

Enhancing the Epistemological Project in the Rhetoric of Science



Information Infrastructure as Tool for Identifying Epistemological Commitments in Scientific and Technical Communities

Nathan R. Johnson

*Purdue University
West Lafayette, IN*

Poroi 10,2 (December 2014)

Keywords: information infrastructure; rhetoric; science and technology studies; Alan Gross; standardization

Alan Gross's contributions to the Rhetoric of Science are colossal. He's an instigator, taking stances that are often radical simply for the sake of exploring an idea. His radical notoriety is especially true for people reading him from outside his home discipline. For example, consider how critiques of his original *Rhetoric of Science* frequently took him to task for taking the indefensible position that all science is rhetorical. Gross notes that historian Susan Haack thought that,

Gross's view of science is like an atheist's view of theology. A belief that a deity exists is the psychological anchor that makes a life in theology possible; however, an atheist would add, since this belief is false, theologians (though not exactly frauds, and not exactly writers of religious fiction) are in effect self-deceived fabulists. But science is different from theology in a crucial respect: that there is a real world which in a certain way and not other ways is a presupposition not only of scientific inquiry, but of all empirical inquiry, including the most ordinary everyday investigation into whether that check has cleared, what caused that leak in the roof, etc. (Gross, 2006, 3)

In this text, Gross self-consciously highlights a review that depicts him as an anarchist. This is the way he works: He takes a radical position from the status quo to explore an idea.

As evidence, consider how Gross's more recent work on visual rhetoric is a far departure from his older radical rhetoric of science (Gross, 2009). Gross may occasionally be critiqued as brash or inexact, but he makes big ideas happen by making claims that others wouldn't attempt (Harris, 1991). As he had previously suggested,

No doubt, were I born in the thirteenth century, I would have selected the *Summa Theologica* as my research site, and have been burnt at the stake for my pains. My reasoning went something like this: If rhetoric was epistemic, then the rhetorical analysis of science should yield an appropriate epistemological and ontological harvest. And it has, to my satisfaction, at least (Gross, 1990, 304).

In particular, two of Gross's ideas are fundamental for rhetoricians of science. First, he pushes the epistemological limits of rhetoric in science (Gross, 1990). His stance has been so extreme at times that it changed the norm of rhetorical epistemological in science. And while many assume that Gross goes too far, he has pushed the boundaries that allow space for less extreme but useful stances.

Second, and relatedly, he has encouraged interdisciplinarity. Most notably, his work became a lightning rod to which other extra-disciplinary scholars felt compelled to respond. Sure, he has been frequently critiqued as mistaken, but attention of any kind builds community. This is particularly useful for rhetoricians of science, since interdisciplinarity has historically been a problem in the discipline (Ceccarelli, 2005; Graham, 2011). Rhetoricians of science have typically been importers of other literature and largely ignored by others in Science Studies.

In this paper, I extend Gross's work in two ways. First, I locate new topics for energizing an extreme view of rhetoric in science as epistemological. I suggest that by focusing on information infrastructure, rhetoricians of science can better argue for a deeply rhetorical science. Second, I provide a bridge to Science Studies by focusing on a key area that is particularly fruitful for collaboration, the STS sub-discipline of information infrastructure studies. Current STS research takes interest in infrastructure as a global force (Bowker, Baker, Millerand, and Ribes, 2007; Edwards, 2003) and in acknowledging that data transfer is necessary for science

today (Edwards, Mayernik, Batcheller, Bowker, and Borgman, 2011; Edwards, 2010).

To achieve the aims of this article, I first review information infrastructure studies (including aims that it shares in common with epistemologically-grounded rhetorical studies). Then I provide a case that highlights how one infrastructure was developed to reflect epistemological and professional values. In conclusion, I point toward the ways in which the study of information infrastructure complements the rhetorical study of science. Rhetoricians of science might fruitfully participate alongside scholars of information infrastructure studies for our shared epistemological project.

Information Infrastructure Studies as Complement to Rhetorical Studies of Science

“Infrastructure” is frequently used to describe any support system. In this spirit, some have called writing centers and other campus resources infrastructures (DeVoss, Cushman, and Grabill, 2009). This isn’t wrong, but it doesn’t provide a good way to make sense of infrastructure as it might enhance work in the rhetoric of science, which can easily fade from plain sight (Bowker and Star, 1999).

