An Operational and Policy Information Technology Brief: Dimensions of Merit and Demerit Regarding "Does IT Matter" by Dr. N.G. Carr

Submitted in partial fulfillment
of the requirements of
CSE P 590TU
Information Technology & Public Policy (PMP)
Department of Computer Science & Engineering
The University of Washington

December 10, 2004

By

Diana Bullion
Erik Jansen
Joanna Muench

Executive Summary:

Harvard's Dr. Nicholas G. Carr recently published a book laying out his provocative and controversial thesis, "Does IT Matter?". In it, he asserted that the strategic and economic advantages often associated with (and even conventionally expected from) the business deployment of information technologies has faded as a direct consequence of their being broadly available to all as a commodity. He consequently recommends that managers view investments in information technologies as being a source of more downside financial risk than upside strategic opportunity. His work has sparked a broad (and often intense) debate as to its truths, mistruths and misconceptions.

In this brief, we will:

1. Provide a basic backdrop summary of Carr's key points and assumptions, focusing upon:
    a. putting his advice within a context that makes it valuable and useful to today's information technology managers; and,
    b. identifying those points that appear to us to be more nebulous, incoherent (based upon miscast facts or premises) or are most often misinterpreted by his detractors;
2. Develop and test several alternative views regarding the deployment of information technologies in a business environment, including:
    a. identifying portions of the broad range of Modern Information Technologies (we will call it MIT) that can be classified as a classic commodity;
    b. putting forth an approach to help understand those that do not fit that category;
    c. calling out those areas of MIT that defy commodity categorization (especially in larger scale and scope MIT-based deployments);
3. Provide a focused and more grounded interpretation of what Carr's (and our) conclusions practically mean to a cognizant business executive who must make concrete and timely MIT investment decisions. We will test conclusions that

logically flow from Carr's thesis, including assertions that a broadly available and adopted (and thereby well understood) MIT process and product portfolio should:

   a. exhibit increased rates of deployment success (lower costs, more timely completion, broad organizational acceptance, etc.) over time;
   b. stimulate the expanded acceptance of standard professional practices and certifications;
   c. offer only spurious cases of systemic strategic upside opportunity; and finally,

4. Put forth some additional thoughts as to the impacts of these and Carr's arguments in formulating proactive and positive public policy.

We conclude that Carr's base advice to executives (to view information technology-based investments conservatively and with an eye toward minimizing downside risk) to be consistent with 'common business sense' and normal scrutiny applied to major capital investment expenditures. However, we also find that several of Carr's specific conclusions and basic premises are overly simplified or subject to misinterpretation (including his unfortunate choice of such an aggressive title). While his conclusions and advice do tend to match many common truths and experiences held by today's MIT executives, we would warn against accepting them at face value or as the prime predictors of "best practices" for organizational use of MIT.

Much of this corroboration with executive experience may not be so perfect or complete: it may also be blissful coincidence. The overly simple definition of Carr's Information Technology (we will refer to it as CIT to avoid confusion with MIT) -- he excludes all people, processes, data, policies and politics from his literal and limited "hardware+software only" definition – makes his conclusions quite difficult to apply in a real world environment. We believe he simply ignores too much of the true nature of MIT's poorly controlled complexity, problematic social and training issues and product interdependence (undocumented or unexpected interactions). Carr's overly simplified context of IT (CIT) allows him to fall into a chasm of causal confusion: while carefully developing a research-based platform for his point that CIT has generally failed to deliver positive results, he jumps to the conclusion this is due to a commoditized state. He fails to consider an alternative explanation for failures that we see as closer to the root cause: the exceedingly complicated and closed nature of many IT products has

made their smooth deployment over time essentially an impossible task to manage in dynamic business contexts. Curiously, in blissfully falling into his own trap, Carr misses the chance to call out the direct, positive and potentially systemic opportunities this challenge leaves for the most capable of consistent MIT users!

In a related analysis, we found Carr's reference to CIT as a commodity fails to pass muster. Our analysis indicates much of applied MIT simply does not exhibit these commodity attributes in practical business use. This is especially exacerbated in large scale and scope deployments.

Further, while Carr develops and extrapolates heavily from several well-understood, historical, network-based resource analogies – rail, electricity and the telegraph – we believe that the most important (and troublesome) attributes of MIT are not comprehended in these templates. Most notably, these include the "locking-in" that many standard (and *not* commoditized) MIT products subject users to, the vastly differing rates in which some portions of MIT are advancing versus others and the generally poor state-of-the-art for MIT tools that aid managing complexity or migrating different platforms one from another. While Carr correctly identifies the troubling trend of MIT products' often dictating organizational process upon users (a highly dysfunctional outcome for most users), he fails to consider how this may buck powerful social trends increasingly expecting otherwise.

As for guidance for MIT-managing executives, we believe that Carr's generic advice closely matches current-day MIT common business sense:

1. spend less or more slowly; a
2. avoid "first mover" risks; and,
3. focus management's attention primarily on avoiding downside mistakes in deployment.

However, just as indistinguishable products beget commodities, so identical behaviors and processes beget lackluster (non-existent) business opportunities. Carr acknowledges this by advocating the possibility that certain (unnamed) businesses or industries may need to adopt alternative decision-making priorities – but his admission is without sufficient self-deprecation or development: he waits until the last two dozen lines of the book to state as much. As such, we would advise a very different generic management approach: focus first on identifying distinctive ways MIT can amplify advantage. This is most often be done by pairing MIT with

other unique assets, retain the best deployment and management talent, parlaying strategies to transform MIT-derived success into other non-MIT domains (such as brands, scale, customer centricity for example) -- and then and only then, as a secondary effort, screen these prime and appropriate opportunities through Carr's three-point sieve.

In making suggestions as to the best deployment practices of MIT in a business environment, we focus on four concepts that are central to Carr's arguments. We:

1. question Carr's limited definition of what IT truly is;
2. investigate whether, even under Carr's limited definition, IT is truly available and affordable to all;
3. analyze if the commodity label can be applied equally to all types of CIT, or if some types of IT are, by their very nature, ill suited to a commodity label; and, finally,
4. investigate if, as Carr asserts, best practices for any given component of IT are established and broadly disseminated in today's business application realm.

Our analysis finds that none of these premises hold up under all circumstances. This is a serious undermining to the validity of Carr's conclusions. The reasons for their failure vary, but in general are a function of a very few key facts that Carr underplays the:

- economic network effects that play a very significant role in all of these areas, keeping prices higher than they might otherwise settle out to in a classically competitive market place;
- difficulty in which CIT (or MIT!) systems scale up to broader deployment (application complexity and interactions become very difficult to robustly manage); and, finally
- significant costs that accrue from interactions between humans and MIT in creating business processes that work (he simply removes all social issues from his definition CIT!). Theses social costs of integrating MIT into business practice are far from being fully understood and remain a ripe source of research and profound discovery.

While his conclusions are flawed for generalized MIT use, we find Carr raises many good points about the use of MIT in some businesses. Many smaller MIT systems are essentially standard configurations of commonly available, relatively simple products. They can, at a first level of security and risk, be easily implemented as if they were a true commodity. If simple businesses can work around some of the limitations due to network effects and avoid

overshooting by their suppliers, they can maintain capable, relative simple and cost effective MIT solutions.

We conclude with a discussion of how policy makers may need to comprehend these factors in evaluating various forms of alternative (or complementary) public policy to better benefit society. The first of these includes a description of a new, hybrid form of MIT IP protection we would prescribe. Briefly, this new legal IP category or tool:

- draws upon the value of public disclosure as present day patent protection requires;
- has literal protection akin to a copyright, but not quite so literally;
- has a life that is variable and dynamic (not fixed) and would be modulated by the state of a networked market as it is near a tipping point; and
- may be either integrated into existing antitrust law (as an interpreted condition of a market demanding supplier compliance) or relied upon as a separate and voluntary protection mechanism, available to and preserving the rights of business to use it as a tool in developing its strategy for creating new products and markets.

In addition, we briefly expose as folly the concept of attempting to use policy tools to accelerate or otherwise regulate the (natural and necessary) creation of standards. This may sometimes erroneously be thought of as a way to avoid proprietary or monopolistic exploitation of new standards before they become addressable by (available to) all.

Finally, we issue a warning of the risks embedded within the "Trusted Computing" (TC) paradigms that are beginning to arrive in the MIT marketplace. Whether pronounced as a panacea for many of the ills affecting the MIT industry, TC will force a political face off placing suppliers' desires directly in opposition to the safety and sanctity of users' privacy and rights to best use what is rightfully theirs. This will be a powerful debate that will likely create a whole new level if industry (and user) regulations and legal warfare. Mindful executives will embrace such supplier initiatives with care and balance – perhaps as Carr originally recommends, delaying, deferring and minimizing its use within while being mindful of unknown downsides.

Carr's Seminal Starting point – What is 'IT'?

Carr's basic thesis is this:

> "You gain an edge over rivals by having something or doing something that they cannot do. By now, the core functions of IT – data storage, data processing and data transport – have become available and affordable to all. Information technology's very power and presence have begun to transform it from a

> potentially strategic resource into what economists call a commodity input, a cost of doing business that must be paid by all but provides distinction to none" [1]

Seminal to Carr's thesis is his unique and quite specific definition if IT (which we will subsequently refer to as "CIT" to maintain a clear distinction from the more popular view of Modern Information Technology, which we will refer to as "MIT" and define momentarily as including all IT-related resources and costs):

> "...denoting all the technology, both hardware and software, used to store, process, and transport information in digital form. It is important to stress that I am talking about the technology itself. The meaning of 'IT' does not encompass the information that flows through the technology or the talent of the people using the technology."[2]

We (and many of his detractors[3]) find this view incomplete and overly simplified. Inasmuch as technology alone has no intrinsic value – it must necessarily be deployed and used by people through a real process having impact on real assets (tangible or intangible) before it can create value – Carr's work, while intellectually engaging, necessarily suffers from very limited practical application or interpretation in the end. While Carr's gross simplification of the fundamentally complex state of affairs in MIT today *does* allow him to set forth some simple historical analogies to buttress his view of the maturation of the CIT, the very narrowness of his starting premises similarly makes the extensibility of them a nearly pointless endeavor. Perhaps Smith and Fingar best described Carr's tradeoff here in their work in response to Carr's approach:

> "While Carr serves as a lightning rod and shocks people into thinking about IT, after reading the article, we ask business leaders to *think again.* Although Carr's article embodies several 'individual truths,' his assumptions, premises and conclusions all merit closer examination if, as in the Indian story of the Blind Man and the Elephant, the whole picture of IT and business strategy are to come into clear focus. ...By restricting his definition, he looks only at parts of the

[1] "Does IT Matter", Nicholas G. Carr. Harvard Business School Press, Boston, Massachusetts, 2004. pp. 7.
[2] Ibid. pp. xii.
[3] For a collection of critical responses to Carr's original 2003 Harvard Business review article, "IT Doesn't Matter", see "Does IT Matter? An HBR Debate", Letters to the Editor, Thomas A. Stewart, Harvard Business Review, June 2003, pp. 1-16.

> elephant, and therein lays the grave danger for business leaders who read it and draw conclusions for the whole of IT."[4]

At its best, Carr's work has likely stimulated a valuable commercial discourse as to the effectiveness or potential of MIT. We see this as a noble goal and a healthy process -- unfortunately, however, his zealous assertion of it meaning more than that more is not. The principal danger here is that readers solely of Carr's work are left without any awareness of its true impertinence. Our efforts will focus on surfacing the concepts and supporting facts to buffer and modify his simple conclusions against the reality of Modern Information Technology (MIT). Hopefully, the result will be closer to representative of the current state of affairs and best practices and more appropriate for integration into executive intuition and actions. Repeating, by MIT, we mean Carr's CIT *plus* the talent, expertise, information and processes that make up the application, management and downstream sharing of the products of information technology use today.

