? When you said that all they cared about was just you being honest and—

Robinson: Yes, exactly.

The President:—doing your best in some enterprise.

Robinson: In whatever. Exactly.

The President: It's interesting, because we're talking in Iowa; people always, I think, were surprised about me connecting with folks in small-town Iowa. And the reason I did was, first of all, I had the benefit that at the time nobody expected me to win. And so I wasn't viewed through this prism of Fox News and conservative media, and making me scary. At the time, I didn't seem scary, other than just having a funny name. I seemed young. Sometimes I look at my pictures from then and I say, I can't believe anybody voted for me because I look like I'm twenty-five.

But I'd go into these towns and everybody felt really familiar to me, because they reminded me of my grandparents and my mom and that attitude that you talk about. You saw all through the state—and I saw this when I was traveling through southern Illinois when I was first campaigning for the United States Senate—and I actually see it everywhere across the country.

The issue to me, Marilynne, is not so much that those virtues that you prize and that you care about and that are vital to our democracy aren't there. They are there in Little League games, and—

Robinson: Emergency rooms.

The President:—emergency rooms, and in school buildings. And people are treating each other the way you would want our democracy to cultivate. But there's this huge gap between how folks go about their daily lives and how we talk about our common life and our political life. And people describe it as the distance between Washington and Main Street. But it's not just Washington; it's the way we talk about our politics, our foreign policy, our common endeavors. There's this gap.

And the thing I've been struggling with throughout my political career is how do you close the gap. There's all this goodness and decency and common sense on the

ground, and somehow it gets translated into rigid, dogmatic, often mean-spirited politics. And some of it has to do with all the filters that stand between ordinary people who are busy and running around trying to look after their kids and do a good job and do all the things that maintain a community, so they don't have the chance to follow the details of complicated policy debates.

They know they want to take care of somebody who's sick, and they have a generous impulse. How that gets translated into the latest Medicare budgets [isn't] always clear. They know they want us to use our power wisely in the world, and that violence often begets violence. But they also know the world is dangerous and it's very hard to sort out, as you talk about in your essay, fear when violence must be met, and when there are other tools at our disposal to try to create a more peaceful world.

So that, I think, is the challenge. I'm very encouraged when I meet people in their environments. Somehow it gets distilled at the national political level in ways that aren't always as encouraging.

Robinson: I think one of the things that is true is that many Americans on every side of every issue, they think that the worst thing they can say is the truest thing, you know?

The President: No. Tell me what you mean.

Robinson: Well, for example—I mean, I'm a great admirer of American education. And I've traveled—I mean, a lot of my essays, you know, are lectures given in educational settings—universities everywhere. And they're very impressive. They are very much loved by people who identify with them. You meet faculty and they're very excited about what they're doing; students that are very excited, and so on.

And then you step away and you hear all this stuff about how the system is failing and we have to pull it limb from limb, and the rest of it. And you think, have you walked through the door? Have you listened to what people say? Have you taught in a foreign university?

We have a great educational system that is—it's really a triumph of the civilization. I don't think there's anything comparable in history. And it has no defenders. Most of the things we do have no defenders because people tend to feel the worst thing you can say is the truest thing you can say.

The President: But that's part of what makes America wonderful, is we always had

this nagging dissatisfaction that spurs us on. That's how we ended up going west, that's how we—"I'm tired of all these people back east; if I go west, there's going to be my own land and I'm not going to have to put up with this nonsense, and I'm going to start my own thing, and I've got my homestead." ...It is true, though, that that restlessness and that dissatisfaction which has helped us go to the moon and create the Internet and build the Transcontinental Railroad and build our land-grant colleges, that those things, born of dissatisfaction, we can very rapidly then take for granted and not tend to and not defend, and not understand how precious these things are.

And this is where conceptions of government can get us in trouble. Whenever I hear people saying that our problems would be solved without government, I always want to tell them you need to go to some other countries where there really is no government, where the roads are never repaired, where nobody has facilitated electricity going everywhere even where it's not economical, where—

Robinson: The postal system.