To help, STS researchers have developed a number of different frames for understanding infrastructure. Sociologist Susan Leigh Star states that infrastructure is relational, meaning that it is an ever-present foundation that provides the background to a more noticeable foreground. So then, it helps to think of infrastructures as relational support. Still, relationality is a difficult concept with which to work. What do infrastructures support? What activities? For example, is infrastructure the supporting mechanism of active political citizenship? Is it the social programs that support libraries in this country? Is it the software that engineers depend on to perform their work? Is it always present? Infrastructures are multiple. Better conceptual tools, it seems, are needed to understand them (Bowker and Star, 2000).

Four concepts from STS help make sense of infrastructural relationality: standards, classifications, protocols, and algorithms. These entities are durable, providing the foundation for more noticeable action.¹ I define them thus:

¹ They are versions of Bruno Latour’s immutable mobiles, objects that retain key features as they reappear in different contexts (1987). When Latour uses the word, he is frequently describing things like journal publications that physically move from place to place. Infrastructure

Standards are agreements about uniformity (Feng, 2003). They are frequently created through committees because of the amount of work that it takes to distribute them throughout infrastructure. For example, railroad ties needed to be standardized so that locomotives could depend on the same rail size (Schivelbusch, 1986).

Classifications are consistent naming conventions. Classification systems are the linguistic equivalent of kitchen cupboards and shelves for utensils. They provide an intellectual naming space. When libraries, for example, use naming conventions for books and then duplicate them in different spaces, they create an infrastructure for dependable storage, access, and retrieval. Classifications are human language and hence particularly responsive to rhetorical critique. Classifications are central to information infrastructure because of the rhetorical work of naming (Olson, 2002).

Next, *protocols* are standardized procedures. Whereas standards are frequently static measurements, protocols are rules to follow (Galloway, 2004). For instance, the Internet's TCP/IP technology defines a set of back and forth responses for error checking and data sharing. It's an agreed-on set of rules for computers.

Finally, *algorithms* are reproducible techniques of analysis (Johnson, 2012). They are recipes that emerge as analytic tools because of the standards, classifications, and protocols of infrastructure. Algorithms depend on the consistency of the other infrastructural elements for their sensibility. They are procedures of calculation. For instance, logical algorithms frequently depend on the standards, rules, and procedures defined in symbolic logic paradigms. Together with standards, classifications, and protocols, algorithms provide the dependable durability of infrastructure.

Because of these dependable objects, infrastructure makes work easier. For instance, medical infrastructure helps with diagnosis. Without it, it's not easy to decide when someone has a disease or not. What collection of symptoms could possibly add up to what disease? A classification system like the International Classifications of Diseases provides rules that help diagnose thousands of diseases by providing cookbooks (Bowker and Star,

components are different. Because they are standardized, they don't move. Multiple copies of the same object appear throughout infrastructure. For instance, the *size* of a railroad tie is a standard. Infrastructure depends on the same form being accessible in different locations so that the train can roll on.

1999). Providing sensibility is a fundamental way that infrastructure rhetorically acts on the world. Those classifications don't fit everyone: they provide sensibility partly by fitting their objects to their spaces.

Consider a medical classification again. Medical diagnosis is hard work. Symptoms of disease frequently are blurry. They don't straightforwardly add up to a disease. And what about the symptoms that have not been detected or articulated (Mol, 2005)? Differences between individuals start blurring themselves to fit the language of the classification. For instance, individual differences based on BMI or weight produce different variations of disease. Even though BMI is favored as a means of defining disease, it still took a committee of medical professionals to standardize the measurement and then classify weight related diseases with it. That standard works better for some people than others.

This example shows how infrastructural standardization is an important epistemological tool. Measurements like BMI need to have standards that define what healthy/unhealthy are. They are kept in a calculated table. The algorithms that define that table in combination with the standardized measurement produce the ability to classify disease based on height and weight criteria. Medical professionals create standardized forms of disease for standardized publics. Those standards, algorithms, and classifications are then used to diagnosis an endless spectrum of specific situations, from wailing babies to cranky elders to muscly bodybuilders.