Carr further restricts the domain of his discourse to developed world business use of CIT[5] (excluding the emerging markets of the world and consumer and home-related realms). We argue that both of these areas are having profound impacts on the development of next generation MIT and expanding the potential of MIT in the process. In our MIT nomenclature, we call out no such market or geographic exclusions. As Smith and Fingar explain it,

> "The careful reader will demand to know not only where [Carr] is right, but also what he has omitted – that is, the *full* scope of IT as it relates to competitive advantage. After all, an elephant's tail is not an elephant."[6]

Unfortunately, even this level of disclaim may not be sufficient to protect Carr's readers from inappropriate detailed conclusions. Taking many of Carr's points at face value can be dangerous. It appears that in at least a few occasions his casting of third party research in support of the points appears to be questionable. As an example, early in his development of the importance of CIT in business, he cites data from the U.S. Department of Commerce's Bureau of Economic Analysis describing a dramatic escalation of IT expenditures as a percentage of

[4] "IT doesn't Matter – Business Processes Do", Howard Smith and Peter Fingar. Meghan-Kiffer Press, Tampa, Florida, 2003. p. 19.
[5] Ibid., "Does IT Matter", Carr. p. xiii.
[6] Ibid., Smith and Fingar, p.19.

overall business capital spending from the period 1965 to the turn of the century.[7] We do not dispute the rising intensity of spending on MIT. However, Carr uses this data (defined as including "gear, software and services" – the latter two clearly being outside scope of his definition of CIT) to drive his point that CIT is expanding in importance. Because he mixes contexts, it necessarily raises suspicions that his point is necessarily a logical truth. This style of argument places Carr's work on a dangerous path from the start. We have been unable to find another significant research study that defines IT as narrowly – or dysfunctionally – as does Carr in his framing of CIT. We would speculate that it is due to this solitary starting point that he had little practical choice but occasionally miscast the available data to try to make his desired point.

We found other examples in Carr's text where he suspiciously allies external research or capricious uses CIT-based analysis to make MIT-related conclusions. Examples include:

- Carr ignores the real case of how MIT is penetrating (or arguable morphing and emerging from within) the borders of the controlled economy of China. This phenomenon is not new and significantly predates the publication of Carr's work. To ignore this as not being germane to the way the MIT users will manage their products and strategies or react to policy is sheer folly. Penetrating the Chinese market and making its users pay for their software (rather than violating copyright or other IP laws in pirating it) has been described as Bill Gate's passion.[8]
- Carr's exclusion of the consumer segment of commerce essentially invalidates from the onset large dimensions of any discussion of policy he would assert later in the work (which he does). After all, policy is not targeted at the improvement of corporate profits (or even the augmentation of equity values in pensioneers' accounts) but rather, raising the good standing of all of society. Carr's foundation places his policy assertions as unsupported cantilever arms.
- Carr totally ignores the element of scale in his CIT analysis. We believe that the CIT products purchased by typically smaller businesses vary dramatically (in mix, function, term of use and price) from those incrementally used in larger commercial infrastructure counterparts. This is true in a CIT-context as it is in an MIT- one). These differences are explored later in this text alongside Carr's highly flawed application of the concept of a commodity to many CIT products.
- In his original 2003 article, Carr develops three industrial analogies to help support his prognosis of the future for CIT: Railways, Electric Power and the Telegraph. He subsequently actively encourages the reader to compare and recognize the similar (monotonic, "up and to the right" shapes) of their early market growth or product adoption curves.[9] While temptingly similar to the eye, Carr proceeds to rely on their early life similarity (36 years for railways, 32 for

[7] Ibid., "Does IT Matter", Carr. p. 4.
[8] For a detailed description of Microsoft's approach and goals for the Chinese market for MIT, see CNN.com posting on February 23, 2000 at: http://archives.cnn.com/2000/TECH/computing/02/23/microsoft.china.idg/
[9] Ibid, "IT Doesn't Matter", Carr, p. 8.

Electric Power but only 12 for CIT) to pave the way for his claims of overall lifetime comparability. The problem here is a breakdown in logic: his conclusion simply does not follow from his premises. Early adoption behavior (as he shows) is simply not necessarily indicative of later life market dynamics (such as commoditization). Adoption curves are not the same as saturation "S-curves". In this way, Carr leaps to his conclusion that CIT products have already become commodities "just as his two analogous markets did" at some point. In reality, all he reliably showed was that he found three successful products (railways, electric power and IT), that, at some point after introduction, resulted in steadily growing demand for a period: no demonstrated comparability of later market penetration rates, saturation points, changes in adoption dynamics, market extensions, etc. In thrusting his conclusion on the reader, he attempts to gloss over the necessary development of the exact issues of market saturation rates which he later relies upon heavily to further his (errant) claims of the commoditized status of the CIT markets.

- Carr repeatedly documents what he sees as a failure of CIT-based projects and investments by users to deliver strategic results or demonstrable financial payback.[10] Evidence to the contrary exists in equally respected sources[11] as well as countless successful companies entering the public equity markets in the same time frame.[12] In addition, Carr implicitly dismisses (by not discussing it) a series of common answers executives give for poor ROI in a number of research work that predated his research. These rationales include:
  - overly complex MIT systems that do not have effective tools to manage that complexity;
  - the extreme difficulty in comparing alternative investment scenarios and suppliers;
  - unresolved social challenges related to MIT adoption within the organization; and,
  - undocumented and/or dysfunctional interactions between installed and upgraded systems and products that fail to achieve the performance claimed by their vendors (related to Clayton Christianson's 'overshooting' phenomenon and producers' accelerated claims often mortgaging the future for their customer's ability to successfully deploy the "not yet ready for prime time" products).

  The attributing studies and their other conclusions are discussed in detail later in the management advice section of this brief. In any case, the answers given above are hardly reasons that imply (or may be inconsistent with) Carr's conclusion that CIT has become a present-day commodity. Carr's analysis and defense seems to be incomplete at best and a predisposed approach designed to drive toward his desired conclusion at worst. Moreover, lacking the need to explore alternative possibilities, he presses on without a mention of these possible explanations.

[10] Ibid., "Does IT Matter?", Carr, chapter 6.
[11] "U.S. Productivity Growth:1995-2000", The McKinsey Global Institute, October 2001.
[12] The NASDAQ market accepted for successful listing over 300 CIT-related companies over the six years ending 2000.

- Carr makes the generalization that with identical basic CIT inputs, it is unreasonable to expect strategic opportunity or differentiated results from necessarily identically poised companies.[13] Yet he ignored a glaring counter-example that existed(s) *within* the CIT industry, even as he narrowly defined it: The $200+ billion dollar worldwide semiconductor industry. In 2003, over 12% of this worldwide chip market was proprietarily and uniquely developed and sold by 130 different "fabless" companies[14]. Significantly, these companies *all* necessarily relied on identically available software design tools, commonly available manufacturing plants and the same finite set of product assembly subcontractors to create their proprietary products.[15] Why would Carr ignore their existence in developing his thesis? Semiconductor supply is widely known to be one of the worlds largest industries and considered to be among the most strategic for modern economic development.

CIT: Available and Affordable To All?

Carr's second basic tenant is the assertion he makes that his IT (CIT) "[has] become available and affordable to all."[16] He often refers to CIT as having become a commodity. A necessary consequence of such a state in Carr's world of CIT is that users would enjoy:

- little or no costs to change CIT suppliers;
- CIT costs that are competitive and only marginally above those necessary to fully- amortize makers' costs (little or no dead weight loss in the market); and,
- ancillary operating costs that well understood, relatively free of nonlinear- or unknown scale-related issues and be largely under their own direct control.

Regarding the first of these three conditions, we *strongly disagree* with Carr's assertion that customers in the CIT market's can freely (and should) choose the most cost efficient CIT product for their application. In our view, CIT (and MIT) markets have *not* universally been a commodity for over a decade. We see much of available software as the primary counter example and believe there are considerable hardware examples as well (proprietary CPU peripherals where customers cannot freely choose alternatives as a typical one). Further, our rationale for this statement has been stated well in the context of markets enjoying network effects (as we and Carr both agree exist in CIT and MIT markets):

---

[13] Ibid., "Does IT Matter?", Carr, chapter 4.
[14] The term "fabless" has been coined in the semiconductor industry to identify a chipmaker who out sources the physical production of his internally designed products. Costing typically several billion dollars to build and facilitate, few small companies can afford the equity to make the direct investment to own their own semiconductor fabrication facility.
[15] Fabless Semiconductor Industry Association, "FSA's quarterly Global Fabless Financials Report", December 8, 2004. see: http://www.fsa.org/pubs/gffr/default.asp#files
[16] Ibid., "Does IT Matter?", Carr., p. 7.

> "…there are often large costs to users from switching technologies, which leads to lock-in. Such markets may remain very profitable, even where (incompatible) competitors are very cheap to produce. In fact, one of the main results of network economic theory is that the net present value of the customer base should equal the total costs of their switching their business to a competitor."[17]

In practice, the ability to lock the customer into to existing (or more often), future or ancillary applications products has become the standard state of affairs in the software industry. Whether accomplished through any or all of many supplier strategies including:

- product tying (potentially illegal),
- proprietary data formats,
- closed application program interfaces (APIs),
- preclusive contract terms,
- practically obligatory subscription or use fees, or
- features of a Trusted Computing (TC) architecture[18],

the effect is the same – controlling and minimizing alternatives or more often, raising the costs that users face in order to change their supplier allegiance.

As an example, a relatively basic implementation of a standard Database Management System (DBMS) product from a leading supplier (Peoplesoft) can typically $50,000-$100,000 to pay for the initial access to the appropriate software license upon which the system is based. Typically, annual licensing and renewal costs after this initial payment would be substantially less[19]. The labor and internal process costs to use such a system would be essentially the same, regardless of choice of supplier.

If however, after the initial system is completed and commissioned, the same company were to choose to make a fundamental change in its system software supplier, it would face several times its initial costs to make that change. These would include:

1. not only *writing off* all previously paid (capitalized) software and systems investments (due to the new need to be compatible with the new supplier's proprietary approaches, the old one's being rendered practically obsolete), but,
2. *restarting* the high initial licensing fee process, *and*

[17] "Intermediate Macroeconomics –A Modern Approach", Hal Varian, WW Norton & Company, New York, 1999.
[18] For a thorough discussion of the issues surrounding Trusted Computing Initiatives see "'Trusted Computing' and Competition Policy – Issues for Computing Professionals", Ross Anderson, Cambridge University.
[19] Recurring fees for maintenance, tool access and annual re-licensing of system level software typically run 20-40% of initial costs. Practically speaking, these added costs are considered to be non-discretionary to users who must be assured of continuous access to error- and security-related updates to the original products. In general, these utility updates are *not* offered separately from new product enhancements that may (often) include additional proprietary technologies – *whether the customer wants them or not.*

3. paying large incremental costs to convert the existing system's now-incompatible legacy information and internal processes so as to be compatible the new supplier's proprietary protocols – *which can amount to several times the initial cost of the system software itself.*

As if these three costs together were not enough of a *dis*incentive any change software suppliers, it is often *not* the cash outlay of the conversion process that hemorrhages most IT departments – but the *lack of availability of experienced talent and personnel to do the conversion work in the first place.* Further compounding the situation in networked CIT [MIT]: *it is clearly in the best financial interest of the CIT [as well as MIT] suppliers (and their shareholders) to do exactly this so as to make the lock-in effect with their customers as high as possible.* Carr totally overlooks this reality in dismissing the social, proprietary protocol and contracting nature of MIT in his definition of CIT.

While many cases of claimed antitrust behavior have been filed, litigated or settled against certain software and hardware companies, the reality of continued use of many of these lock-in tools by suppliers in the marketplace, implies that none of the legal motions have had their intended impact of lowering the effectiveness of customer lock-in. We revisit this topic and dynamic further in our later discussions of MIT franchises and formulating appropriate public policy toward MIT consumption.