The President:—the postal system doesn't work, or kids don't have access to basic primary education. That's the logical conclusion if, in fact, you think that government is the enemy.

And that, too, is a running strain in our democracy. That's sort of in our DNA. We're suspicious of government as a tool of oppression. And that skepticism is healthy, but it can also be paralyzing when we're trying to do big things together.

Robinson: And also, one of the things that doesn't take into account is that local governments can be great systems of oppression. And it's a wonderful thing to have a national government that can intervene in the name of national values.

The President: Well, that was the lesson of the entire movement to abolish slavery and the civil rights movement. And that's one thing—I mean, I do think that one of the things we haven't talked about that does become the fault line around which the "us" and "them" formula rears its head is the fault line of race. And even on something like schools that you just discussed, part of the challenge is that the school systems we have are wonderful, except for a handful of schools that are predominantly minority that are terrible.

Our systems for maintaining the peace and our criminal justice systems generally work, except for this huge swath of the population that is incarcerated at rates that are unprecedented in world history.

And when you are thinking about American democracy or, for that matter, Christianity in your writings, how much does that issue of “the other” come up and how do you think about that? I know at least in Gilead that factors into one major character, trying to figure out how he can love somebody in the Fifties that doesn’t look like him.

Robinson: Iowa never had laws against interracial marriage. Only Iowa and Maine never had [them]—

The President: Those were the only two.

Robinson: Yes. And [Ulysses S.] Grant really did call [Iowa] the shining star of radicalism, and so on. We never had segregated schools; they were illegal from before, while it was still a territory, and so on. And these laws never changed and they became the basis for the marriage equality ruling that the Supreme Court here [in Iowa] did.

So that whole stream of the culture never changed. And at the same time, the felt experience of the culture was not aligned with the liberal tradition [of the] culture. And so in that book, Jack has every right to think he can come to Iowa, and yet what he finds makes him frightened even to raise the question.

The President: I’m going to shift gears for a second. You told me that when you started writing it just kind of showed up in some ways. When you started writing your novels, that it was just forced upon you and that you didn’t map it out. Tell me about when you were writing Gilead and Home and some of my favorite books, how did you decide, I’m going to start writing about some old pastor in the middle of cornfields?

Because by that time you had gone to the East Coast, you had traveled in France.

Robinson: The Midwest was still a very new thing for me. I got a voice in my head. It was the funniest thing. I mean, [I’d] been reading history and theology and all these things for a long time. And then I was in Massachusetts, actually, just [waiting to spend] Christmas with my son[s]. They were late coming to wherever we were going to meet, and I was in this hotel with a pen and blank paper, and I started writing from this voice. The first sentence in that book is the first sentence that came to my mind. I have no idea how that happens. I was surprised that I was writing from a male point of view. But there he was.

The President: He just showed up.

Robinson: He just showed up. And the first things that I knew about him—that he was old, that he had a young son, and so on—they create the narrative.



The President: Are you somebody who worries about people not reading novels anymore? And do you think that has an impact on the culture? When I think about how I understand my role as citizen, setting aside being president, and the most important set of understandings that I bring to that position of citizen, the most important stuff I've learned I think I've learned from novels. It has to do with empathy. It has to do with being comfortable with the notion that the world is complicated and full of grays, but there's still truth there to be found, and that you have to strive for that and work for that. And the notion that it's possible to connect with some[one] else even though they're very different from you.

And so I wonder when you're sitting there writing longhand in some—your messy longhand somewhere—so I wonder whether you feel as if that same shared culture is as prevalent and as important in the lives of people as it was, say, when you were that little girl in Idaho, coming up, or whether you feel as if those voices have been overwhelmed by flashier ways to pass the time.

Marilynne Robinson: I'm not really the person—because I'm almost always talking

with people who love books.

The President: Right. You sort of have a self-selecting crew.

Robinson: And also teaching writers—I'm quite aware of the publication of new writers. I think—I mean, the literature at present is full to bursting. No book can sell in that way that *Gone with the Wind* sold, or something like that. But the thing that's wonderful about it is that there's an incredible variety of voices in contemporary writing. You know people say, is there an American tradition surviving in literature, and yes, our tradition is the incredible variety of voices...