The issue is larger than medical diagnosis, though. My study on web standardization showed how much rhetorical work goes into creating standards that will ultimately benefits some at the expense of others (Johnson, 2009). The Web Standards Project worked to enforce a standard that was defined by an international community rather than a proprietary vendor. But even that standard ultimately served the technological interests of existing parties by fitting into the practices that had been established 10 years earlier. It changed the economic field by providing more institutional power to some vendors who had already integrated versions of the standards into their products. Because of this, the way web space was represented favored some media over others. Sites created for non-standardized browsers were less usable as the standard was distributed.

Proprietary interests are one thing, but these standards are more pervasive than simply favoring some technologies over other. They change space and time. Wolfgang Schivenbusch's research on the British railroad demonstrated how the speed produced by a

standardized infrastructure changed how people understood distance and speed between locations (Schivenbusch, 1986). When it's easier to get somewhere faster, it seems that places are closer together. Infrastructural objects are hence epistemological objects. They change the perception of reality. The arguments surrounding the nature of those standards are important for creating that perception of time and space that is then used for more foregrounded action. If a standard, classification, protocol, or algorithm changes how the past is understood, if a standard, classification, protocol, or algorithm changes how space is understood, they are the infrastructural mechanisms that are rhetorical in the study of science, technology, medicine, engineering, and mathematics.

To show the fundamental complementarity of the work of rhetoricians of science with current understandings of infrastructure, I provide a case study in which an idea of the past is built through the standardization process. This study analyzes a dispute over one web browsing protocol—the DOCTYPE switch—that changed time perception. I analyze the arguments surrounding the protocol and show how the technological protocol was considered before it was institutionalized as a legitimate timekeeping device for the World Wide Web's infrastructure. In doing so, I push Alan Gross's notions of rhetorical epistemology forward by locating a niche within the study of science that is rife for cross-disciplinary work.

The DOCTYPE Controversy: Setting Standards for Web Pages

Web pages are displayed in browsers with two standardized technologies: HTML and CSS. The HTML standard defines a language for arranging documents. The CSS standard defines a visual language for changing how documents look. Popular browsers like Internet Explorer, Mozilla Firefox, or Google Chrome depend on those standards to display pages to users.

The HTML and CSS standards are under constant development. This has been the case for HTML since Tim Berners-Lee published a description in 1991. CSS has been under development in different forms since around 1994. Both standards are constantly changing. One of the reasons the web of 2000 looks different than the web of today is that its technical standards have been updated to meet the changing demands of content creators. New standards provide new web capabilities. Early versions of both standards were simpler and therefore web pages were simpler. As more people became

interested in using the web, web writers placed more demands on how it worked. The semantic, visual, and performative language of the web changed. The standards were updated as part of this process.

Various committees of the World Wide Web Consortium handle standards updates. The committees publish documents that explain how software manufacturers should incorporate the standards into their technologies. This process is neither straightforward nor easy. Incorporating a standard requires interpreting a complex technical document. Standards documents are just as open to interpretation as other texts. For instance, a difference in how browsers should interpret CSS's "box model" led to different renderings of the same text and graphics based on the browser being used. When Microsoft's interpretation deviated from other manufacturers', web pages looked different when displayed in Microsoft's Internet Explorer. Web designers disliked this because it created graphic design problems (Johnson, 2009). In this case, designer desire for uniform visual layout was exposed as a central infrastructural concern of the web. Protest groups were even deployed to resist the manufacturers' interpretation of the standard. One misinterpreted standard can highlight important issues about what is important for people publishing content on the Web.

As standards are developed, browser technologies need to be updated to support the newly standardized technologies. This process often takes time. After the standard is available to the public, engineers need to redevelop the old technology to meet the new standard specifications. This is not unlike when emissions standards are passed for autos. It takes time for manufacturers to put the new standard in their manufactured technologies. Grace periods are the norm. In addition, some users will continue using older technologies for a while after the new standards are passed. Because different manufacturers develop browsers, some manufacturers end up supporting newer standards before others. Therefore, competing web browsers have never worked exactly the same way. In addition to controversy over what standards are important, controversy emerges because of the various timelines for technological adaptations to the new standards are involved.

