

1. A World without Email .....	2
1.1 Part 1: The Case Against Email .....	2
1.2 Part 2: Principles for a World without Email .....	4

# A World without Email

by Cal Newport (2021); [Amazon link](#)

**Keywords:** asynchronous messaging, attention, attention budget, automation, autonomy, capacity planning, design sprint, Extreme Programming, Frederik Taylor, Getting Things Done, human cognition, information asymmetry, information theory, interruptions, Kanban, knowledge work, locus of control, multitasking, office hours, outsourcing, Peter Drucker, process aversion, productivity, protocols, Scrum, short emails, social loafing, specialization vs. generalization, support vs. specialist, work execution, workflow, written vs. non-verbal

---

From Neil Postman's "Five Things We Need to Know about Technological Change"... "Technological change is not additive; it is ecological... A new medium does not add something; it changes everything. In the year 1500, after the printing press was invented, you did not have old Europe plus the printing press. You had a different Europe."

---

**Hyperactive Hive Mind** – workflow centered around conversation fueled by unstructured and unscheduled messages delivered digitally (e.g., email, Slack)

Issues with email and other asynchronous communication:

- This style of communication – volume, back-and-forth style, irrelevant information – induces the hyperactive hive mind.
- Our brains work better in serial than parallel.
- We're socially wired for 1:1 communication, or at best 1:few.
- Text-based communication dampens other communication channels we're hard-wired for.
- Because email reduces the friction of communication, we have induced demand to communicate even more.
- Delays are unpredictable and sometimes unmanageable / invisible.
- We build cycles of responsiveness (i.e., a fast response gets you more requests).
- Communication at scale is required in the modern workforce, yet our biological hardware is ill equipped to manage this well.
- Knowledge work requires autonomy; it also creates two streams of effort: the work, managing work.

Autonomy is required for **knowledge work** (i.e., tasks that can't easily be broken down into to sequences followed without thinking). Most people focus on work execution; managers need to focus on **workflow** – how work is identified, assigned, coordinated, and reviewed (i.e., the structure of the work).

Limit what you commit to, and deliver on what you do commit to.

Combining several available tools such as **Kanban** (Trello) and **Getting Things Done** help manage requests and make information visible to requesters without endless emails.

---

## Principles

1. **Attention Capital** – The productivity of the knowledge sector can be significantly increased if we identify workflows that better optimize the human brain's ability to sustainably add value to information.
2. **Process** – Introducing smart production processes to knowledge work can dramatically increase performance and make the work much less draining.
3. **Protocol** – Designing rules that optimize when and how coordination occurs in the workplace is a pain in the short term but can result in significantly more productive operation in the long term.
4. **Specialization** – In the knowledge sector, working on fewer things, but doing each thing with more quality and accountability, can be the foundation for significantly more productivity.

- [Part 1: The Case Against Email](#)
- [Part 2: Principles for a World without Email](#)

## Part 1: The Case Against Email

### Introduction

- Example case: Nish Acharyna (director of innovation and entrepreneurship appointed by Obama) had his network effectively shut down and couldn't use email because of a virus. This is a **forcing function**.
- Constant communication is how work gets done, making simple efforts to reduce distraction either short-lived or likely to be seen as stunts (e.g., no-email Friday).
- In the 1980s and 1990s, email offered *low-friction communication at scale*.
- **Hyperactive Hive Mind** – workflow centered around conversation fueled by unstructured and unscheduled messages delivered digitally (e.g., email, Slack)
- Knowledge workers spend their day tending to ongoing hive mind communication (e.g., checking inboxes every 6 minutes).

- We're hardwired to be tribal/social, so it feels odd to ignore requests for our attention. Modern tools + ancient brains.
- This book isn't about Luddism or anything anti-technology; it's about understanding the hyperactive hive mind so that our technology doesn't hinder us.