And [now] you don't get the conversation that would support the literary life. I think that's one of the things that has made book clubs so popular.

The President: That's interesting. Part of the challenge is—and I see this in our politics—is a common conversation. It's not so much, I think, that people don't read at all; it's that everybody is reading [in] their niche, and so often, at least in the media, they're reading stuff that reinforces their existing point of view. And so you don't have that phenomenon of here's a set of great books that everybody is familiar with and everybody is talking about.

Sometimes you get some TV shows that fill that void, but increasingly now, that's splintered, too, so other than the Super Bowl, we don't have a lot of common reference points. And you can argue that that's part of the reason why our politics has gotten so polarized, is that—when I was growing up, if the president spoke to the country, there were three stations and every city had its own newspaper and they were going to cover that story. And that would last for a couple of weeks, people talking about what the president had talked about.

Today, my poor press team, they're tweeting every two minutes because some new thing has happened, which then puts a premium on the sensational and the most outrageous or a conflict as a way of getting attention and breaking through the noise—which then creates, I believe, a pessimism about the country because all those quiet, sturdy voices that we were talking about at the beginning, they're not heard.

It's not interesting to hear a story about some good people in some quiet place that did something sensible and figured out how to get along.

Robinson: I think that in our earlier history—the Gettysburg Address or something—there was the conscious sense that democracy was an achievement. It was



not simply the most efficient modern system or something. It was something that people collectively made and they understood that they held it together by valuing it. I think that in earlier periods—which is not to say one we will never return to—the president himself was this sort of symbolic achievement of democracy. And there was the human respect that I was talking about before, [that] compounds itself in the respect for the personified achievement of a democratic culture. Which is a hard thing—not many people can pull that together, you know.... So I do think that one of the things that we have to realize and talk about is that we cannot take it for granted. It's a made thing that we make continuously.

The President: A source of optimism—I took my girls to see Hamilton, this new musical on Broadway, which you should see. Because this wonderful young Latino playwright produced this play, musical, about Alexander Hamilton and the Founding Fathers. And it's all in rap and hip-hop. And it's all played by young African-American and Latino actors.

And it sounds initially like it would not work at all. And it is brilliant, and so much so that I'm pretty sure this is the only thing that Dick Cheney and I have agreed on—during my entire political career—it speaks to this vibrancy of American democracy, but also the fact that it was made by these living, breathing, flawed individuals who were brilliant. We haven't seen a collection of that much smarts and chutzpah and character in any other nation in history, I think.

But what's most important about [Hamilton] and why I think it has received so many accolades is it makes it live. It doesn't feel distant. And it doesn't feel set apart from the arguments that we're having today.

And Michelle and I, when we went to see it, the first thing we thought about was what could we do to encourage this kind of creativity in teaching history to our kids. Because, look, America is famously ahistorical. That's one of our strengths—we forget things. You go to other countries, they're still having arguments from four hundred years ago, and with serious consequences, right? They're bloody arguments. In the Middle East right now, you've got arguments dating back to the seventh century that are live today. And we tend to forget that stuff. We don't sometimes even remember what happened two weeks ago.

But this point you made about us caring enough about the blood, sweat, and tears involved in maintaining a democracy is vital and important. But it also is the reason why I think those who have much more of an “us” versus “them,” fearful, conspiratorial brand of politics can thrive sometimes is because they can ignore that history.

If, in fact, you don't know much about the evolution of slavery and the civil rights movement and the Civil War and the postwar amendments, then the arguments that are being had now about how our criminal justice system interacts with African-Americans seem pretty foreign. It's like, what are the issues here? If you're not paying attention to how Jefferson and Madison and Franklin and others were thinking about the separation of church and state, then you're not that worried about keeping those lines separate.

Robinson: Exactly. I believe very much in teaching history. I spend an enormous amount of time working with primary sources and various sources and so on. And I think that a lot of the history that is taught is a sort of shorthand that's not representative of much of anything. I think that's too bad.

The President: Do you pay a lot of attention to day-to-day politics these days?