Complicating matters, new browser standards do not automatically update web pages designed for previous standards. Timeliness is important for all technologies involved in a networked infrastructure because of the technological dependabilities involved. If a technical standard deviated substantially from past rules for displaying the web, it can change how older web pages are displayed to users because those pages had been designed with the

older technology in mind. This problem is often handled by authoring new standards specifications that are backwards compatible with other technologies, meaning that new standards documents add to old rules without overwriting older capabilities. Ideally, new technologies are designed in ways to keep the old web pages working. Although that's the ideal, it's not always possible. Occasionally newer technologies make big problems for old content and interdependent technology. This is especially true when you consider that not only do newer technologies render web pages differently, sometimes the browsers also render the standards differently. Occasionally, large-scale corrections are needed to fix messes created by the cycle of standards, content, and browser development.

DOCTYPE as Response to Large Scale Infrastructural Changes

Because of these cycles of change, it has often been important to develop techniques that help with large-scale infrastructural changes. One of these techniques was the DOCTYPE switch. Around 2000, the rendering in web browsers had become so diverse that a new display technique—quirks mode—was added. Manufacturers started releasing browsers so that a developer could add a command to HTML documents that either triggered a normal mode or a “quirks mode.” Documents displayed in quirks mode used older outdated standards. The newer standards mode would be used if the browser recognized a DOCTYPE command at the beginning of the HTML file. The idea behind the technique was that web creators who knew what they were doing would use the command to start making documents with the new standards. Documents previously created for the older standards would continue to be displayed as they had been with the older standards. The large-scale change meant adding a new rule that required expertise and practice. Developers needed to know the difference between normal and quirks mode.

That was the ideal, but several challenges kept that idea from working. Quirks mode never worked across browsers in the way that it was ideally intended. Developers often created content that was “hacked” as a way to handle the uneven distribution of web browser standards. This meant that code was added to web pages that specifically fixed some browser default settings while deviating from the standards. Historically, these “hacked” documents have become buggy when valid updates are made to browsers. Second, novice web developers often wrote content that appeared to work but did not meet the standards requirements. These web pages

worked immediately, but frequently didn't last very long as browsers were updated.

There have been several reasons designers “hack” or don't always write HTML documents to exact specifications. One of them is that designers have often been self-taught. Typically, self-teaching has consisted of copying the techniques of others by viewing and duplicating code techniques. Frequently, this also meant copying code that wasn't understood by the learner. Non-standardized practices frequently work in the short-term, and they are easy to disseminate, but they frequently fail in the long-term. When designers copy from many different examples, they create hybrid documents with some standardized and some non-standardized code. Self-taught designers are often unaware that standards even exist. They frequently produce documents that work immediately, not necessarily those that conformed to an exact technical, especially when a hack works. This type of buggy content often works immediately, but often stops working as browser technology improves. The DOCTYPE switch was frequently copied into documents without the developers realizing what it did. The rest of the document might not have been written to any sort of specification even though the DOCTYPE indicated that it had been.

Third, some production tools like Dreamweaver produce content that appears correct but doesn't necessarily enforce all development standards. When browser technologies are updated, pages produced that way often degrade as later browser updates are made. When the DOCTYPE switch was used as a part of a technology, developers who were producing buggy code frequently would use it with buggy code. This meant that the DOCTYPE switch, although well intentioned, was used in unanticipated ways almost immediately. The big web fix didn't work as well as it should. In total, this meant that the quirks mode fix depended on a number of social practices that couldn't be easily fixed.

In the end, the original quirks mode failed as a large-scale technological fix and simultaneously added a new snag to designing web pages.² The DOCTYPE switch initially depended on a piece of code that browsers used to determine that the web document was correctly formed. The DOCTYPE switch was a hack that depended

² Anselm Strauss calls this a cumulative mess trajectory (Strauss, Fagerhaugh, Suczek and Wiener, 1997). This occurs when a technological fix creates a set of new problems. The next technical fixes result in even more problems. And so on and so on. (Strauss et al., 1997, 161–181).

on developers writing an impossibly perfect document. Work practices in the wild simply didn't support this kind of technique. Depending on developers to consistently perform an important part of the web infrastructure wasn't feasible. The DOCTYPE switch solution depended on human infrastructure that didn't exist.