## Email Reduces Productivity

- The hidden costs of the hyperactive mind
  - Author surveyed 1500+ people about the role of email/chat
  - Issues: sheer volume, endless back-and-forth, increased amount of irrelevant information
- Constant multitasking craziness
  - We no longer have to sit in the same room to get work done (asynchronous communication).
  - Knowledge workers focus attention on...
    - Executing tasks
    - Managing conversation about said tasks
- The sequential brain in a parallel world
  - Adam Gazzaley and Larry Rosen in *The Distracted Mind*: "Our brains do not parallel process information."
  - Your brain needs time and cognitive resources to switch between tasks.
  - Our brains were not designed for these parallel work streams.
- Email is not a job
  - Paul Graham wrote an essay (2009) "Maker's Schedule, Manager's Schedule". Managers go to all the meetings, and makers should attend few/no meetings. The author thinks this distinction is too crude; managers would also benefit from less distraction.
  - George Marshall, US Army chief; 30 major and 350 minor commands under his control
    - He delegated much of the day-to-day operations to others, freeing up his time.
    - He trusted his team to execute decisions he made without involving him in the details.
  - "Boxed In by Your Inbox" in *2019 The Journal of Applied Psychology*... Interruptions affect managers as well, such that when they feel overwhelmed, they end up mostly focusing on small tasks and responding to queries instead of more important things like big-picture work.
  - **Minders** – roles that provide administrative or logistical support in knowledge work orgs
    - Many people argue that having a hyperactive hive mind is central to minders' usefulness.
    - Counterexample: IT department that was very responsive spent more time fielding communication *about work* than *doing work*. Solution: Ticketing system.
    - Minders are not automatons – they think as well, which is corrupted by too many distractions.
  - Example: CEO explained that there would be a two-hour segment of each day where engineers were unreachable (so they could get work done). From Geoff: This probably worked because it came from the CEO; titles matter sometimes.
- Beyond the hive mind
  - You can't just kill email and chat; you need something to replace functions they serve.
  - Example: Having in-person daily standups instead of emailing people for status updates.

## Email Makes Us Miserable

- An epidemic of silent suffering
  - A commonly suggested solution of only checking email at certain times (batching) doesn't make it less stressful, especially for those that score highly on the OCEAN's *neuroticism* trait (worrying about urgent messages you're ignoring).
  - When stressed, people answer emails faster, but not better.
  - Harvard Business School professor Leslie Perlow experimented with **predictable time off** where teams were unavailable by email. The results were generally positive.
  - "We depend on email, but we also kind of hate it." This misery makes us perform worse.
- Email scrambles our ancient social drives
  - "The drive to interact with others is one of the strongest motivational forces humans experience." This is also what has helped us succeed as a species.
  - We feel distress when 1:1 interaction is neglected.
  - How do you explain to your brain that neglected emails aren't pivotal to our survival?
  - One solution from "Thrive Away" (Arianna Huffington): Auto-responder that tells the sender their email has been deleted. If it's important, resend when the receiver is available.
- Email communication is frustratingly ineffective
  - We're now making decisions using **writing**. Again, our hardware wasn't optimized for this; writing is only about 5,000 years old.
  - From Alex Pentland's *Honest Signals: How They Shape Our World*... Information is processed largely **unconsciously**.
    - Influence – degree that another can match a speaking pattern
    - Activity – physical movements during a conversation

- Study: reading business plans in writing vs. hearing them presented. (The book didn't seem to indicate which version was better, just that they were *different*.)
- We overestimate our correspondents' ability to understand our messages. Fun example: pick the US national anthem or "Happy Birthday", then tap out the rhythm to someone else to see if they can discern which tune you picked.
- Email creates more work
  - Because email has less friction, we're using it to ask more questions and delegate more tasks than before, leading to more overload.
  - David Allen's *Getting Things Done* book was likely so appealing because it provided a concrete answer to dealing with the increasing amount of work dumped on our plates.
  - Sometimes adding friction to email will reduce the number of requests; too little friction can lead to feedback loops.
  - ? What if you have remote colleagues where you don't have the option/friction to go over and talk to them?
  - "When we made communication free, we accidentally triggered a massive increase in our relative workloads."