Robinson: I do actually. I read the news for a couple of hours every morning.

The President: Right. And how do you think your writer's sensibility changes how you think about it? Or are you just kind of in the mix like everybody else, and just, ah, that red team drives me nuts, and you're cheering for the blue?

Robinson: Well, if I'm going to be honest, I think that there are some political candidacies that are much more humane in their implications and consequences than others. I mean, if suddenly poles were to be reversed and what I see as humanistic came up on the other side, there I'd be. I think in my essay on fear I was talking about the assumption of generosity in this culture, you know?\* We have done some very magnanimous things in our history.

The President: Yes.

Robinson: Which seem in many ways unifying, defining. And then you see people running on what seem to be incredibly mean-spirited, tight-fisted assumptions, and you think, this is not us. This is not our way forward. Well, I'm getting all too political, but insulting people that you know will become citizens—however that's managed—giving them this bitter memory to carry into their participation in the national life. Why do that?

The President: We're going through a spasm of fear. And you're seeing it elsewhere. This is not unique to the United States. You see the emergence of the far-right parties in Europe. I think that it's a moment of great change, and the change happens fast. And there have been periods in our history where change happened fast like

this, and people just are trying to find firm footing.

When you're looking for firm footing, one of the easiest places to go is, somebody else is to blame. And the market system globally right now does create a situation where workers—ordinary people—have less control.

When you were growing up, when I was growing up, the majority of people had confidence that if they lost their job, it would be temporary, that they often would be with the same company for years, that there would be a pension in place, that they would be able to support a family, and that their kids would probably have a better life than they did. And people feel less confident about that because workers have less leverage, and capital is mobile and labor is not. And we haven't adapted our systems to take into account how fast this is moving.

What's frustrating to me is just that it wouldn't take that much for us to make the system work for ordinary people again.

Robinson: If I could strike one word out of the American vocabulary, it would be "competition." I think that that is the most bogus thing that has been entered into our [laughter]—

The President: Now, you're talking to a guy who likes to play basketball and has been known to be a little competitive. But go ahead. [Laughter.]

Robinson: But what we're really telling people is that if they do not acquire nameless skills of a technological character, they will not have employment. It will be shipped out of the country. So basically it's a language of coercion that implies to people that their lives are fragile, that is charged with that kind of unspecific fear that makes people—it's meant to make people feel that they can't get their feet on the ground.

The President: Right. Now, the argument would be, though, that that's the reality that people are feeling because companies can go anywhere and—

Robinson: Exactly, but when I look at these other economies we're supposed to be competing with, they're fragile. They're very fragile. And we're seeing that now. So all the competition has meant, it seems, is that labor is cheap and environmental standards are low. Look at, frankly, China. China has a vile ecology around its industrial centers. It's running out of appropriate cheap labor. And it's going into crisis. And what does that mean? It means that all of that capital will bundle itself up and land in another place that's relatively more advantageous. So what are we compet-

ing with? We run China into the ground, is that our great mission?

The President: Well, in fact, historically, the way we “competed” was we educated our kids better. We put more money into research. We believed in science and facts, as opposed to being driven by superstition. We welcomed talent from all around the world. We put in place a social safety net so people felt that they could take risks without—

Robinson: That’s crucial.

The President:—without being utterly destitute.

Robinson: And having good bankruptcy laws. We have very liberal bankruptcy laws. But you know, we generate fantastic ideas—ideas move as fast as capital does. We can have the most brilliant population in the world, and if the best ideas that we have are sent offshore, we’re still in the same position.

The President: Right. We made progress on all these fronts. Slowly but surely. Where I completely agree with you, Marilynne, is that we have everything we need to thrive. And it is interesting watching the current political season for me because I’m not on the ballot, so although obviously I still have a huge stake in the outcome as a citizen, in addition to soon being an ex-president—and there are times where I’m listening to folks make these wild claims about how terrible America is doing, and I want to just press the pause button here for a second and remind them that by almost every economic criterion we are hugely better off than we were just seven years ago; that we have done far better than almost every advanced country, and certainly every large advanced country on earth, in terms of growing the economy, driving down unemployment, managing our budgets.