Commodity Attribute #2: Are Software CIT Maker's Recovering their Development Costs?

One need only examine the profitability of the world's largest software maker, Microsoft Corporation, to see a contradiction to the second term of congruence to Carr's view of CIT. After tax profit has averaged 33.6% for Microsoft over the five year period between 1999 and 2003[20]. This is approximately three times the level of the Fortune 500 overall and many more times than that for a market basket of technology stocks in general (where an exact figure is incalculable as many were either not as profitable nor consistently profitable at all over the same period).[21] This has placed Microsoft in a virtual tie with General Electric for the world's most valuable corporation (as measured by overall market capitalization), while it stands over

[20] Microsoft Corporation Annual Report, 2003.
[21] http://www.fortune.com/fortune/fortune500

thirty companies behind GE in total revenues worldwide.[22] Clearly, Microsoft is able to operate in supranormal profit mode and is not generally having to price its products at amortized costs (breakeven) plus a conventional margin of markup. (Divisional profits are reported in the company's financial filings with the Securities and Exchange Commission. These numbers make it clear that the product divisions garnering essentially all of Microsoft's gross profits are its long-established Windows and office automation tool product lines[23].)

Commodity Attribute #2: Can Users Predict the Costs of Using their Product Alternatives?

Finally, the predictability of user operating costs of commonly procured software has been undergoing a virtual revolution in recent years as makers have wrestled with dramatic changes from historical fixed-price purchase policies (including preset annual maintenance fees). More recently, motivated largely by Wall Street's demand for continued revenue growth despite a slump in primary end markets unit volume purchases, many software makers have begun applying new time-based subscription pricing models to their proprietary software products. In this scheme, companies can change the terms of product support (access to updates, bug-fixes and new products as a group) and costs as a function of customer breadth of use, timeframes of commitment and other variables beyond the scope of the customer to functionally compare between suppliers or control.[24] While the flux in these new policies has yet to stabilize, the clear trend is to raise net (total enterprise) costs to users who standardize on more suppliers than theoretically necessary (buying all their software from a single or the fewest possible suppliers) and not moving toward commoditized market status. These developments are certainly not lessening the suppliers' attempts to tighten their hold on customers.

Carr's Definition of Commodity:

Carr's primary argument is that over time, information technology has moved from a source that could provide a competitive advantage for companies to the "simple cost of doing

[22] Ibid,
[23] Microsoft Annual Report, 2003, as filed in form S-1 with the Securities and Exchange Commission.
[24] Countless software companies have proposed or adopted pricing and revenue recognition models that emphasize recurring revenues (akin to annual or periodic subscription payments) over historical fixed purchase prices with annual maintenance fees (of approximately 20% of the original purchase price). Examples can be found at Microsoft: http://www.microsoft.com/licensing/default.mspx or among Wall Street Brokerage reports such as D.A. Davidson research on Design Automation Software providers such as Synopsis which can be found at: https://www.dadco.com/clientaccess2/a_research/files/reports/SNPS081804.pdf

business." [25] Carr asserts that with his definition of IT (CIT), IT is on a commoditization path, therefore it will offer little differentiation or competitive advantage to individual companies. Carr recommends that companies manage IT as a commodity input; companies should strive to achieve the necessary IT capabilities needed to compete, for lowest possible cost and risk. Carr claims that the only meaningful advantage most companies can hope to achieve from IT is a cost advantage.

We must consider scale when analyzing a company's ability to leverage potential commodities in support of their MIT needs. There may be services readily available that a small company could leverage and use as commodity IT components (in support of a small customer-base). The complexities inherent in dealing with large businesses, a customer-base that includes millions of customers, may prohibit an IT organization from being able to leverage those same services successfully.

CEO's would benefit if Carr's commoditization claims turned out to be true, that would imply IT is standardized, proven and less costly enabling companies to focus and invest in areas that can provide a distinctive advantage. It should be noted that Carr does caution companies against investing in IT innovation, fifty percent of innovation investments fail and innovation has a long lead time; but if the innovation pays off it can provide phenomenal success.

"Commodity" Defined

David Stutz defined *commodity* to represent available material used to supply a heavy demand, things that serve as basic building blocks for different purposes. More than one producer sources commodities, they conform to standards and consumers may substitute one producer's product for another's.[26] An economist could define *commodity* as 'a comparatively homogeneous product that can typically be bought in bulk'.[27] Does this imply a product whose interface conforms to standards, competition is based on price and the product is easily replicated could be considered a commodity?

The commodity business is a competitive market with low-margin and high-risk. A vender that produces commodity products needs to maintain the lowest possible cost, as a side-

[25] Ibid., "Does IT Matter?", Carr., p. 66.
[26] Stutz, David. "Some Implications of Software Commodification." Creative Commons. 2004.
[27] http://www.economist.com/research/Economics/alphabetic.cfm?LETTER=C#COMMODITY.

effect, vendors may not have their products driven by visionaries instead their product evolution may be driven by the bottom-line.

Carr defined commodity from the IT perspective, as a product that is available to all competitiors but provides distinction to none.[28] He defined commodity input for IT as a product that may not be a commodity (to the vender) but is shared by most competitors.[29] A commodity input will not provide a distinction to any individual company.

Carr states that IT is an infrastructural technology, a necessary technology shared by all competitors. Since all competitors share the same infrastructural technology, this establishes a level playing field. The industry gains the shared benefit of using the technology; no individual company gains an undue advantage.

Carr called out the characteristics that are moving IT toward commoditization:[30]

- Data Transport
- Highly replicative
- Internet, by providing a delivery channel for generic applications
- Fulfilling requirements by using third-party web services

According to Carr, IT is a necessity for a company to compete but insufficient for a competitive advantage. Whole industries benefit from IT while individual companies gain little distinctive advantage. Carr's advice to CEO's is to treat IT as a commodity input; the cost of doing business that must be paid by all but provides distinction to none. Carr claims that IT innovation is easily duplicatable, he advises companies to let competitors absorb the risk and innovation expense. Carr believes CIO's should have their IT organization follow, instead of lead, learning from the successes and mistakes of others.

To treat MIT as a commodity input; minimize IT investments. Steve Berez, vice president, IT Practice, at Bain & Co. estimates that 85% of the MIT budget goes for commodity software, hardware and LODO (Lights on Doors Open) while innovation gets 15% of the budget. Berez points that it is critical for 85% of the budget to be a successful investment (or you will fall behind the competition); the other 15% is a gamble, which may or may not succeed in

[28] Ibid., "Does IT Matter?", Carr., p. 152.
[29] Ibid., "Does IT Matter?", Carr., p. 152.
[30] Carr, Nicholas. "IT Doesn't Matter." Harvard Business Review. June 2003.

providing innovation[31]. Carr advises companies to reduce the 15% gamble, reserve investment opportunities for resources that can provide differentiation and deliver higher profits. Investment in a 'commodity input' that is shared across the industry could produce a temporary return but over time, the profits tend to end up in the hands of the consumer.

RFID is an example of innovation that will provide Wal-Mart a large competitive advantage. The competitive advantage may be short lived for Wal-Mart (shorter than most technical innovations) but the benefit they gain by knowing where its inventory is at all times is priceless. Wal-Mart is mandating their top 100 supplies implement RFID tags (at their own expense) by Jan 2005. By taking this preemptive action, Wal-Mart is proactively, assuring RFID standards will meet their *terms* and meet their *specification.* This is something of a reversal from most MIT systems where customers *adapt process to fit the software* rather than getting the product architecture supplied to them in *the way they really want it*. (From a policy makers perspective, the Wal-Mart RFID process may also signal a turn in the MIT markets from one of solely monopoly power to one also showing a degree of control by the largest consumers of MIT – a monopolistic development -- not unlike the way Sears dominated the production of white consumer goods in the 1950's and 1960's) Suppliers that have gone through the expense of deploying RFID tags (for Wal-Mart) will most likely offer RFID tags to their other customers which will lower their own operational costs and simplify their own processes by treating all customers the same, with an added perceived benefit of providing their customers a bonus feature the customer could elect to take advantage of. This Wal-Mart mandate guarantees the new technology is here to stay and the innovation will benefit the industry as a whole. Wal-Mart competitors will have the opportunity to leverage the benefit of RFID faster, easier, and cheaper with less risk.

Should a commodity be easily replicatable? Carr sends out mixed signals when he talks about IT being replicative. In "IT Doesn't Matter," Carr stated, "IT is also highly replicatable. Indeed, it is hard to imagine a more perfect commodity than a byte of data – endlessly and perfectly reproducible at virtually no cost."[32] While in the same document, Carr acknowledges that there are many facets of IT and the commoditization of technologies does not necessarily translate to making the best IT implementations easily replicable. These replication references

[31] Wharton School of Business, Information Technology: Value creator or commodity? CIO Magazine. March 2004.
[32] Carr, Nicholas. "IT Doesn't Matter." Harvard Business Review. June 2003.

demonstrate that while Carr's IT definition is restricted to hardware and software, Carr does recognize that IT system implementations are more complicated and do not necessarily fall into his 'highly replicatable' classification.

Commoditization is the process of turning a product into a commodity. The product typically starts out as a specialized technical innovation at high cost. High demand earns the vendor a large profit. As competition rises, demand for the product leads to standardization. The product evolves to be largely homogeneous, competition becomes fierce, profit margins become low and venders lose branding value. Commoditization shifts the market power from vendors to buyers.

Standardization and commoditization do not imply innovation will slow down. Once products have become commodities, their increased availability and decreased cost make the commodity readily available for use by innovators in experiments encouraging further innovation. Today's commodities can be the building blocks of tomorrow's innovations. Some economists believe that the faster the pace of innovation, the more common commoditization will be.[33]

Carr wrote that IT is prone to commoditization; causing discussion as to whether IT has already become commoditized. The discussions included the amount of in-house IT development companies are doing and the percentage of customization typically performed on off-the-shelf software. If we assume that IT could turn into a commodity, is IT at that point now? Is IT on the commoditization path?

To determine if IT components can be classified as a commodity, we will build a set of assertions that will allow us to test various facets of IT. As previously stated, a commodity starts out as a product that could have been derived from a specialized technical innovation; through demand, competition and standardization it evolves into a component that has the potential to serve multiple useful purposes. Demand and potential profit create strong competition through multiple venders.

Standardization follows demand resulting in a largely homogeneous product. Products will compete on price rather than features. This does not imply that the market will not have a variety of products with unique features; rather the varieties of products compete on price.

[33] http://www.economist.com/research/Economics/alphabetic.cfm?TERM=CONTESTABLE%20MARKET#COMMODITISATION

Vendors of commodity products must adhere to standards, creating commodities with a common feature set and interface. Consumers should be able to swap one vendor's product for another's with minimal switching costs.

For the purposes of this policy, the commodity assertions will be:

- Product is a useful component with multiple purposes
- Competitively supplied
- Venders' products compete on price not features
- Consumers may swap one vendor's product for another's with minimal switching costs

Dan Farber stated, "IT is a commodity if the technology itself is built out of fairly standard components that don't vary greatly among vendors or provide truly unique advantages".[34]

Carr's Dimensions of IT

Carr's definition of IT is restricted to hardware, software and the network; Carr stated that any assessment of whether IT is becoming a commodity input must examine IT as both hardware and software.[35] When looking at hardware with respect to IT we will look at desktop PCs and enterprise servers. Desktop PCs were the first hardware to be commoditized. Obviously, Dell, Compaq, HP, Gateway, and Sony competitively supply PCs. With the onset of web applications the vast majority of personal computers, required for an organization, can be entry-level computers; as long as it runs Windows (or the company's preferred OS), reads e-mail, has a browser that can provide efficient access to web-service based applications and supports basic office applications - price is the determining factor in the purchase. Along the same line, as long as a PC runs the company's preferred OS, reads e-mail … a company can easily swap one vendor's PC for another's. Therefore, desktop PCs pass the Commodity test.