## Email Has a Mind of Its Own

- The rise of email
  - This book is more about **asynchronous messaging** in general, where the receiver doesn't have to be present.
  - Having physical messages (e.g., internal mail cart) is impractical for efficiently coordinating and sharing time-sensitive info. Pneumatic tubes made this process slightly faster.
  - Email could have simplified or made **one part** of office work more efficient, so what did it become universal?
- What does technology want?
  - When IBM introduced email as a way to communicate more quickly, it overloaded the initial setup within a few days because people used it more than the designers predicted. One theory is that this new tool allowed CC'ing emails. (Remember that notes left on a desk are 1:1, not 1:many.)
  - The Like button on Facebook was designed to reduce the number of comments on posts (which mostly said, "cool", "nice"). Instead it ended up being a **social approval indicator** with all kinds of downstream effects.
- Stumbling into the hive mind
  - Driver 1: Hidden costs of asynchrony
    - Email helped deal with the growing size of offices.
    - The problem with asynchrony is you don't know when the message will be read.
    - Delays are unpredictable.
    - "While the business world came to see synchrony as an obstacle to overcome, computer theorists began to realize it was fundamental for effective communication." People are not computers.
    - Synchrony is expensive, but so is trying to coordinate in its absence; this leads to the hyperactive hive mind workflow.
  - Driver 2: Cycle of responsiveness (from Perlow's research)
    - It starts with legitimate demands of your time.
    - You respond quickly and solve their problem, which leads them to make more requests.
    - Now you check your phone/email more frequently to continue this level of responsiveness.
  - Driver 3: Caveman at the computer screen
    - Source: archeologists Aviad Agam and Ran Barkai
    - The most natural way for small groups to coordinate is in a free-form manner (ad hoc, unstructured, adjusting to the current needs).
    - Free-form works great for six people, but is woefully ineffective for dozens of people or large organizations.
    - See also: Maximilian Ringelmann's rope-pulling experiment where the average force exerted by each person decreases as group sizes grow **loafing effect** (the more people working on something, the easier it is to get away with putting in less effort)
    - Large military groups usually use strict chains of command because free-form communication devolves quickly at scale.
- Peter Drucker and the tragedy of the attention commons
  - Drucker was one of the first to study how big organizations operated.
  - The idea of not focusing too much on micromanaging how a worker gets their job done came from Drucker working with GM CEO Alfred P. Sloan in the 1940s.
  - This was in opposition to Frederick Taylor's "scientific management" where you followed workers around with stopwatches, and treated workers as automatons executing the skillful plans of the managers.
  - Drucker coined the term **knowledge work**, where the new productive output was **cerebral** rather than physical. Most often the workers knew more about their specialities than their managers did. "Knowledge workers have to manage themselves. They have to have autonomy."
  - The burden of autonomy is that "...it's hard to fix a broken workflow when it's no one's job to make sure the workflow functions."
  - Tragedy of the commons – it's to our own individual interest to be hyperactive, yet the system suffers for it

## Part 2: Principles for a World without Email

## The Attention Capital Principle

The productivity of the knowledge sector can be significantly increased if we identify workflows that better optimize the human brain's ability to sustainably add value to information.

- On Model Ts and knowledge work
  - Cars used to be constructed using the “craft method” – workmen brought parts to the assembly area, like how you'd build a house.
  - Over time, tools became more precise and interchangeable so that parts fit together more easily and required less custom fitting, sanding, etc. This led to another approach: bring the cars to the people (assembly line).
  - Example of experimentation: Lasse Reheingans (German entrepreneur) introduced a 5-hour workday in his company in response to them “running all the time without getting anywhere” (talking about work rather than doing it).
- Case study
  - An entrepreneur switched from email to Trello to deal with task status and management.
  - The boards allowed teams to focus on one thing until they were ready to move on.
- Build structures around autonomy
  - Peter Drucker realized that knowledge work **couldn't easily be broken down** into a series of repetitive tasks that could be prescribed by managers (like with manual labor) and followed unthinkingly. Ex: developing a business strategy, innovating a process.
  - Autonomy is necessary because the work is too complex. However, autonomy requires coordination (i.e., hyperactive hive mind).
  - Two components of knowledge work
    - **Work execution** – executing the underlying value-producing activities (e.g., coding, writing)
    - **Workflow** – how work is identified, assigned, coordinated, and reviewed (i.e., the structure of the work)
  - Drucker meant for autonomy to be on the *execution* side, and that others should intentionally define workflow, as the most effective systems are unlikely to arise naturally. Ex: don't tell developers how to code, but help them prioritize what they'll be coding.
- Minimize context switches and overload
  - Moving from task to task with interruptions has a large cognitive cost.
  - Find ways to (1) minimize mid-task switches, and (2) minimize communication overload.
- Don't fear inconvenience
  - The hyperactive hive mind persists because it's *convenient in the moment* – no systems to learn, rules to remember – just message people when you want something.
  - **Any** change will be inconvenient, and many efforts (e.g., communication etiquette) usually fall flat because it's difficult to go against the status quo.
  - Henry Ford (switching from craft method to assembly line) took a reliable and intuitive process for building cars and replaced it with something more expensive to run, required more managers and overhead, wasn't initially intuitive, and broken down frequently in the beginning.
  - Typically innovation is left to products and services instead of **the actual mechanics of how work is assigned, executed, and reviewed**.
- An aside: weren't assembly lines awful for workers?
  - Assembly lines were not a cure-all. The work was relentless and repetitious.
  - See also: the factory scene from *Modern Times* (<https://www.youtube.com/watch?v=ANXGJe6i3G8>)
- When implementing changes, seek **partners**, not **forgiveness**
  - Find ways to get your employees to change the way they work. For example, there may be things you as a manager can delegate.
  - **Locus of control theory** – motivation is closely connected to whether people feel like they have control over their ultimate success in a task
  - Have the team that will change the way they work design the new way of working.
    - Make sure people understand the difference between execution and workflow.
    - Get buy-in from the people having to actually use the workflow.
    - Make it easy to change the workflow when issues arise.
  - There was a counterexample from Tim Ferris where he tried to solve the email overload problem by having an autoresponder tell people how he handles email (e.g., only reading at 12pm daily). This came across as condescending, and ultimately pushed the problem back to the sender instead of addressing the broken system.
  - Limit what you commit to, and deliver on what you do commit to. “If people trust you to handle work they send your way, then they're generally fine with not hearing back from you right away.”
  - Another solution: implement queues via a ticketing system, and make it have a seamless interface to get things into it (e.g., email address that creates new tickets).
  - (Note from Geoff: The author describes his productivity system, which is very similar mine – Getting Things Done implemented with Trello.)
  - People don't like changes they can't control.