And the only thing that right now is holding us back is Washington dysfunction. We could knock off another percentage point on the unemployment rate if we started rebuilding roads and bridges and airports. You travel—it’s embarrassing when you go to other airports in other countries. Ours used to be the nicest ones.

Robinson: They were nice first, and then all [laughter]—

The President: Yes. Now they’re a little worn down. We got to keep them up.

The same is true with our education system. It is outstanding, but we’ve got—everybody else is caught up. We got to step it up.

So one of the reasons I'm here in Iowa is to talk about two years of college education—or two years of community college education for everybody, as free as high school was before. Research—we have fallen behind in basic research that created all these amazing technological wonders upon which our economic engine ran.

And finally, making sure that people get paid enough money that they can support a family. Because all the evidence in history shows that when workers get paid a reasonable salary, then they spend it, businesses do better, the economy does better, and our political system does better. I mean, what is true is that when people feel pinched, then the generosity that you describe narrows to my immediate family, my immediate community, my immediate group.

Robinson: It's amazing. You know, when I go to Europe or—England is usually where I go—they say, what are you complaining about? Everything is great. [Laughter.] I mean, really. Comparisons that they make are never at our disadvantage.

The President: No—but, as I said, we have a dissatisfaction gene that can be healthy if harnessed. If it tips into rage and paranoia, then it can be debilitating and just be a self-fulfilling prophecy, because we end up blocking progress in serious ways.

Robinson: Restlessness of, like, why don't we do something about this yellow fever? There's generous restlessness.

The President: That's a good restlessness.

Robinson: Yes, absolutely. And then there is a kind of acidic restlessness that—

The President: I want more stuff.

Robinson: I want more stuff, or other people are doing things that I'm justified in resenting. That sort of thing.

The President: Right.

Robinson: I was not competing with anyone else. Nobody knew what my project was. I didn't know what it was. But what does freedom mean? I mean, really, the ideal of freedom if it doesn't mean that we can find out what is in this completely unique being that each one of us is? And competition narrows that. It's sort of like, you should not be studying this; you should be studying that, pouring your life down the siphon of economic utility.

The President: But doesn't part of that depend on people having different definitions of success, and that we've narrowed what it means to be successful in a way that makes people very anxious? They don't feel affirmed if they're good at something that the society says isn't that important or doesn't reward.

Probably the best example for me is the teaching profession, where I can't tell you how many kids I meet—and I used to meet them in law school when I was teaching there—who had taught for two, three, four years, they loved teaching, and they thought it was just the most important thing. And you could tell that this was their calling, and at a certain point they couldn't afford to raise a family on it and they got discouraged, and—

Robinson: Somebody was looking over their shoulder.

The President: Somebody was looking—or they'd get some comment from a classmate who had gone on to become an investment banker, they just eventually got discouraged and you didn't have a society that supported what they were doing, despite the fact that—talk about a complicated, magnificent art. Teaching. Being able to transmit ideas to young minds.

And so I like your definition of what America and freedom should be. But it does require all of us to have different definitions. And you have systems—or it requires a broader set of definitions than we have right now. And that's true for businesspeople, as well. I can't tell you how many businesspeople I meet [for whom] their joy is in organizing things to create products and services, and to help people be useful in various ways. And because they've got quarterly reports to shareholders and if they've made a long-term investment that may pay off way down the line, or if they're paying their employees more now because they think it's going to help them retain high-quality employees, a lot of times they feel like they're going to get punished in the stock market. And so they don't do it, because the definition of being a successful business is narrowed to what your quarterly earnings reports are....

So my last question to Marilynne is, when you think about your books and you think about your faith and you think about your citizenship as an American, when do you feel most optimistic? What makes you think, you know what, this experiment is going to keep going, I feel encouraged?

Robinson: Well, you know, I mean, when I do book signings, for example, and people come up one by one and talk to me about their lives, if there's time [to] do that, how earnest they are, how deeply committed they are to sustaining people they feel close to or responsible for and so on—there they are, the people that you think of as

the sustainers of a good society.