Until recently, IT organizations typically have had mission critical enterprise applications run on enterprise-class hardware. Enterprise-class hardware competes on features not price where the price could range from $50K to $1 million plus. Google is an example of a company that uses generic hardware for enterprise servers and has set an example for other companies. Google requires extensive processing power yet it was able to build its systems out of

[34] Farber, Dan. "The end of IT as we know it?" Tech Update, May 2003.
[35] Ibid., "Does IT Matter?", Carr., p. 32.

inexpensive, basic servers. The trend to replace enterprise-class hardware with generic commodity servers is taking hold in the industry. Grid computing offers a lot of promise in this area. While there are low-end (commodity) servers that companies can use for mission critical enterprise applications this trend is still in its infancy and there are limitations to its widespread implementation; those limitations include supporting the minimum hardware requirements for the existing enterprise applications. IT's enterprise servers are not a commodity but they may be moving down the commoditization path.

Carr recognized two of the most significant criticisms to his commoditization claim were the (almost) unlimited potential for software innovation and the continuous evolution of IT architecture. Carr concedes that software is malleable and adaptable, traits that make software less susceptible to commoditization but he believes software has other qualities that make the commoditization argument. He also noted that in the future, most corporate software would be a commodity good churned out by cubical workers spread across the globe.[36]

Carr believes the trends in IT architecture over the years, from mainframe systems to client server systems to web-service-based systems, have led to more standardized, more homogenized architectures as companies have connected to broader networks. IT architecture is still evolving, but Carr argues that the current trends have moved away from proprietary architecture toward open, shared, standardized architecture that focuses on performance and reliability.

It is difficult to analyze software with respect to IT; 'software' is too broad with unlimited possibilities. To differentiate between various types of software, I will refer to 'packaged' software as shrink-wrapped software, such as MS Word, that can be easily installed; this type of software does not allow for extensive customization. 'Off-the-shelf' software will refer to vendor supplied software that allows (or requires) a high degree of configuration and customization such as a CRM solution. 'Proprietary' will refer to custom software that is build for a specific organization whether the code was written by contractors, the in-house IT developers, out-sourced or off-shored development. 'Hosted' solutions will refer to systems that are built, maintained and hosted by an external vendor.

[36] Carr, Nicholas. "Does Software Matter?" May 2004. http://www.darwinmag.com/read/050104/carr.html

Instead of looking at CIT (simply hardware and software) we will explore the potential of software commoditization from the system perspective; we will look at MIT core systems to see if they can pass the commontization test. MIT is more than hardware, software and networks; it is how the technology is applied, it is the supported end-to-end processes, it is the talent-pool of the IT department. IT reduces business costs thru productivity, improves customer satisfaction, streamlines the supply chain, enables global operations, enables regulatory compliance and enables business innovation.

The core IT systems within a company include billing and collections systems, care systems, finance systems, internal IT share services, marketing systems, people systems, sales systems and systems integration. Due to the unlimitied possibile structures of various IT organizations, this policy will sample some core systems that a company may depend on suchas ERP, shared services, and desktop applications.

While analyzing various IT systems, to see if they could be classified as a commodity, we must consider scale. It may be possible to identify examples of systems, for small-business use, that would pass the commodity test while the very same system would not meet a medium to large company's needs. For the purpose of this commodity analysis, we will restrict discussion to systems that have the potential to meet the IT needs of medium to large sized companies. Analyzing software available strictly for small business could be considered a trivial task. Small businesses have the ability to use a collection of packaged commodity software that can meet their individual IT needs.

Enterprise Resource Planning (ERP)

Enterprise Resource Planning (ERP) is a multi-module back office system consisting of off-the-self software designed to support multiple enterprise functions. ERP modules can include software for manufacturing, order entry, accounts payable, accounts receivable, finance, purchasing, CRM and HR. The term ERP implies the use of 'off-the-shelf' software rather than proprietary software written for a single organization.

ERP software is typically written for specific vertical markets; within that market, the software is vendor-customized to support the market's specific needs. When implementing an off-the-shelf system, such as ERP, best practice is to minimize the amount of customization a company performs (to ease vendor technical support issues and upgrade difficulties). While

keeping customization minimized is an ideal goal, it's difficult to achieve since business processes are so deeply engrained in the ERP solution; it's difficult for a business to resign how they conduct their business to an off-the-self product that may not mesh (in some areas) with how the company wants to conduct business. In an effort for IT to be responsive to business innovation, to be flexible to company requirements, customization tends to creep beyond the recommended percentage. IT typically strives to drive the top 10 percent of its customizations back into the off-the-self-software product as requests for enhancements (RFE), minimizing the amount of customization required for future upgrades.

Applying the commodity test, we see that ERP is a multi-module application designed to support various enterprise functions; implying it is a useful product with multiple purposes. ERP is competitively supplied by various ERP vendors such as Oracle, SAP and PeopleSoft to name a few. While some ERP vendors started out with different focuses, (PeopleSoft had an HR focus while SAP started in manufacturing); the functionality that evolved out of the vertical tracks remains fairly consistent. Is ERP product selection based on price? For one of the big vendors (Oracle, SAP, and PeopleSoft) we are talking a multi-million dollar investment. Small and medium sized companies can get away with a smaller investment using one of the smaller niche ERP vendors (QAD, i2, Ross).

Can one vendor's ERP solution be swapped with another for minimal switching costs? Absolutely Not! A big ERP solution cannot even be upgraded with minimal costs. ERP is expensive and difficult to install (average installation 1-3 years). It is not easily replicated. Once established in a company the work never ends - there will always be more modules to install, version upgrades to support and Enterprise Application Integration (EAI) opportunities. Carr implies that companies should avoid upgrading; do not think version upgrades can be ignored or avoided indefinitely … as venders roll out newer versions of their software, they set an end of life (EOL) date on older versions. Companies need to maintain technical support on mission critical applications, forcing them to upgrade. It is possible (and common) for large companies to apply pressure postponing the end-of-life for up to a year but eventually, the organization will need to upgrade. The level of integration and customization increases the complexity, cost and time required for the upgrade. It is possible for the ERP upgrade to have a snowball effect, ERP versions have certified on certain platforms, certified with certain products. An organization may find that they have to upgrade their OS version, directory version, database version, integrated

application versions (such as external authentication or billing systems) in order to maintain technical support. A company that relinquishes their business processes to an ERP solution cannot afford to go without tech support on their implementation. Unless the ERP vendor is hungry; it is unrealistic to expect a vendor to support every possible platform version and every possible integrated product version for every customer.

Once a company buys into an ERP solution, ERP turns into a project that never ends. The benefit of ERP is a single platform that provides a global view. The curse of ERP is a single platform that impacts IT's ability to be responsive to business needs; there is a strain on the development factory attempting to organize simultaneous projects on the same code base. The quality risk sometimes outweighs the responsive need; many large organizations are currently discovering and attempting to deal with this issue. In theory, ERP solutions may seem like a commodity but it does not pass the commodity test due to installation/integration complexity, companies may find themselves locked into an ERP solution due to the cost impact of having to change business processes (again). ERP is not on the commoditization path. ERP is not a commodity input due to the multi-million dollar investment that does not have a solid success record for companies rolling out ERP solutions. History may show that competitors who elect to invest wisely in their proprietary solutions may end up with a more cost effective solution that is flexible to the company's needs.

Shared Services

Shared services were also known as, 'application services' but now are more appropriately called web-based application services. Web-based services exploit the existence of the Internet and the end user's browser for a ready-made user interface.

Web-based services refer to a standardized production framework for the development of loosely coupled business applications. Web-based standards and tools enable easily deployable services; providing a framework to expose functionality and data. The framework allows the encapsulation of re-usable functionality enabling functionality to be independent of hardware and software. These services support B2B, B2C and internal use. Security and access control play a large role here (or need to be playing a large role). Organizations need to protect the confidentiality and integrity of the potentially sensitive business information for their partners and customers.

There are may different types of web-based services; proprietary services, hosted or fee-based services, and out-sourced services. Proprietary services are developed for/by the organization and have the potential to provide value-added functionality in key competency areas, they can be used to replace fee-based services when there is a cost advantage. By definition these proprietary web-based service do not qualify as a commodity since they are built for an individual company filling a specific need. It should be noted that the proprietary web-based services are built out of commody components such as operating systems, web servers, databases, network…

Fee-based web services, or third-party web-based services offer a complete solution for a narrow problem, a credit check service is a good example. The type of shared services a company needs is industry dependent; possible services could include a calculate tax service, credit card service… Many companies use these services to out-souce an application's care and feeding, if the TCO for a third-party vendor is lower than the TCO for a similar proprietary service, companies should switch to/use the fee-based services. Fee-based web services are too broad a subject to apply to the commodity test. It can be assumed these services will support the standard frame work, but we cann't generalize the services and deduce if each type of service competes on price instead of features or if it's competitvly supplied therefore we will not be able to catorgize them as a commodity, but they should be catorgized as a commodity input. Looking at Carr's definition of commodity input, these services will be available to all competitors but provide distinction to none.

The Desktop

Every desktop in an organization is most likely imaged with an basic set of software; such as the OS, office-productivity tools, e-mail, browser, virus scanner, potentially a personal firewall and other control utilities. Let us look at Microsoft Office as the provider of the office-productivity tools. MS Office does not pass the commodity test because it does not compete on price. Carr classified Microsoft Office as a commodity input that is shared by most competitors giving a distinctive advantage to none.[37]

[37] Ibid., "Does IT Matter?", Carr., p. 152.

As open-source software continues to improve, there will be fewer reasons for a company to purchase software licenses for software packages like MS Office. Some companies, on a tight budget, may consider Openoffice an adequate replacement for Word. Does this imply that software like MS Office is on the commodization path? Open source has been called the 'commoditization of software', Robin Good stated that MS is 'getting rightly concerned' as is evident by the pdf Microsoft published showcasing the differences between Microsoft office and Openoffice.[38] While Openoffice may start MS Office down the commoditization path, we are not there today.

Regulatory Compliance:

Sarbanes Oxley (SOX) is a recent example where regulatory compliance had costly IT implacations for many publicly traided companies. Sarbaines 404 requires publicly traded companies to make assertions regarding the internal controls over the company's financialy significant reporting. The focus is security; access control and segregation of duties. To meet year one compliance, many companies put manaual procedures in place, but these procedures will need to be automated for long term effectivienss. If an IT organization needs to support regulatory complieance, all their competitors should be in the same boat. A temporary cost advantage will go to the company who can be responsive in a fast and cost effective manner. IT organizations need to invest in security and be in a position to response to regulatory compliance.

Is IT a Commodity? *No*

IT is more than simply hardware, software and networks, it's the information, the mission-critical systems, the end-to-end processes. Whether mission critical applications use off-the-self , propretary or packaged software the analysis shows that currently IT can not be considered a commodity. Systems that support mission critical processes may not compete on price; while price is a consideration, the available features may take top billing. Systems with the potential to scale enough to support a large organization may not support minial switching costs. Monopolies may prevent packaged software (such as MircroSoft Office) from being competitievly supplied. If IT was indeed a comodity, that would be good for individual

[38] Good, Robin. "Open Office vs Microsoft Office (according to MS)." March 2004. http://www.masternewmedia.org/news/2004/03/26/open_office_vs_microsoft_office.htm

companies and the industry as a whole. Companies would save money, be vendor indepent and allow the business to focus resources on enhancing their core compentencies.

Is IT on the Path Toward Becoming a Commodity? No

You will find components of IT may be considered on the commoditization path. IT as a whole, with a purpose to automate business processes and provide a global view of data, has been shown not to be commoditized. We do not anticipate the future commoditization of all aspects of IT, but there is commoditization of industry standard components that serve as the foundation for the bulk of MIT's system solutions (as seen with web-based services). Commoditization results in little differentiation between similar classes of products, commoditization erases the technical advantage and eliminates vendor dependency. The market advantage will continue to reside with the organization that knows how best to use IT.