In 2008, the team at Microsoft tried to work around that problem by changing how the quirks mode technology worked. Microsoft's update to the DOCTYPE switch was called "version targeting." Instead of using a quirks mode and a standards mode, they created a technique that required that developers specify which browsers they knew were compatible with their web pages. So instead of simply adding a command that indicated the document was written correctly, the developers would need to specify which browser or browsers the document worked with. It required metadata that recorded a version and the specific technology involved (Zeldman, 2008). If no version was found by the browser, the page would be rendered with the oldest available browser version.)

The details of that code are less relevant to this study than the discourse surrounding the event, which highlighted how cultural values and epistemological assumptions become part of information infrastructure.

Technological Infrastructure and Epistemological Assumptions in the DOCTYPE Case

The cultural values motivating and manifested in the DOCTYPE case were highlighted in a series of articles published within *A List Apart*, one of the more popular sources of information web designers about standards, examined how the vendors were implementing DOCTYPE in their browser technologies (Gustafson, 2008; Keith, 2008; Meyer, 2008; Zeldman, 2008). In particular, the articles focused on the drastic changes that Microsoft was making to the DOCTYPE. This series was unusual for *A List Apart* in that each was written as part of a deliberative forum. Several prominent speakers and writers weighed in on the issue, including Aaron Gustafson, Jeremy Keith, Eric Meyer, and Jeffrey Zeldman. More importantly, these articles were significant for the mundane nature of their conceptual topography and obviousness of the claims to the communities involved. These articles are particularly relevant for how they distilled the issue for designers while talking about how the web was valuable for a larger community. The articles in *A List Apart* were concerned with how vendor changes in DOCTYPE technologies would affect websites created in the past as

the vendors updated their technologies to support newer standards. These articles were about what vendors should do to ensure that older sites written with revised web standards would be handled by newer browser technologies.

The *A List Apart* forum distilled the issue for readers. The articles suggested that as vendors increasingly supported more standards, they were forced to implement current standards correctly while supporting websites that were created with older non-standard technologies--to ensure that older sites written with revised web standards would be handled by newer browser technologies.

Complexity is heralded as a solution to a sociotechnical problem. If the problem was that the artifacts of the past are becoming more complex, the answer is that the programming needs to become more complex in order to save the past and safeguard the future. And much *A List Apart* discourse was devoted to using this heuristic to offer solutions to the problem. But this was a point of contention. Many didn't support further atomization and complexity.

This was the context that elicited the *A List Apart* forum articles. In that forum, although the discussion was richly detailed, essentially two sides emerged during the *A List Apart* forums: those who favored the version targeting proposed by Microsoft, and those who didn't.³ While there were only four primary authors within the forum, the comments on each article ran into the hundreds. Those in favor saw version targeting as a way to protect the work of the past while saving work on behalf of the developer. Essentially browser vendors offering version targeting were assuring web creators that they would continue supporting older formats indefinitely—those developers simply needed to adjust for the increased complexity. Those who opposed version targeting described it as harmful for the future of the web. In essence, version targeting “locked” a site to a specific browser version, supposedly guaranteeing that the site would never again be able to render with standards created in the future. And although this may seem relatively minor, it's important to note that many designers think of sites as complex living entities that continue to grow and adapt as technology changes. This philosophy is part of design patterns called “progressive enhancement” or “responsive design.” This philosophy was popularized through an article by John Allsopp called “A Dao of Web Design,” which espoused fluidity and

³ This discussion can be seen linked to each of the primary articles.

impermanence as part of web writing (Allsopp, 2000). Critics believed version targeting would make future updates to sites difficult if not impossible without fully rewriting and redesigning sites with every new browser release. In other words, when a choice was presented to the *A List Apart* community over protecting the past or building for the future, the writers and surrounding reading community were split.

This split was over fundamental differences in how each group understood standards. On the one hand, the individuals that supported the new Microsoft DOCTYPE switch liked the idea of a more flexible and evolutionary perspective on the standards. Consider the following statements from Eric Meyer, one of the most prominent supporters of the switch, urging readers to give DOCTYPE a chance.