And it's only—really, if we could all just turn off media for a week, I think we would come out the other side of it with a different anthropology in effect. I wish we could have a normal politics where I disagree with people, they present their case, we take a vote, and if I lose I say, yes, that's democracy, I'm on the losing side of a meaningful vote.

The President: And I'll try to make a better argument the next time.

Robinson: Exactly.

The President: I'll try to persuade more people the next time.

Robinson: And I think in little groups, like my department at the university or something—people get together, talk something over, take a vote, and that's it. And it's a little microcosm of democracy. That's what it's supposed to be.

The President: Yes, but that does require a presumption of goodness in other people.

Robinson: Absolutely.

The President: And that's not just what our democracy depends on, but I think that's what a good life depends on. Occasionally, you'll be disappointed, but more often than not, your faith will be confirmed.

Robinson: I believe that.

# Multithreaded Toolkits: A Failed Dream?

Graham Hamilton, Java.net, October 19, 2004

The question came up recently of “should we make Swing truly multithreaded?” My personal answer would be “no”, and here’s why...

## The Failed Dream

There are certain ideas in Computer Science that I think of as the “Failed Dreams” (borrowing a term from Vernor Vinge). The Failed Dreams seem like obvious good ideas. So they get periodically reinvented, and people put a lot of time and thought into them. They typically work well on a research scale and they have the intriguing attribute of almost working on a production scale. Except you can never quite get all the kinks ironed out...

For me, multithreaded GUI toolkits seem to be one of the Failed Dreams. It seems like the obvious right thing to do in a multithreaded environment. Any random thread should be able to update the GUI state of buttons, text fields, etc, etc. Damned straight. It’s just a matter of having a few locks, what can be so hard? OK, there are some bugs, but we can fix them, right? Unfortunately it turns out not to be so simple...

From observation, there seems to be an amazing tendency towards deadlocks and race conditions in multithreaded GUIs. I first heard about this issue anecdotally from people who had worked with the Cedar GUI libraries at Xerox PARC in the early 80’s. That was a community of extremely smart people who really understood threading, so the assertion that they were having regular deadlock issues within GUI code was intriguing. But maybe that was flawed data or an exceptional situation.

Unfortunately that general pattern has been repeated regularly down the years. People often start off trying for multithreading and then slowly move to an event queue model. “It’s best to let the event thread do the GUI work.”

We went through this with AWT. AWT was initially exposed as a normal multi-threaded Java library. But as the Java team looked at the experience with AWT and with the deadlocks and races that people had encountered, we began to realize that we were making a promise we couldn’t keep.

This analysis culminated in one of the design reviews for Swing in 1997, when we



reviewed the state of play in AWT, and the overall industry experience, and we accepted the Swing team's recommendation that Swing should support only very limited multi-threading. With a few narrow exceptions all GUI toolkit work should occur on the event processing thread. Random threads should not try to directly manipulate the GUI state.

Why is this so hard?

John Ousterhout gave a great Usenix talk on Events versus Threads in 1995 that explores some of the tradeoffs between thread-driven and event-driven programming and he correctly points out many reasons why multi-threaded programming is hard and why event driven programming can be simpler. I don't necessarily agree with his analysis for all kinds of programs, but I do agree for GUI programs.

The particular threading problems of GUI toolkits seem to me to arise from the combination of input event processing and abstraction.

The problem of input event processing is that it tends to run in the opposite direction to most GUI activity. In general, GUI operations start at the top of a stack of library abstractions and go "down". I am operating on an abstract idea in my application that is expressed by some GUI objects, so I start off in my application and call into high-level GUI abstractions, that call into lower level GUI abstractions, that call into the ugly guts of the toolkit, and thence into the OS. In contrast, input events start off at the OS layer and are progressively dispatched "up" the abstraction layers, until they arrive in my application code.

Now, since we are using abstractions, we will naturally be doing locking separately within each abstraction. And unfortunately we have the classic lock ordering nightmare: we have two different kinds of activities going on that want to acquire locks in opposite orders. So deadlock is almost inevitable.