## The Process Principle

Introducing smart production processes to knowledge work can dramatically increase performance and make the work much less draining.

- The power of process
  - “We largely ignore processes, investing our energy instead into figuring out how to make people faster. We obsess over hiring and promoting stars. We seek leadership consultants to help us motivate people to work longer and harder. We embrace innovations like the smartphone that allow more hours of the day to be punctuated with work. ... Not surprisingly, this hasn’t worked out well at all.”
  - It’s not just about efforts; those efforts need coordination.
  - The expense of overhead is worthwhile because it helps clarify and optimize the work.
  - Why are we so averse to process? By definition, rules about how work happens reduce autonomy. “There’s a belief, implicitly held by many knowledge workers, that the lack of processes in this sector is not just an unavoidable side effect of self-management, but actually a *smart* way to work. A lack of processes, it’s commonly understood, represents nimbleness and flexibility – a foundation for the type of outside-the-box thinking we’re constantly told is critical.”
  - Like Thomas Hobbes (from *Leviathan*), the structures that emerge naturally and informally are not utopian. The aggressive people **impose dominance hierarchies** and the agreeable people **get overwhelmed** by saying yes to too much. These hierarchies – not conducive to knowledge work – are unavoidable unless you have a countervailing force.
  - Without structure, human nature is to **minimize energy** (social loafing) in coordinating efforts and we instead rely on instinct. We find the easiest way to get the work off our plate.
- Case study: optimizing the optimizers
  - The founder of Optimize Enterprises (media production company) went through an exercise to break down work into processes that could be clearly stated so that more time was spent on useful work than managing inboxes.
  - They built a shared spreadsheet that showed status and updates. They eventually moved to Flow ([getflow.com](http://getflow.com)) and daily standups. Agile fundamentals helped them improve existing processes.
- Cards in columns: the task board revolution
  - Tools like Trello, Asana, and Jira allow you to organize work into columns that denote an item’s status.
  - This idea is not new – hospital ERs have used tracking boards to know about patients and nurses.
  - The book goes on to describe how traditional project management approaches (e.g., Gantt charts) for simple projects. More complex and longer-term projects have not proved to be useful applications of these techniques. Plus there’s added uncertainty in what the customer wants from the product you’re building.
  - “A key idea driving agile project management is that humans are naturally pretty good at planning. You don’t need complicated project management strategies to figure out what to work on next; it’s usually sufficient to just have a group of informed engineers get together and discuss what makes sense.”
  - Best practices
    - Cards should be clear and informative
    - When in doubt, have the columns be To Do / Doing / Done
    - Hold regular review meetings to increase accountability using real-time communication
    - Use the commenting/conversation functionality to replace hive mind chatter
- Personal Kanban: organizing your professional life with individual task boards. This is basically Getting Things Done by David Allen; the book describes how one would set this up.
- A follows B: automate processes
  - “Not all processes, however, can be made automatic. For this strategy to apply, the process in question must produce some output in a highly repeatable fashions, where the same steps are implemented, in the same order, by the same people, each time.”
  - Steps
    - **Partition** processes into discrete sequential states
    - **Signal** when states change so the next action can occur
    - **Channel** information closer to the context where work is being done
  - Rory Vaden’s 30x rule: “You should spend 30x the amount of time training someone to do a task than it would take you do that task yourself one time.” In other words, if you do something 30x a year and it’s automatable, it’s likely worthwhile to do so.