Should IT be treated as a commodity input? Yes

Carr makes a solid recommendation that makes good business sense; CEO's should treat IT as a commodity input, run IT like a business. Structure internal IT departments to cost less, provide higher quality, be flexible and make the business efficient. This policy supports Carr's recommends, if companies treat IT as a commodity input organizations will get the IT they need for less money and with less risk. Carr recommends companies move away from cutting-edge technology and not invest in innovation unless it is necessary. If common sense prevails, this recommendation would serve as a general guideline (or policy if you will) but the final decision needs to take the enterprise road map into account as well as the project's business requirements. A small portion of the IT budget typically goes toward innovation; the risk of these projects failing is high, the reward even higher. A company should resist innovation unless it strategically aligns with the organization's enterprise roadmap (similar to Wal-Mart's RFID project).

The Standardization of IT in Business Use

The fourth premise that we investigate is Carr's argument that an infrastructure technology becomes a commodity when, "Both the technology and modes of use become

standardized."[39] Certainly many IT implementations in businesses are standard, with little variation between companies. However, most obvious examples of these standard uses can be found in small businesses with uncomplicated needs, leaving the question if more complex MIT solutions really are becoming commoditized by Carr's definition.

If MIT use has indeed become standardized, there should be evidence of this in both high rates of IT project success and broad adoption of standard training for IT professionals. Following Carr's analogy to older technological revolutions, we note that even in large building complexes the electric system usually works once installed and training for electricians is governed by strict guidelines. Carr notes the shortcomings of CIT projects and the related high cost overruns, but dismisses problems in CIT project implementation as growing pains as best practices are established.[40]

To explore Carr's claim of the standardization of IT use, we look at three recent surveys of IT project success, investigate factors that these surveys claim lead to project success (i.e. best practices) and then compare those factors with a study of IT project success rates from 1984. Our comparisons show that many best practices in IT project implementation were understood at least twenty years ago, indicating that Carr's optimism regarding the outcomes of businesses rapidly employing well-understood best practices may be misplaced. We also discuss other factors in the ongoing struggle for companies to implement complex IT projects.

To further understand the degree to which IT use is becoming standard within companies, we examine the training and job requirements for IT professionals. We explore job advertisements for three groups of professionals, systems administrators, web masters and project managers. The advertisements are analyzed both for required certifications or training and also for uniformity in desired skills and abilities. We find certification is not yet widespread among these classes of professionals, and also find significant diversity between skill sets required for the same position by different companies, with interesting significance to Carr's suppositions.

---

[39] "Does IT Matter", Nicholas G. Carr. Harvard Business School Press, Boston, Massachusetts, 2004. pp. ??.

[40] In this section of the book Carr strays from his straight and narrow definition that we have been referring to as CIT, and seems to converge on a more broadly understood definition. Therefore we return to a more neutral IT syntax.

Examples of Commodity IT Use by Business

There is evidence to support Carr's supposition that in many industries, IT adoption confers no strategic advantage, is easily deployed and enforces business best practices. An obvious example is accounting software available for small businesses from Intuit, providing bookkeeping, payroll services and retail support.[41] With a starting price of $199, these packages satisfy basic financial management needs and include customization by industry as well as tutorials on finance and business. The large market share commanded by Intuit results in interfaces to most banks, allowing users to automatically update account information. Given the low price, ease of use and widespread adoption, the use of accounting packages for small businesses seem fairly standard.

A similar product with broad adoption is Macromedia's suite of web development applications.[42] These packages provide tools that enable businesses to rapidly develop and deploy websites, using best practices such as test deployment and content control at relatively low cost ($399 for DreamWeaver). There is a subtle difference between web development tools and accounting tools, however. Accounting and payroll practices are well documented and understood, while web development remains more of an art than science. As recently as 2002, 61% of survey respondents found the Internet difficult to navigate and reported frustration trying to make purchases online.[43] Amazon invests considerably in research to determine the most effective page layouts, and keeps the results of this research private.[44] So while the technology to produce a website is certainly a commodity, we do not yet have an understanding of how to optimize a web site for customer interactions. We will return later to the lag between available technology and usable technology, for now noting that developing web sites remains only a partially standardized practice. Carefully seeking out and implementing best practices is probably enough for most businesses, but innovate use can still yield a competitive advantage.

---

[41] http://quickbooks.intuit.com
[42] http://www.macromedia.com/resources/business/
[43] Rich Internet Applications (http://download.macromedia.com/pub/solutions/downloads/business/idc_impact_of_rias.pdf accessed November 30, 2004)
[44] CSE Colloquia Oct 28, 2004

Success Rates of IT Projects

Once you move beyond shrink-wrapped software packages such as those discussed above, implementing new IT projects within a large company is a known source of nightmares. The widely cited Standish Group study in 1994 brought attention to the extremely high cost of IT projects and their relatively low success rate.[45] The topic remains an active one in IT publications and government projects are not immune; the IRS currently has IT project overruns of over $500 million.[46] Carr notes that continued loss of capital due to failed IT projects is a significant problem for businesses[47], but does not explore the reasons for those failures.

Follow-up Standish Group surveys in 1998 and 2003 still found high rates of failure and significant cost overruns, with some signs of improvement. From 1994 to 2003, the survey reported an increase in the percentage of successful projects from 16% to 34%.[48] Other studies find similar results. A large survey of project managers in the UK by Computer Weekly found lower project success rate than the Standish Group, with only 16% of projects being successfully completed, using the same criteria as the Standish Group but different project selection criteria.[49] Finally, a 2003 study by the Hackett group also found a 30% project failure rate in large projects.[50] In all these studies, attempts were made to understand the factors leading to project successes and therefore quantifying best practices for IT project implementation.

The following table from the Computer Weekly study quantifies changes in IT project management success from 1995 to 2003 by comparing the results of the Standish Group with their own. While there are distinct differences between the surveys, likely due to differences in underlying populations and methodologies, they all show that IT projects rarely fully succeed. However, the most recent Standish Group results illustrate that failure rates have dropped significantly over the last decade, from 31% to 15%.

---

[45] Standish Group, "The Chaos Report (1994)," Report of the Standish Group, 1994.
[46] For the IRS There's No EZ Fix (http://www.cio.com/archive/040104/irs.html , accessed November 29, 2004)
[47] Ibid, "Does IT Matter", Carr. p. 111.
[48] Latest Standish Group CHAOS Report Shows Project Success Rates Have Improved by 50% (http://www.standishgroup.com/press/article.php?id=2 accessed November 28, 2004)
[49] State of IT Project Management in the UK 2002-2003 (download from http://www.computerweeklyms.com/pmsurveyresults/surveyresults.pdf November 10, 2004)
[50] Three Out of Ten Major IT Projects Still Fail According to Latest Research from The Hackett Group (http://www.thehackettgroup.com/ab_pr.asp?id=289 accessed November 10, 2004)

	Abandoned	Challenged	Successfully completed
Standish Group 1995	31%	53%	16%
Standish Group 1999	28%	46%	26%
Standish Group 2003	15%	51%	34%
Computer Weekly 2003	9%	75%	16%

While it is tempting to attribute the increase in success to standardization of IT use, this reduced failure rate alone does not indicate that adoption of more uniform business practices. The Computer Weekly survey found that project size directly correlated with chance of project failure. Less expensive, shorter-term projects ran failure rates of 6% or less, while more expensive, longer term projects had failure rates close to 15%. In the 2003 report, the Standish group mentions that average project budget was only half that in 1995, but does not break down project failure by size. Therefore, a significant reduction in the overall project failure rate could be ascribed to undertaking smaller, more manageable projects rather than quantifiable improvements in process.

Analysis of project success rates over time does not conclusively support or dispute Carr's argument that IT usage is becoming standardized. Carr does not comment extensively on the continuing struggle of IT managers to implement new IT projects but calls these failures, "…a natural consequence of the process of trail and error that goes on as any new technology is adopted by business."[51] Since the three reports cited above intend to analyze this trial and error, we continue our analysis by investigating the factors that these groups feel lead to project success.

Factors leading to successful implementation of IT within a business can be broken down into two broad categories, organizational factors and those pertaining to individual projects. All three surveys look at organizational factors and, as shown in the chart below, highlight similar areas that have an impact on efficiency of IT use within a business.

	Standish	Computer Weekly	Hackett
Centralized project management office or methodology	**X**	**X**	**X**
IT support at high levels or organization	**X**	**X**	**X**
Standardization of IT products	**X**		**X**
Career track and training for IT project managers		**X**	

[51] Ibid, "Does IT Matter", Carr. p. 112.

All three studies show a high degree of consensus in organizational factors that improve the efficiency of IT project implementation within a company. They agree on the importance of a firm commitment to IT project success from the top levels of the company, and support at least adoption of a centralized project management methodology, preferably an actual project management office. All of the factors speak to IT being recognized as an important part of the organization.

The Hackett study does not discuss individual project issues in as much detail as the other two surveys, both of which focus on this dimension of project success. Again the degree of consensus between the studies is high. Here we include a fourth, earlier study for comparison.

	Standish	Computer Weekly	Hackett	ITT Study
User involvement	X	X		X
Experience project manager/personnel	X	X		X
Alignment with business objectives	X	X		
Minimized scope	X	X		X
Unchanging requirements	X	X		X
Ability to measure success	X		X	

Clearly there is a significant degree of overlap in these success factors, indicating at least an awareness of what could be considered best practices. Both the Computer Weekly survey and Standish Group note that projects will be more successful if users are involved, requirements don't change, personnel are experienced and project scope is small. How unique are these findings? In 1980 a group at the ITT Advanced Technology Center applied a statistical analysis to 44 programming projects within the company, looking for factors that predicted project productivity (In 1980 most IT projects were programming projects).[52] They found that smaller projects were more productive, as were projects with stable requirements, those with client participation and projects employing experienced personnel.[53] These findings have been widely sited and are generally well known within both the software development and IT project management communities. However both the Standish and Computer Weekly surveys indicate that a significant number of companies do not follow these practices. This failure could be understood in a number of ways. First, that companies are extremely slow at adopting practices

[52] Vosburgh, J, B Curtis, R Wolverton, B Albert, H Malec, S Hoben, Y Liu. "Productivity Factors and Programming Environments," *Proceedings of the 7th International Conference on Software Engineering*, IEEE Computer Society, 1984

[53] Other issues, such as speed of available CPUs and developing new hardware platforms, have ceased to impede project success

that could save them significant amounts of money, second, that impediments to their adoption exist, or finally, that best practices are not enough to insure IT project success. We will return to these questions after exploring standardization of IT professionals.

Standardization of professional practices

Workers in the fields that Carr cites as mature commodity markets are often certified and generally receive standardized training before entering their field. For instance, in the state of Washington, electricians must be certified by the state to work in their trade.[54] Certification generally requires an exam and pursuing continued approved education. For electricians, certification was instituted as early as 1899 in some states, less than 20 years after the construction of the first electric power station.[55] Over 30 states require certification of electricians.[56] Similarly, railway engineers must complete training and pass a qualifying test.[57]

Fifty years since the business use of the first mainframe computer, the training and certification of IT professionals remains uneven. While licensing is not yet required for any IT professions, training for some positions is more standard than others. We will not try to measure a change in certifications over the past decades since such an analysis would require deconvolution of many interdependent factors such as the size of the labor pool and demand for a given position. Instead, this section will look at several IT positions and examine the training and job skills required for these positions.