This problem will initially surface as a series of specific threading bugs. And people's first reaction is to try to adjust the locking behavior to resolve the specific bugs. Let's release that lock there and then lets use more clever locking over here. Well, that is kind of a fun activity, but it is trying to fight back an oceanic tidal force. The cleverer locking typically turns into a combination of subtle races (due to lack of locking) or clever and intricate deadlocks (due to the clever and intricate locking). We went through a bunch of that in 95-97.

Notice that the problems extends beyond the GUI toolkit layers and also appears between the toolkit layer and the application level. With great difficult one might

try to adopt a single lock for all activity within the GUI layer, but the same problem then resurfaces a level up.

So what's the answer? Well, at some point you have to step back and observe that there is a fundamental conflict here between a thread wanting to go "up" and other threads wanting to go "down", and while you can fix individual point bugs, you can't fix the overall situation.

This led to the solution that the Swing team adopted and which is used by most leading GUI toolkits: run all GUI activity on a single event thread. This means that in some sense all GUI activity becomes event driven, and the "down" threads become just a new kind of event.

This demonstrably works. It is possible to write complex GUI apps that work reliably. Hurrah! But it does make managing long running activities tougher. I wrote a smallish Swing program that I use periodically to selectively zap large boring attachments from my email archives. I don't want to hang the GUI while it reads tens of megabytes of emails, and I also want to display a progress monitor, so I ended up having to carefully balance handing off big activities to worker threads and handing GUI activities back to the event thread. It is probably more complicated than it would be if I had a magic multi-threaded library, but it has the significant saving grace that it actually seems to work reliably.

## Subtleties

Are things really so black and white? Surely there have been people who have used multi-threaded toolkits successfully? Yes, but I think this demonstrates one of the characteristics of the Failed Dreams.

I believe you can program successfully with multi-threaded GUI toolkits if the toolkit is very carefully designed; if the toolkit exposes its locking methodology in gory detail; if you are very smart, very careful, and have a global understanding of the whole structure of the toolkit.

If you get one of these things slightly wrong, things will mostly work, but you will get occasional hangs (due to deadlocks) or glitches (due to races). This multithreaded approach works best for people who have been intimately involved in the design of the toolkit.

Unfortunately I don't think this set of characteristics scale to widespread commercial use. What you tend to end up with is normal smart programmers building apps that don't quite work reliably for reasons that are not at all obvious. So the authors

get very disgruntled and frustrated and use bad words on the poor innocent toolkit. (Like me when I first started using AWT. Sorry!)

Another wrinkle: it is possible to have multiple simultaneous GUI activities within a Java VM by using multiple event threads. That works provided the different activities are almost entirely isolated, have their own distinct GUIs (no shared components or mixed hierarchies) and provided the very lowest toolkit level can correctly dispatch events to the right event thread with minimal locking. This is useful in (for example) running multiple applets within one JVM. But it isn't a very general solution - most applications need to live within the constraint of only a single event thread.

In this note I've most been covering why Swing and other toolkits are essentially single-threaded. Chet recently blogged on some related topics around why multi-threading complicates user programs and normally won't help raw graphics performance.

Also, before I forget, some people are probably remembering that "processes and monitors are duals". Well, yes, it's true. In some sense we are using the event thread to implement a global lock. We could invert things, and create a global lock that is equivalent to the event queue. This would be fairly ugly and would require wide coordination and undermine a lot of abstractions. But the larger problem is that Java developers tend to use multiple locks and if they are to preserve the equivalence with an event queue model, they will need to follow various non-obvious rules about how they interact with these other locks.

The event queue model makes the central single lock much more visible and explicit, and on the whole that seems to help people to more reliably follow the model and thus construct GUI programs that work reliably.

## Conclusion

I guess the bottom line is that like many others I would really like to see a flexible, powerful, truly multi-threaded GUI toolkit. But I don't know how to get there - at this point there is fairly strong experience that the obvious approaches for multi-threading don't work. Maybe in future years people will come up with a radically new and better approach, but for now the answer seems to be that events are our friends.