## The Protocol Principle

Designing rules that optimize when and how coordination occurs in the workplace is a pain in the short term but can result in significantly more productive operation in the long term.

- The invention of information
  - Claude Shannon studied **information theory**; by adding complexity to the rules we use to structure information, the amount of information can be reduced. Example: the simplest letters in Morse code are the ones that are most frequent.
  - In terms of *cognitive cycles* (say, the number of 5-minute intervals in a given week that are interrupted to deal with an issue), how costly is your protocol?

- In terms of *inconvenience or other cost* (e.g., cost of delay by waiting until Friday at noon to answer client requests), how costly is your protocol?
- Hybrid approach: Have someone watch for inbound client requests and reply promptly indicating receipt and next expectations, and wait until Friday to discuss in detail. This has the best **average cost**.
- Meeting schedule protocols
  - Very common practice: several rounds of ping-pong emails to find a mutually agreeable time to meet
  - There are tools (e.g., Calendly) that let hosts define availabilities, clients pick from those, and then sets up the meeting itself.
- Office hour protocols
  - Some companies have tried setting up guaranteed times for responses via **office hours**.
  - This is difficult to arrange when your work spans multiple time zones.
  - Basecamp used office hours so that experts could be available to solve problems. This solved their hyperactive hive mind problem, but seems to work best for activities that are **not too negatively impacted by delays in responses**.
- Client protocols
  - Sometimes clients can ask you to do things at any moment. An example company in the book addressed this by adding a "Communication" section to every statement of work. This section spelled out the rules for both the client and the company (e.g., standing meetings, how notes would be shared, how to deal with urgent matters).
  - Clients probably don't like sending you endless messages. They want to feel in control or to at least know what's going on.
  - Having the protocol spelled out in the contract makes both parties take it more seriously.
  - We also need to be okay with the notion of firing customers because they break these protocols.
- Non-personal email protocols
  - The book proposes having a project-oriented email address instead of individuals. (From Geoff: The downside is that now *everyone* in that group distribution list gets pinged when someone contacts it, unless the group is managed by a single person.)
  - The author set up an inbox (interesting@calnewport.com) with an auto-responder saying that he's thankful for the input, but may not respond.
- Short message protocols
  - Idea: Treat emails like text messages.
  - <http://five.sentenc.es>
- Status meeting protocols
  - This section basically describes Scrum, where you have daily standup meetings that are short and focused.
  - The premise is to have more frequent meetings that are of higher quality.

## The Specialization Principle

In the knowledge sector, working on fewer things, but doing each thing with more quality and accountability, can be the foundation for significantly more productivity.

- A productivity puzzle
  - "Why did the arrival of personal computers in the workplace fail to make us as productive as we predicted?" Edward Tenner (author of *Why Things Bite Back*) claims that instead of reducing labor, computers end up creating more work. They are more difficult to understand. The technology changes every few years, meaning there's even more to learn.
  - Economist Peter G. Sassone studied the shift of more specialized people (e.g., lawyers) taking on more administrative work done via computer, with the notion of not needing so many support staff positions. This however made the lawyers less effective because they weren't focusing on their specialty, and were essentially diluted.
  - "The sheer quantity and variety of tasks in a non-specialized work environment make the hive mind workflow *unavoidable*." (From Geoff: This is also called **full-stack** in developer parlance.) You don't have enough time to think about better workflows because you're too busy handling the incoming work requests.
  - "There are few things more valuable than someone who consistently produces valuable output, and few approaches to work more satisfying than being given the room to focus on things that really matter."
- Case study: working at the extremes
  - This section gives an overview of extreme programming ([https://en.wikipedia.org/wiki/Extreme\\_programming](https://en.wikipedia.org/wiki/Extreme_programming)).
  - Some topics: pair programming, minimizing disruptions by having the product manager be the liaison between the customer and the developers.
  - From Geoff: Some disadvantages...
    - The developers become even more removed from the context in which they're developing.
    - Specialists can focus, but you still require **multiple specialists** to deliver, meaning that they all depend on one another to complete work.
- Do less, do better
  - Strategy 1: outsource what you don't do well
    - For example, you may need marketing to be successful as a business, but if you focus too much on that, you won't spend time on why you're in business in the first place. Therefore, use an external firm to manage this.
    - You will pay for this in the short term, but should see longer-term benefits.
    - (From Geoff: The downside is that you lose touch on how other parts of the work gets done.)