To provide some coverage of the range of IT professionals, three common positions will be discussed: systems administrator, Webmaster and IT project manager. For each of these positions the range of relevant professional certifications will be presented. Furthermore, by analyzing two recent weeks of postings on Monster we will explore the importance of those certifications to employers. In addition, the split between quantifiable and abstract job requirements will be approximated to provide some indication of requirements that could be

[54] Facts About State Certification for Electricians (http://www.lni.wa.gov/IPUB/500-004-000.pdf , accessed November 21, 2004).
[55] History of the Board of Electricity (http://www.electricity.state.mn.us/Stat_rul/Brd_hist/ accessed on November 21, 2004)
[56] The Committee for Electrician's Certification – FAQ's (http://www.ibewnecaforcertification.org/faqs.htm#f accessed November 21, 2004)
[57] Career Browser: Rail Transportation Workers (http://www.collegeboard.com/apps/careers/0,3477,32-244,00.html accessed on November 21, 2004)

standardized. While a required proficiency in Perl can be measured and could be included in a certification exam, business analyst skills are less concrete. The following analysis is neither scientific nor exhaustive, but does provide some insight into the current degree of standardization in IT professionals.

Some certifications are available for systems administrators, although they are not a requirement for many jobs. Only 12% required a specific certification (Microsoft, Sun and HP); certification was preferred in an additional 12% of positions. Systems administration positions are very technologically specific, with all postings including detailed lists of known operating systems and many listing required scripting languages and enterprise applications. The range of required technical skills listed within the 25 postings examined was considerable including 5 different operating systems, 10 different scripting languages and numerous different network, backup and firewall applications. In addition to technical skills, 68% of the Monster postings mention strong communication skills as a requirement for a systems administration position. Other non-technical skills listed include providing customer service, writing proposals and project management. Most job postings require a bachelor’s degree or higher, preferably in computer science.

The range of technical skills required across all 25 postings is likely a symptom of the degree to which IT products haven’t settled down to a significant degree of standardization. As noted earlier, network effects require most companies to maintain multiple operating systems with their associated software. While additional certification and standardization of system administrators is possible, lack of standardization and continued innovation in software and hardware systems reduces incentives to either seek or require certification. Also, non-technical capabilities remain important, since systems administrators often fill roles of customer support and project management within smaller businesses and groups.

Webmaster positions do not have any associated certifications. Even the education requirements for these positions are vague, with fewer than 30% requiring a bachelor’s degree or higher. The range of technical skills was not surprisingly narrower than for systems administrators. Web sites are built in HTML, a highly standardized format. Other required skills included proficiency in several common software packages (such as Photoshop, DreamWeaver and ColdFusion), knowledge of SQL and strong communication skills. Despite documented issues with web navigation, only 2 (8%) positions required knowledge of usability as a job

prerequisite, while 6 (24%) positions required graphic design. Some larger companies will often employ usability engineers to develop the structure of a website, but that remains a very small field.

Given the relatively narrow range of technical skills required for webmasters, knowledge of skills could potentially lead to national certification, and supporting organizations currently exist. Graphic design and usability can also lead to certifications, such as those available at the University of Washington Extension program.[58] In general, this position continues to change as tools decrease the difficulty to create basic websites, but user demands on functionality increase.

There is no unique certification for IT project managers, but the PMP certification is broadly recognized in the IT community. Almost 30% of job postings for Senior IT Project Managers mentioned certification, most of them specifying the PMP by name. None, however, made this certification a requirement for the position. In general the job requirements for project managers focused less on technical skills and more on experience and organizational skills. Not surprisingly, interpersonal skills show up in all postings as an important skill.

A 2003 Standish and Group survey discusses project management attributes that lead to project success and find business, technical and project management skills are the three most important traits. The job postings generally reflect desire for project management expertise (all mention experience in the field) and most (52%) mention specific technical understanding. References of experience or training in a specific field appear in 40% of the postings. While expertise within a field can come from experience and is difficult to certify, the broad range of technical skills mentioned within the 25 postings exhibit similar impediments to standardizing technical training of systems administrators.

This survey of 25 positions in 3 different areas also reveals a surprising degree of overlap within job responsibilities. Some of the project management positions required extensive knowledge and support of computer systems, similar to those found in the system administration position, while others required web development experience. Likewise the systems administration positions occasionally required project management skills or ability to deploy and maintain a web site. Finally, the Webmaster position remains poorly defined in many organizations with a significant number of job titles mixing other positions, such as

[58] http://extension.washington.edu/ext/certificates/bytopic.asp

Webmaster/Network Administrator. Project management skills are also found extensively within the stated requirements for various Webmaster positions. The overlap between these seemingly distinct positions, and for Webmasters and Project Managers, overlap into non-IT areas, indicates how important IT skills have become in the workplace. While it is tempting to follow Carr's argument that this is in the process of changing, each new technological innovation seems to increase the degree of technical knowledge required by the general workforce, not decrease it.

Unlike positions in older commodity industries, certification of IT professionals remains rare. Indeed the range of skills required by many IT professionals is sufficiently diverse that only multiple, specialized certifications make sense. Both network effects and rapid evolution of the field are important factors to this lack of standardization. Two additional factors may play a role. One is that unlike electricity and the railroad, IT use has only recently been seen as an issue needing to be controlled for the public good. The introduction of HIPPA and the Sarbanes Oxley Act may be the first steps in altering this balance. Another distinction between IT professionals and the commoditized workers that Carr cites is the presence of unions. In the US, telegraph operators and railway workers both unionized in 1863[59] [60] and electricians in 1891[61], all within 30 years of initial adoption of the technology. Unionization of IT workers remains unusual, for complicated political and social reasons.

Why does use of large IT systems remain difficult?

Despite Carr's assertions that IT use is standardized in businesses, the above findings suggest that, outside simple applications, many businesses continue to find fully implementing MIT solutions to be costly and difficult operations. Advice on how best to pursue these projects has not changed significantly over the last 20 years, indicating a) that businesses are very slow learners, b) these practices are difficult to adopt or c) that we don't yet have a full understanding of how to use available MIT capabilities. While there is certainly a grain of truth to the first supposition, most companies will seek to adopt practices that whose payoffs outweigh their implementation costs.

---

[59] Franklin Pope (http://www.telegraph-history.org/pope/ accessed on November 28, 2004)
[60] About the BLE (Brotherhood of Locomotive Engineers) (http://www.ble.org/about.asp, accessed on November 28, 2004)
[61] INTERNATIONAL BROTHERHOOD OF ELECTRICAL WORKERS (http://www.ibew.org/History/brotherhood.htm accessed on November 28, 2004)

The second supposition is worth some scrutiny. The four factors leading to project success found common over two decades are smaller projects, stable requirements, client participation and employing experienced personnel. Making projects smaller is not always an option. Employing experienced personnel is common, although it does require a company invest in hiring and retaining good personnel. Keeping requirements stable presents a greater, potentially impossible challenge, especially when including client participation. In his seminal essay on why software construction remains so difficult, Fred Brooks discusses the inevitability of demands for change on any software solution.[62] He traces this demand on the fact that a software system embodies a function and exists within a malleable cultural matrix. Even on a short time scale, understanding the needs of the system often changes through construction, leading either to change of requirements or delivering a system that does not meet the needs of the user.

Carr drastically underestimates the difficulty of change in his discussion of software, and by extension, software systems. Referring to object-oriented programming languages Carr optimistically states, "Modularization makes it possible for programmers to quickly replicate- or surpass- the functionality of existing programs."[63] Unfortunately facts do not bear this out. Despite widespread adoption, object-oriented development hasn't resulted in order of magnitude increases in programmer productivity.[64] Meanwhile, software development practices are evolving to allow for the inevitability of change, with the increasing adoption of so-called agile methodologies. IT project managers are also exploring some of these methodologies.

A final explanation for the lack of adoption of best practices in MIT use is that those best practices are not yet fully understood. This is especially obvious in the realm of user interfaces, where one study estimated that poor user design cost thousands of dollars per year in lost productivity on common desktop machines.[65] MIT handles human interaction with information instead of the physical world. While railroads and electric lights all had earlier analogies, prior to the advent of computers the primary means of storing and relaying information was on paper.

---

[62] Brooks, FP The Mythical Man-Month Anniversary edition. Pp 184-5.
[63] Carr pg. 52
[64] Potok, TE, M Vouk and A Rindos. Productivity Analysis of Object-Oriented Software Developed in a Commercial Environment. Software – Practice and Experience, Vol. 29, No. 10, pp 833-847, 1999
[65] Constantine, LL and LAD Lockwood. 1999. *Software for Use*. ACM Press, New York. P. 10.

The past fifty year represent only the very beginning of humans developing new ways of thinking about information, so the current state of confusion is perhaps not surprising.

Some Thoughts on Forming Effective Public Policy Within the MIT Markets

As we interpret his case, Carr would have his readers believe that the market's "invisible hand" will do the job of the policy maker (as a commoditized space, has already done it!) and deliver products and markets that properly balance the interests of all for optimal social good. He might say,

> "CIT users are the defects! They simply fail to adopt best practice and the lowest cost alternative "commodity" choice!"?

Would Carr's implicit "policy" simply have MIT customers eliminated by the competitive dynamics of the markets as the suppliers continued to thrive on sales of their "commodities" and lay waste to better uses of the industry-wide IT development potential in the process? We see this as a problematic rationale leading to the wrong conclusion. Even if such a *laissez faire* interpretation of Carr were inappropriate, because of the flaws in his foundational arguments, his legitimacy to speak to policy issues in the real MIT world (versus his imaginary CIT realm) has to be seriously questioned, even as he truly provides little advice in his text. So, the reader of Carr is largely left to develop his or her own view.

Commodity vs. Most Popular Alternative: The Double Fulcrum

Determining the best course for public policy to take seems to be at least as difficult as analyzing and describing the current state of MIT markets (and therefore the need for proactive policy in the first place). Our research leads us to believe that perhaps the best place to start is acknowledging the difference between:

a commodity MIT product
(as Carr advocates all are within CIT),

versus

the most popular alternative
(that most have elected to use,
i.e. the winner of MIT's natural
network effects).

In a commoditized world, the "invisible hand" rewards innovation appropriately for the right period of time and resources are redirected to perfect the offering of the next innovation to the market. This dynamic is implicit in the definition of a commodity and we have described this in detail in the previous pages.

But in the real MIT realm, network effects *in combination* with naturally occurring purchaser "lock-in", act to "gum up" the frictionless movement toward economic rebalancing as in a free market. This stalling out of market evolution goes far beyond the classically described economic "stickiness" that many markets exhibit due to human psychological tendency to resist change. The user of MIT has little or no practical choice and from there is gets worse – not only have the macro economics of the market tipped in the favor of his choosing the monopolist supplier, but once making any decision, his own micro economics tip in favor of his own status quo. He pays dearly to change his mind, regardless choice. This is important as the latter, microeconomic dynamic applies even if the user actually made a sub-optimal purchase choice in the first place (he chose something other than the dominant supplier's product). This is best seen in the case of Apple users, who, while using the far less than dominant product, *still must pay high costs to convert* to, say, the leadership platform (Microsoft Windows) *or any new revision*. Whereas classic economics analogizes the common monopoly macro market as one with a single, balanced lever, teetering non-linearly to one side or another (at the "tipping point" of adoption), the MIT market is more complicated than this. This curious compounding of macro tipping with micro (self) tipping more closely resembles an analogy of a double pendulum, initially balanced on both fulcra, only to have both tip to extreme supplier advantage upon user commitment: first the market fulcrum tips (making the product an "expensive" decision) and then the personal one tips, further compounding the problem (making any decision "expensive" per se).

Looked at another way, in the first step, the monopolist gets locked-in as the best choice in the market, then, because of the monopolists tools to make a decision essentially difficult to reverse, the customer locks himself in by deciding per se. In MIT today, the most commonly used product is most often sourced by a single (or a very few) supplier(s). Microsoft's Windows family of operating systems is the classic example we would cite. With it, the monopolist sets the product terms, its (new) features and pricing. If done with managerial acumen and efficacy, the increasingly locked-in customers are held at the precipice of revolt (in the form of a poorly

organized, expensive mutiny to a less embraced alternative product. Did we say anything about "the best product"? No, it is not an issue that matters, practically speaking, if management knows where the precipice is.) MIT-producing monopolists (or oligopolists in some cases) maximize their profits for extended periods at the practical exclusion of alternative (and potentially similar) products.