Since the inception of the web, with the sole exception of DOCTYPE switching, browsers have been a “what I do is what you get” proposition. Developers have been forced to conform to past browsers’ behaviors while making educated guesses about what future browsers would do. . . . So in the end, and much to my surprise, it turned out that I don’t hate the idea after all. Version targeting allows browsers to much more easily develop new features and fix bugs and shortcomings in existing features, which has the potential to speed up the evolution of web design and development. That alone is reason enough to give it a chance (Meyer, 2008).

By suggesting that the new technology may be able to overturn past development problems caused by “developers” being at the mercy of “past browsers’ behaviors,” Meyer suggests that the benefits gained by browser switching outweigh the negatives. In this case, the benefits outweigh the negatives of losing a large chunk of the past. For Meyer, the future is better as it would have “the potential to speed up the evolution of web design and development.” He suggests that the future is too unpredictable to plan for, and it is better to provide safeguards for preserving the past, even if it means that older web content won’t necessarily be able to be guaranteed to work in newer browser technology. Some sort of certain access is better than depending on an unknown future technology to provide seamless access.

Meyer is not alone in his suggestion that the browser switch is a positive change. Aaron Gustafson seconds his belief that “progress always comes at a cost.” He goes further, suggesting that an

implementation may cause immediate discomfort for developers, but that the break from the past will enable a better future web:

In addition to the headaches of writing cross-platform styles and scripts, we've had to deal with the fallout from new browser releases that inevitably broke something we couldn't possibly have anticipated. It's never fun explaining the cause of an unexpected break to our clients, bosses, and users. But with IE8's introduction of version targeting, there is a light at the end of the tunnel. I, for one, hope other browser vendors join Microsoft in implementing this functionality (Gustafson, 2008).

A similar sentiment is expressed by both Meyer, Gustafson, and in the comments of their supporters. The web, in terms of development and design, will be better if a break is made with the past, despite the immediate discomfort for developers. Meyer and Gustafson both highlight a past that has been less-than-ideal in order to make this claim. Content on the web should be expected to have a time and technology related context. The infrastructure can't support content ubiquitously as though those constraints do not exist.

On the other side of the debate were individuals interested in safeguarding current work well into the future. The concept of a web standard is something completely different for the forum participants who were against the new version targeting. For them, a web standard is something more static and stable. Those opposed to the switch essentially argued that although the standard may change, there is no reason to panic if pages aren't displayed identically, as evidenced in statements against the new DOCTYPE switch:

First, what's at issue here is not "the web" but "some websites". Second, rather than "breaking", it's more accurate to say, "displaying differently." Finally, it's important to remember that we are talking about how websites are displayed in one browser: when the IE team talk of "breaking the web," what they really mean is that their browser will display documents in much the same way as other modern browsers do. Would that really be such a bad thing (Keith, 2008)?

Jeremy Keith and others agree that it's better to depend on browser technologies to display pages appropriately well into the future. There is no need for designers to "lock" their web content to a certain type of browser. It's more important that changing

browser technology can always display content regardless of its historical circumstances.

This case demonstrates an epistemological controversy in which the decisions that are made will generate an infrastructure that favors one type of knowing over another. If the Microsoft DOCTYPE switch were adapted as an infrastructural technology, web content would be easier to understand because of the built-in provision that pushed designers to identify a browser and version for the web content they produced. If the alternate DOCTYPE technology were to be used, documents wouldn't be marked as belonging to a specific technology and time. They would be published and distributed as ubiquitous parts of a larger pool of content.

In this case, the Microsoft DOCTYPE was never adapted. Web content is distributed with one correct DOCTYPE that acts as a universal statement across browser technologies. Browser technologies are therefore hidden as important historical parts of web production.

Infrastructure and Epistemology

The DOCTYPE controversy exposes how standards highlight contrasting epistemological understandings inherent in a profession. Professionals discussed different views of how past work in the field should be understood and accessed into the future. A conclusive agreement of the DOCTYPE technology was built into future technology. Browser developers decided that requiring that targeting individual browsers for providing more seamless access to the past was not a feasible plan. Henceforth, the decision has resulted in the development of technological standards in which the browser cannot be depended on as part of the archival process. Information about the past does not include information about the types of browser technologies and tools that helped inform the content.