- Some things may be worth automating so that knowledge workers can use their talents elsewhere. For example, let computer systems grade school worksheets (instead of teachers) so that teachers can think about better lessons.
- Strategy 2: trade accountability for autonomy
  - Busyness is controllable; it's easy to be visibly busy. This sometimes lets people get complacent on the harder tasks.
  - The example in the book is where an associate asked her manager to not field random questions, and to instead be left alone to work on strategy. (From Geoff: There's an assumption that people are poised to work this way and just need someone to give them permission to work on big things.)
  - "If you spend more time on the high-value activities at the expense of spending less time on the low-value activities, you'll produce more value overall. (From Geoff: True, but don't conflate low-value with no-value.)"
- Sprint, don't wander
  - The author talked about a **design sprint** mentioned in *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. The concept is that you get people in a room for one week and essentially isolate them. People are there to concentrate on a single, important problem.
  - (From Geoff: This concept assumes you have every dependency under control (i.e., everyone you need to answer any bottlenecking question in the room), and that the only reason you haven't done this before is because you have too many interruptions.)
  - "If you work for a larger organization, enthusiasm for sprints must emanate from the top." (From Geoff: Totally agree; things too frequently never take off without management support.)
- Budget attention
  - There is **asymmetry** when you ask someone for help. You may not think asking someone to give 30 minutes a week on a committee seems like much; however, you don't know *what else* is already on their plate.
  - Problem: **small requests accumulate**. "...suddenly we're desperately overwhelmed with work that has little to do with our main objectives... To say no to any one of these requests in isolation makes you seem curmudgeonly or lazy."
  - However, you likely still need these ancillary projects/committees – they serve some purpose. Similarly, experts can't refuse to answer people's questions and focus only on their work.
  - Solution: **capacity planning** (a.k.a. attention budget)
    - Premise: Your time and attention are limited.
    - Quantify how much time/attention is going to what kinds of work.
    - Before taking on new work, the requester (which may be yourself!) must reconcile their request with your current commitment.
  - Having a quota/limit/budget gives you an understandable response: "I wish I could help, but I'm already on three committees." Someone would have to counter with "You should be on more committees," which is difficult to defend.
  - Think about the **ratio of deep-to-shallow work** you do.
- Supercharge support
  - You probably need more support staff, instead of asking knowledge workers to do more.
  - "Most modern knowledge work organizations treat individuals as general-purpose computers that execute a turbulent mixture of value-producing and administrative tasks – often unequally distributed, and not at all optimized for any particular big picture objective."
  - Structure support
    - For example, it may be slower to walk a form down the hall, but by the time you get back, the job is done; synchronous but predictable. (From Geoff: This is also called **workflow efficiency** – how much time work is actively being processed, that is, not sitting in a holding pattern.)
  - Build better interfaces between support and specialists
    - The goal here is to clearly define what the specialists do and how they get support.
    - The author points out an ethical pitfall of inadvertently creating an **underclass** of support professionals confined to overload. Protect a culture of "support people are essential." There's also economic arguments that support people optimizing the specialist's higher-return work.
  - Last resort: simulate your own support staff
    - Partition your time into support and specialty (e.g., support during 12-1pm and 3-5pm). You could also dedicate certain days to admin tasks.

## Conclusion

- From Neil Postman's "Five Things We Need to Know about Technological Change"... "Technological change is not additive; it is ecological... A new medium does not add something; it changes everything. In the year 1500, after the printing press was invented, you did not have old Europe plus the printing press. You had a different Europe."
- We don't have 1991 office plus email; we have a hyperactive hive mind to deal with never-ending, ad hoc, unstructured messages.
- "Knowledge worker productivity is the moonshot of the twenty-first century."
- Digital-era work is on a historical scale a recent phenomenon. It's short-sighted to think the easy workflows we threw together are somehow the best way to work.