We believe, when unregulated, this double-open-loop state of affairs does not deliver optimal social benefit. The MIT market has resulted in considerable economic dead weight loss in the markets for popular software products. Clearly, if the goal of public policy is to raise consumer gain (to maximize is an elusive goal!), there needs be a public dialog of effective policy to offset what appears to us to be a naturally occurring, supplier-oriented bias that MIT products (with powerful network effects AND poorly managed complexity) exhibit today.

A New Form of MIT Intellectual Property (IP) Protection May Be Needed

While the market needs a more even (and dynamic) balance between user and producer power, the exact mechanism of an effective policy is not so simple to define. While we are not aware of definitive research to support the conjecture, we believe a new form of intellectual property protection might be in order. It would need to better modulate the preemptive ability of producers to set prices and reduce lock-in phenomenon while balancing long-term profit potential with the recovery of development costs and returning excess to social benefit.

We believe the basis of a new form of IP protection needs to target the underlying cause of double tipping and its locking-in: the ability of MIT producers to uniquely know and thereby limit would-be competitor's products' from interacting with their market dominating ones. This would affect both of the two dominant forms of monologist preemption opportunity: ancillary (application-related) or future (forced upgrade). In networked MIT, how products interact with other is more important than any particular internal innovation or literal expression of an idea (patentability or copyrightability).

Currently, MIT suppliers generally protect their intellectual assets and create competitive market barriers with an ad hoc combination of market (see our previous discussion of these) and technical tools – primarily patents, copyrights and contract-based prevention of public disclosure of source code (somewhat akin to trade secret processes). Such a new approach should complement existing IP tools, not exclude them.

Yet a new vehicle of protection would likely take a hybrid form -- somewhat akin to a copyright but in much less literal an expressed fashion. With a shorter lifetime than a copyright, its actual life might be denominated less in fixed time than in response to the nature of the market evolution -- perhaps using a measure of achieved market penetration or adoption to pace its exclusive life feature to change at or near market tipping. Such a tool of policy, endowed more flexibly than a conventional patent with the ability to respond to the dangers and dynamics of network-related markets at the tipping point could potentially help relieve the recent market defects where dominant products are allowed to exist in practical monopoly form for long periods of time and multiple product cycles. Certainly as dominant MIT producers' profits have plateaued at very high levels while users lack the ability to substitute alternatives[66], we believe one path toward delivering higher social benefit from the MIT markets include examining such a new IP approach.

The Improved "IT Patent"

A new IP definition, based on how data is structured and shared in interfaces between disparate (not necessarily different suppliers') MIT products could fill this void. An "IT Patent" (ITP), might not require the hurdle of technical novelty or discovery as does the patent process, but merely the public disclosure of an interface and its use in product/market context in exchange for the filer's guarantee to exclusive use of that interface methodology in context for a variable period of time. (Interfaces and data structures are not necessarily novel or technically innovative as a patent requires – they are merely held closely as proprietary knowledge and invoked internal to a product for function.) In addition, as MIT markets have been shown to move and adopt products at a much faster rate than the typical lifetime granted a copyright (months or a few years vs. decades or more – commonly a full order of magnitude!), the lifetime of an ITP would necessarily be shorter – long enough to allow the producer to elect to defray and amortize the cost of innovation in the form of producer profits, but not so long as to allow the producer to extract from the market excessive profits for longer than necessary to reward the producer for taking the business risk to create the market.

[66] Interested readers are referred to the publicly available annual reports and other financial filings of Microsoft, Oracle, IBM, SAP and other large MIT companies available from the Security and Exchange Commission at http://www.sec.gov/ for detailed and timely analysis of market dynamics and profitability of these companies.

Naturally, just how short in time and by what measure to modulate the period of exclusivity are the critical unanswered issues. As present day antitrust litigation process has repeatedly shown, the challenge of practically (legally) defining "the market" that a product enjoys, is problematic. However, antitrust law relies on the concept – and one might use a similar process for an ITP. Thresholds of market penetration and adoption rates in combination with market size could be used to create a dynamic indicator of when a network-related MIT market is approaching a tipping point – potentially creating a monopoly in the process *and* the time for unlocking the proprietary protection of the ITP. Prior public access to the knowledge would enable attentive (interested) competitors to rapidly enter the market with alternative *and fully compatible* products. No loss of critical market tipping is necessary even as prices fall and (at least some of present day) dead weight loss is avoided.

But as markets evolve and migrate at very different rates, a context free-measure is likely unworkable. After all, why would ITP's necessarily not apply to consumer as well as commercial markets (consumers get locked-into computer-based products at least as easily as do commercial users do)? Whether the ITP process is a stand alone feature of legal process or integrated into existing antitrust concepts, it will necessarily need to be imputable upon product makers and interface claims that might emerge as ad hoc standards in the market. This would prevent the present day situation where a monopolist appears and he never elected to take out an ITP (or has the option to do so) in the first place. One would be imputed on his market position and products, unlocking the market to others' approach in the process. This may, in fact, be the direction antitrust interpretation and precedent is heading today. However, if it is moving in this direction, why not accelerate the process of policy by formally instigating the device?

The value of such a market-based measure for modulating ITP life-duration also has the advantage of offering the supplier (producer) of such a product the flexibility to use ITP life as a strategic tool – price low to make the market penetration fast by virtue of elasticity (and monetize the gains by reducing spending in other areas such as advertising or support) or leaving prices high and spend more in other places that favorably impact adoption. In this way, the net consequence of ITP-derived protection and potential profitability becomes (remains!) the tool of commerce and not the dictate of bureaucratic regulators. In an ITP-world, customers do not pay excessively (over long periods of time) for past decisions made on imperfect information -- markets can self correct better than they can with the double fulcrum, excessively "sticky", MIT

products available today. At the same time, excessive profits derived from unbridled customer lock-in will be better distributed to support competitive offerings: competitors will be able to see in advance what technical integrations are necessary to make their products more assuredly and truly interchangeable in the marketplace (an ITP would require patent-like public disclosure of the salient features being protected). Done correctly (with certified public information trusts, independent compliance testing organizations, mechanisms of insurance for users – all of which are to be expected to develop in a free, ITP-based market), the ability of a supplier to simply bolt new features onto existing "winning" (monopoly positioned) products would be greatly reduced. Unneeded new product upgrades would no longer be forced on users so as to create artificial extensions of historical (and fully earned-out) innovation. Economic dead weight loss would actively be exchanged for consumer gain.

Phased in carefully, such a new form of IP protection might not be perceived by capital markets as so much "destroying" the present day MIT super-franchises such as Microsoft, Oracle and others (measured by superior price-to-earnings multiples in their stock prices) but as one creating more new opportunities for non-incumbent companies (would-be competitors to the monopolist) to innovate with better chances of success. In other words: more vibrant capital markets, less dependence on a few high-probability investments and better choices for MIT users.

Can Markets Be Incentivized to Self Assemble (through Accelerated Standards Development)?

Public policy-based incentives – franchise rights or tax-based credits -- to stimulate cooperative or accelerated MIT standards development are likely to fail in our opinion. This is because of the natural adoption of standards of interoperability (the core of the network effect in MIT markets) is at total odds with the competitive model upon which western commerce is based. Successful "first mover" products tend to define the standards that dictate network compatibility, not vice versa: it is a first-winner-takes-it-all dynamic (not a well known first-to-market issue per se.). Competitive processes seem to nearly always gestate a product faster than consortia can find consensus and its members act upon it with mutually compliant, competitive new products. Consortia can also violate antitrust collaboration and price control issues. A by

no means exhaustive historical list of such major failed attempts by industry standards-creating committees includes:

- The multitude of wireless communications protocols and physical standards that are evolving today (variations on basic GSM, CDMA, 3G, Blue tooth, Wi-Fi, WiMax, etc.) even as companies on the standards drafting committees race non-compliant products to market, hoping to sweep market adoption and penetration before a standardized technology can come together (and remove the proprietary ownership of the interoperability interface);
- The Ace Processor Consortium of the late 1980's – a self assembled industrial committee created out of competitive drive to develop a DEC-Alpha processor-compatible, multiple-sourced alternative to Intel's microprocessor monopoly of the day. Even as progress was made to create the interoperability and interface testing standards (to assure true compatibility in use), the self-perceived technical leaders of the effort left the fold to beat the would-be standard to market for their own gains (...and no standard nor compatible supplier ever resulted – Intel remained practically unchallenged for another decade);
- The multitude of "flavors" of UNIX and its interfaces that evolved in the 80's and 90's from a single "best idea" that evolved in an incomplete state from a leading industrial R&D lab – only to be productized differently by several different companies into as many competing markets (none of which delivered the long term profit levels of the solely-owned and exploited Windows operating system merchandised by Microsoft);
- Today's competing approaches to the languages of web services and their interoperation at data definition and program standards.

In a simple, practical analysis it appears to us that the capture (and regulation through effective policy) of the standards-formation process is not a likely feasible path toward minimizing the social and dead weight losses extorted by monopolists left to their own devices in the marketplace. While we lack evidence to support the assertion, policy makers will likely do better respecting the Darwinian process that western economies generally rely upon to drive functionally networked products and modulate them for more positive social outcomes afterwards[67].

[67] While unanswered in our work, it remains a valid question as to whether the monopolies left in the alone market in the market (or the delays in the standards that would have driven new, open markets) that resulted from these failed consortia (and many others) should be addressed on separate policy grounds. These might include issues beyond the mere profit attribution that we are targeting -- such as the pursuit of more stable employment (less volatility in MIT production), more dispersed geographic prosperity (broadening of talent pools), etc. These issues are beyond the scope of our work here.

Trusted Computing as a Driver of the Public Policy Dialogue in MIT

The coming specter of "Trusted Computing" (TC) initiatives – known by many different names and acronyms from different consortia and MIT companies – is a challenging one as well as a prime opportunity for the impact of proactive public policy in MIT to take hold. TC is proposed by many as a solution for piracy and other producer risks as it allows (mandates) continuous communication between the users' MIT complex and that of his products' producers. However, in order to make TC work, the process of requiring users to open their systems to carte blanche interaction, inspection (inventory analysis) and revision of programming contents without specific knowledge or specific authorization, TC also necessarily requires MIT asset owners to subjugate their rights to use their equipment to that of their suppliers[68]. TC's adoption (which is very likely if not inevitable in a commercial world dominated by MIT-supplier's desires – see our earlier discussion of producers' interests swamping out user choice in the MIT world today) will likely invoke an enormous amount of political ire and involvement because it will pit two fundamental rights in the Free World against one another: freedom of the owner to use what belongs to him (and maintain his privacy in the process) vs. the baser desires and actions of the MIT producers who will go to nearly any length to protect their intellectual property and gain first-mover advantage in emerging information-based markets[69]. It may be obvious that TC will have long and large impact on changes afoot in industries migrating to usage tolls and protection (music, movies, entertainment in general), but it is increasingly demonstrable that TC's impact will be even higher on existing markets for office automation, legal processes and most all industries that face regulatory or other reasons to retain, control or authenticate documents[70]. For reasons of avoiding public relations problems, excessive legal bills or, in the worst case, damaging valuable brand name(s) in the marketplace, a prudent business manager will be well served to carefully prioritize his company's TC commitments and deployment with the thoughtful maintenance of an external image of actively advocating protection of customer's rights. He will also protect his own rights and assets in the process.

---

[68] Anderson, Ross, "'Trusted Computing' and Competition Policy – Issues for Computing Professionals", Cambridge University, 2004. pp. 2-4.

[69] The reader interested in the policy, product and technical issues related to Trusted Computing is strongly encouraged to explore the writings and research of Dr. Ross Anderson of Cambridge University. While not quoted per se in this section of this brief, his works have heavily inspired the claims and statements made here in summary. His works were available as of this writing on the web at http://www.cl.cam.ac.uk/users/rja14/#Econ

[70] Ibid, Anderson., pp. 6-7.