What exactly is lost by not including that information of the past? For one, we lose the ability to discover more complex information about whose technologies were crucial for shaping the content that people enjoy today. Without knowing what browser was targeted, we lose the capability to know which audiences were targeted because they used a specific browser. (This is a substantial amount of information given that many large businesses insist on using specific types of technology in house.)

Perhaps more importantly, however, is what was deemed important to forget. Memory is expensive. That is not because of

the technological capability of storage: far from it. Memory is expensive in the information age because in a world in which everything can be remembered, the decision to forget is an important one. As a community, it is a significant task to consider what we need to forget to move forward.

Epistemology, Rhetoric of Science and Information Infrastructure Studies

In demonstrating that information infrastructures—standardization, classification, protocol, and algorithms—are the epistemological linchpins of a profession, I have suggested how Gross’s radical epistemological stance toward science can be taken up in a second wave. The tools of information infrastructure studies are a toolkit complementary to work in the rhetoric of science.

Gross has suggested problems with the use of case studies as a way of understanding rhetoric in science. He has written, for instance, that, “They pile up; they do not add up” and that he and other rhetoricians of science “must defend case studies as a social scientific method” (Gross, 1994, 11). While Gross makes this defense by suggesting that other modes of scientific inquiry are just as perilous and uncertain, I am suggesting that infrastructure provides a middle ground with loaded traffic between the discursive and material. That is, as infrastructures are discussed and changed, they become material realities for infrastructural users. There are clear implications from noting how specific cases propagate into global use. In the same article, Gross pointed out that in many forums “what counts as a fact depends not on science, but on the trust the public bestows on scientists” (Gross, 1994, 18). Infrastructure is the trust that a public has that they can do everyday work. When it breaks down, publics lose their faith in science and technology. In the case I outlined here, I showed a material future that would never be on the web that was coordinated and deployed smoothly without public intervention. The public trusts a working infrastructure.

In his “Rhetoric of Science is Epistemic Rhetoric,” Gross took on other rhetoricians who balked at rhetorical analysis of scientific texts as epistemic work (Gross, 1990). He defended that issue head on, and I find myself compelled by both him and his interlocutors. While not claiming to have offered a solution as to whether rhetoric of science is always epistemic, I suggest that analyzing how and why people talk about the instruments and techniques of science provides an alternative approach. Knowledge creation might not end immediately in discussion, but one of its important points

occurs in the material standardization of infrastructure. In the infamous Gaonkar discussion, Gross defended invention as the foundation of rhetorical knowledge (Gross, 1997). I say, what better foundation than the discursive inventions surrounding the invention of scientific techniques? This article presents the conceptual efficacy of the technique. Upcoming work will demonstrate the technique itself.

Information infrastructure studies gives us the tools for understanding the work of the building blocks of science while rhetorical studies gives us the tools for parsing the discourse of the building blocks of science. Utilized as complementary tools, we can see even deeper into the epistemological structures and assumptions at work in scientific and technical professions. Rhetorical analysis of infrastructural debates, technologies, and techniques highlight important issues about how rhetoric works in the scientific and technical spheres in the modern age. Following Gross, infrastructural rhetoricians might be burnt at the stake, but the work that they do carries on a legacy that has generated significant intellectual discussion over the last thirty years.

Reference List

- Allsopp, J. "A Dao of Web Design." *A List Apart* 58, (2000).
Retrieved from <http://www.alistapart.com/articles/dao/>
- Bowker, G. C., K. Baker, F. Millerand, and D. Ribes. "Toward Information Infrastructure Studies: Ways of Knowing in a Networked Environment." In Hunsinger, J., L. Klastrup and M. Allen (Eds.), *International Handbook of Internet Research* (Pp. 97–118). New York, NY: Springer Verlag, 2007.
- Bowker, G. C. and Star, S. L. *Sorting Things Out: Classification and Its Consequences*. Cambridge, MA: MIT Press, 1999.
- "Invisible Mediators of Action: Classification and the Ubiquity of Standards." *Mind, Culture & Activity* 7 (2000): 147–163.
- Ceccarelli, L. "A Hard Look at Ourselves: A Reception Study of Rhetoric of Science." *Technical Communication Quarterly* 14 (2005): 257–265.
- DeVoss, D. N., E. Cushman, and J. T. Grabill. "Infrastructure and Composing: The When of New-Media Writing." *College Composition and Communication* 57 (2009): 14–44.

- Edwards, P., M. Mayernik, A. Batcheller, G. Bowker, and C. Borgman (2011). "Science Friction: Data, Metadata, and Collaboration." *Social Studies of Science* 41 (2011): 667-690.
- Edwards, P. "Infrastructure and Modernity: Force, Time, and Social Organization in the History of Sociotechnical Systems." In T. J. Misa, P. Brey and A. Feenberg (Eds.), *Modernity and Technology* (Pp. 185–225). Cambridge: MIT Press, 2003.
- A Vast Machine: Computer Models, Climate Data, and the Politics of Global Warming*. Cambridge, MA: MIT Press, 2010.
- Feng, P. "Studying Standardization: A Review of the Literature." In Egyedi, T.M., K. Krechmer and K. Jakobs (Eds.), *Standardization and Innovation in Information Technology* (Pp. 99–112). Delft, The Netherlands: Faculty of Technology, Policy and Management, 2003.
- Galloway, A.R. *Protocol: How Control Exists After Decentralization*. Cambridge, MA: MIT Press, 2004.
- Graham, S. S. "Analyzing the Discipline: Interventions in Rhetoric of Science and Communication." Presentation, Association for the Rhetoric of Science and Technology NCA Pre-Conference 2011: Communication: Energy, Inquiry, and Risk. New Orleans, 2011.
- Gross, A. G. "Rhetoric of Science is Epistemic Rhetoric." *Quarterly Journal of Speech* 76 (1990): 304–306.
- "The Roles of Rhetoric in the Public Understanding of Science." *Public Understanding of Science* 3 (1994): 3-23.
- "What if We're Not Producing Knowledge: Critical Reflections on the Rhetorical Criticism of Science." In A.G. Gross and W.M. Keith (Eds.), *Rhetorical Hermeneutics: Invention and Interpretation in the Age of Science* (138-155). Albany: NY: State University of New York Press, 1997.
- Starring the Text: The Place of Rhetoric in Science Studies*. Carbondale, IL: Southern Illinois University Press, 2006.
- "Toward a Theory of Verbal–Visual Interaction: The Example of Lavoisier." *Rhetoric Society Quarterly* 39 (2009): 147–169.
- Gustafson, A. "Beyond DOCTYPE: Web Standards, Forward Compatibility, and IE8." *A List Apart Magazine* 251 (2008). Retrieved from <http://www.alistapart.com/articles/beyonddoctype>

- Harris, R.A. "The Rhetoric of Science by Alan G. Gross." *Rhetoric Society Quarterly* 21 (1991): 32–35.
- Johnson, N. R. "Technical Documents as Rhetorical Agency." *Archival Science* 8 (2009): 200–214.
- How to Build an Infrastructure: Rhetorics of Web Standardization* (PhD Dissertation). University of Wisconsin-Madison, Madison, WI, 2011.
- "Information Infrastructure as Rhetoric: Tools for Analysis." *Poroi* 8 (2012): 1–3.
- Keith, J. (2008). "They Shoot Browsers, Don't They?" *A List Apart*, 253. Retrieved from <http://www.alistapart.com/articles/theyshootbrowsers/>
- Latour, B. *Science in Action: How to Follow Scientists and Engineers Through Society*. Cambridge, MA: Harvard University Press, 1987.
- Meyer, E. "From Switches to Targets: A Standardista's Journey." *A List Apart Magazine* 251 (2008). Retrieved from <http://www.alistapart.com/articles/fromswitchestotargets/>
- Mol, A. *The Body Multiple: Ontology in Medical Practice*. Durham, N.C.: Duke University Press, 2005.
- Olson, H.A. *The Power to Name: Locating the Limits of Subject Representation in Libraries*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 2002.
- Schivelbusch, W. *The Railway Journey: The Industrialization of Time and Space in the 19th Century*. Berkeley: University of California Press, 1986.
- Strauss, A., S. Fagerhaugh, B. Suczek, and C. Wiener. *Social Organization of Medical Work*. New Brunswick, NJ: Transaction Publishers, 1997.
- Zeldman, J. "Version Targeting: Threat or Menace?" *A List Apart* 253 (2008). Retrieved from <http://www.alistapart.com/articles/minorthreat/>