The transition to TC applications will probably bring some relief from escalating direct security costs, but not necessarily without a high price elsewhere in the organization's or individual's budgets. It will be a difficult transition and most likely will cause the creation of the largest regulation structure to ever be overlaid upon the MIT industry.

In summary, perhaps the most effective near-term policy that can be adopted to raise social benefit from MIT use may also be the simplest one: take a more active role sooner in the growth of new markets. Assuring faster public access and diffuse ownership of the standards underpinning new products' interactions and interfaces to older and adjacent markets and products will be the watchword. Preventing them from being derailed along the way into the monopolist's tool for long- term, over-profit will be the goal. Whether in the form of new IP protection models, legislation, modification of existing antitrust law or otherwise, new policy will very likely become an integral part of the future of MIT, even as it emerges from having been one of the least regulated infrastructures in history.

Conclusions

In his book, Carr raises some extremely important and valid points. Many uses of simple IT products are commodities or close to becoming so. Companies should definitely choose where to invest their IT dollars carefully and not fall into the trap of thinking that large investments in IT will reap enormous rewards or recoup lost competitive advantage. This is largely due to the widespread use of many forms of IT, as Carr advocates, but also several issues he fails to raise: IT projects do not necessarily scale well, so large projects incur more risk. Much of the impact of IT is in the social costs of processes it impacts (or unilaterally dictates).

While we would advocate IT managers consider carefully Carr's advice in their deploying of organizational resources, primarily in the screening of capital spending. Contrary to the overriding theme of Carr's thesis that strategic advantage has vacated IT assets, we believe the opposite: that significant advantage lies within IT strategies for those with specific advantages and assets (unique knowledge, processes, information, talent pools among others). If after applying this primary screen for suitability of IT investment one were to apply the scrutiny of Carr's conservativeness deferral and lowered expectations, one might have the best practice.

Carr neglects or fails to fully investigate or integrate several key aspects of IT into his thesis. He only briefly mentions network effects and their impact on how MIT products behave in the marketplace. Our analysis indicates that network effects have significant impacts on the costs of IT, the development of alternative (competitive) products in the marketplace and severely limit the opportunity (raise the costs) to switch vendors.

Carr also falls into the trap of assuming that IT scales in an approximately linear fashion – or that scale is an issue in the application of IT at all. While he is not far from right when referring to his narrow CIT definition, this definition does not apply to the enterprise-wide systems about which Carr proffers advice. We believe that the significance of the scalability issue could be reduced if the tools available to manage the (ever increasing) complexity of today's MIT products were more effective, more universal (easier to use and learn rapidly) and kept pace with expanding product complexity, innovation and interoperability. This is not to say solving the tool and complexity issue will be simple – it will not. Multiple interactions create uncountable permutations of state, not all of which can be tested on any new product. Complex systems will always have a certain degree of instability and uncertainty. This leaves the possibility that business practices someday will simplify around more robust MIT products, removing these barriers to the incorporation of standard, business-wide, software solutions. With the rise of global companies needing to deal with diverse financial, legal and cultural systems, businesses seem to be on track towards more complexity rather than less. So until companies can come up with simple business rules, their IT needs may remain expensive.

One of Carr's central arguments is that optimal uses of IT are understood and widely disseminated. However, today, nearly a decade after the launch of the Internet, optimal design of websites is still not widely understood. Many companies (such as Amazon) continue to experiment with techniques of using MIT to increase their productivity (through higher conversion rates) while less than savvy organizations continue to loose relative position in the market from less than aggressive use of MIT (losing sales to competitors due to poor navigation tools and layout styles of their websites for example).

Finally, Carr advocates three basic tenants of CIT decision making:

1. defer and cut (manage) the costs aggressively;
2. Assess it from the perspective of potential losses from implementation (downside risks); and
3. Follow the approaches of the leaders – but don't be one of them.

Another expert in the area has identified the issues that he sees as challenging companies in the face in deploying MIT. His list is longer – and more to the real point in our view. It includes the three that Carr is fixated upon (#2, #5 and #10) but adds seven more of equal importance and includes[71]:

1. Constant technical change
2. Technology assessment and transfer
3. Co-evolving business and technology
4. Realizing business value
5. Managing costs
6. Increasing technical complexity
7. Dependence on experts
8. Developing and delivering systems
9. Systems integration
10. Costs of failure

Further, Gurbaxani believes (as do we) that[72]:

- The Internet and other information and communications technologies will provide substantial opportunities for firms to achieve gains in labor productivity and multifactor productivity
- Future gains will increasingly come to companies engaging in networked partnerships
- Continued [MIT] investment should lead to continuing productivity growth

Carr will most likely and simply have to relax the unrealistic constraints on his definition of IT before he is likely to evolve any profound discoveries or have the right to claim his sacred high ground as the guru of IT prognostication.

[71] Gurbaxani, Vijay, "The Impact of IT Investment on the US Economy", CRA Conference, July 13, 2004, slide 29.
[72] Ibid., slide 35.

Bibliography

Anderson, Ross, "Trusted Computing' and Competition Policy – Issues for Computing Professionals", Cambridge University, 2004.

Brooks, FP. *The Mythical Man-Month, Anniversary ed.*, Addison-Wesley. New York, 1995.

Carr, Nicholas G., "Does IT Matter", Harvard Business School Press, Boston, Massachusetts, 2004.

Carr, Nicholas G.,"Does Software Matter?" May 2004. http://www.darwinmag.com/read/050104/carr.html

Carr, Nicholas G., "IT Doesn't Matter." Harvard Business Review. June 2003.

CNN.com posting on February 23, 2000 at: http://archives.cnn.com/2000/TECH/computing/02/23/microsoft.china.idg/

CollegeBoard.com, "Career Browser: Rail Transportation Workers". Retrieved on November 21, 2004 from http://www.collegeboard.com/apps/careers/0,3477,32-244,00.html

Comstock, Keith. "Does IT Matter? Information Technology and the Corrosion of Competitive Advantage." Public CIO. August 2004. http://www.public-cio.com/story.php?id=2004.08.11-91143

Constantine, LL and LAD Lockwood. *Software for Use*. ACM Press, New York, 1999.

Davenport, T.H., Living with ERP. CIO Magazine December 1998. http://www.cio.com/archive/120198/think.html

Duhl, J. (2003). "Rich Internet Applications". Retrieved November 30, 2004 from http://download.macromedia.com/pub/solutions/downloads/business/idc_impact_of_rias.pdf.

The Fabless Semiconductor Industry Association, "FSA's quarterly Global Fabless Financials Report", December 8, 2004. see: http://www.fsa.org/pubs/gffr/default.asp#files

Farber, Dan. "The end of IT as we know it?" Tech Update, May 2003

Fortune Magazine, "The Fortune 500 – 2004", see: http://www.fortune.com/fortune/fortune500

Good, Robin. "Open Office vs Microsoft Office (according to MS)." March 2004. http://www.masternewmedia.org/news/2004/03/26/open_office_vs_microsoft_office.htm

Gurbaxani, Vijay, "The Impact of IT Investment on the US Economy", CRA Conference, July 13, 2004. See: http://www.cra.org/Activities/snowbird/2004/slides/gurbaxani.pdf

Hacket Group. "Three Out of Ten Major IT Projects Still Fail According to Latest Research from The Hackett Group". Retrieved November 10, 2004 from http://www.thehackettgroup.com/ab_pr.asp?id=289.

IBEW/NECA. "The Committee for Electrician's Certification – FAQ's". Retrieved on November 21, 2004 from http://www.ibewnecaforcertification.org/faqs.htm#f.

Intuit, Inc. (2004). QuickBooks Home: The Financial & Accounting Software Resource for your Business. Retrieved November 21, 2004 from http://quickbooks.intuit.com/.

Koch, Christopher. "The ABCs of ERP." March 2002. http://www.cio.com/research/erp/edit/erpbasics.html

Lee, D. "Building a Truly Data Driven Enterprise: The Amazon.com Experience". University of Washington Computer Science Department Colloquium Oct. 28, 2004. Retrieved November 28, 2004 from http://norfolk.cs.washington.edu/htbin-post/unrestricted/colloq/details.cgi?id=292.

Lohr, Steve. "Does Nick Carr Matter?" http://www.strategy-business.com/press/article/04213?pg=0

Lutchen, Mark. "Managing IT as a business: A Survival Guide for CEOs." Wiley. 2004

Macromedia, Inc. (2004). Macromedia – Business Solutions. Retrieved November 21, 2004 from http://www.macromedia.com/resources/business/.

The McKinsey Global Institute, "U.S. Productivity Growth:1995-2000", October 2001.

Microsoft Corporation Annual Report, 2003

Morgan, Tony. "Business Rules and Information Systems: Aligning IT with Business Goals." Addison-Wesley. 2002

Potok, TE, M Vouk and A Rindos. "Productivity Analysis of Object-Oriented Software Developed in a Commercial Environment." *Software – Practice and Experience*, Vol. 29, No. 10, pp 833-847, 1999.

PricewaterhouseCoopers / Oblix. "Best Practices for Sarbanes-Oxley Compliance: Enforcing IT Controls with Identity and Web Service Management." October 2004. http://www.oblix.com/c/media/webcast/102104/files/loby.html

Sauer, Chris and Cuthbertson, Christine, "The State of IT Project Management in the U.K. 2002-2003". Retrieved November 10, 2004 from http://www.computerweeklyms.com/pmsurveyresults/surveyresults.pdf

Schwartz, Jonathan. "Commodity, Competition and Evolving Technology." SystemNews. October 2004. http://sun.systemnews.com/articles/80/4/feature/13758

The United States Securities and Exchange Commission, "SEC Filings & Forms (EDGAR)" available at: http://www.sec.gov/edgar.shtml

Slater, Derek. "An ERP Package for You … and You … and You…and Even You." CIO Magazine. February 1999. http://www.cio.com/archive/021599/erp.html

Smith, Howard and Fingar, Peter, "IT doesn't Matter – Business Processes Do",. Meghan-Kiffer Press, Tampa, Florida, 2003.

Standish Group. "Latest Standish Group CHAOS Report Shows Project Success Rates Have Improved by 50%". March 2003. Retrieved November 28, 2004 from http://www.standishgroup.com/press/article.php?id=2.

Standish Group. *The Chaos Report*. Report of the Standish Group. 1994

State of Minnesota Board of Electricity, "A Brief Look at the History of the Board of Electricity". Retrieved on November 21, 2004 from http://www.electricity.state.mn.us/Stat_rul/Brd_hist/.

State of Washington, "Facts About State Certification for Electricians" August, 1999. Retrieved on November 21, 2004 from http://www.lni.wa.gov/IPUB/500-004-000.pdf.

Stewart, Thomas A., "Does IT Matter? An HBR Debate", Letters to the Editor, Harvard Business Review, June 2003.

Stutz, David. "Some Implications of Software Commodification." Creative Commons. 2004. http://www.synthesist.net/writing/commodity_software.html

Varian, Hal, "Intermediate Macroeconomics –A Modern Approach", WW Norton & Company, New York, 1999.

Varon, E. " For the IRS There's No EZ Fix". *CIO Magazine*, April 2004. Retrieved November 29, 2004 from http://www.cio.com/archive/040104/irs.html.

Vosburgh, J, B Curtis, R Wolverton, B Albert, H Malec, S Hoben, Y Liu. "Productivity Factors and Programming Environments," *Proceedings of the 7th International Conference on Software Engineering*, IEEE Computer Society, 1984.

Wharton School of Business. "Information Technology: Value creator or commodity?" CIO Magazine. March 2004. http://searchcio.techtarget.com/originalContent/0,289142,sid19_gci955679,00.html

Wyman, Bob. "Is "SuperGoogle" the future of search?" May 2004. http://bobwyman.pubsub.com/main/2004/05/