Virtual Migration

The Programming of Globalization

A. ANEESH

DUKE UNIVERSITY PRESS | DURHAM AND LONDON | 2006

Printed in the United States of America on acid-free paper ♾

Designed by Heather Hensley

Typeset in Adobe Garamond by Tseng Information Systems, Inc.

Library of Congress Cataloging-in-Publication Data appear
on the last printed page of this book.

Contents

enforced by the U.S." as well as the availability of high-speed data communication links: "In 1988, the percentage of onsite development [through body shopping] was almost as high as 90 percent. During 1999–2000, the offshore component is expected to increase to about 45 percent of total software exports" (Nasscom 2000, 22). In 2004 the virtual flow of labor finally surpassed the on-site performance of work. The next chapter analyzes how virtual migration provides a fresh horizon for viewing the changing forms of system integration through a different arrangement of space and time.

CHAPTER 4

Virtual Migration

> All that is solid melts into air . . . —Marx and Engels (1848)
>
> Electronic aids, particularly domestic computers, will help the inner migration, the opting out of reality. Reality is no longer going to be the stuff *out there*, but the stuff inside your head. It's going to be commercial and nasty at the same time.—J. G. Ballard (1971)

Not all that is solid has melted into air, and poor reality, though besieged and beleaguered, still seems to matter. Wars are still written in blood. Self-assured skyscrapers and streets crowded with ostentatious cars are all quite solid and real. Perhaps the question is not about the solidity of reality but rather about its dependence on programming schemes, which precede and govern it. From simulated models of cars to models of economy, the production of solid reality increasingly traces its roots to the virtual. There is a rising tendency to see the world through virtual schemes and then act in ways that make the real conform to the virtual (Carrier and Miller 1998). Economic models do not simply describe an existing, empirical state of affairs; rather, they are performative. For instance, in the world of finance the Black-Scholes-Merton option pricing equation altered traders' behavior and pricing patterns in a way that made itself more true over time (MacKenzie 2003). The equation did not merely represent but performed the real. A simpler example would be the theoretical assumption of economic models that humans are rational, competitive, and calculating individuals. A competitive system based on this model produces such individuals by making difficult the survival of noncompetitive and noncalculative behavior within the system.

While the real may be increasingly virtual in conception, the virtual sounds more and more real and physical. Cyberspace is full of physical, geo-

graphical, and even organic metaphors: site, space, property, window, Navigator, Explorer, superhighway, shopping carts, traffic, spiders, viruses, and worms—the list is long. Lest you think these are mere metaphors, there have been lawsuits in the United States construing cyberspace as physical space: for example, eBay charged Bidder's Edge with "trespass" in a lawsuit for collecting information from its "site" with the help of spiders or bots (programs that run on remote machines), which entered eBay's website despite the signs prohibiting "entry." As absurd as it may sound, eBay's lawyer argued that trespass law is especially suitable to virtual space, because it is easy to establish clear signs about which activities are prohibited and because the site is owned by eBay not merely in its virtual form but also in a physical form (as machines or servers).

Virtual migration is equally complicated. When labor moves from machine to machine across nations, it is hard to capture in conventional categories (as in the case above), since conventional labor migration is generally situated within the framework of body migration. The continuous revolutionizing of the instruments of production, distribution, and consumption has enabled a new labor regime in which labor moves and migrates without the worker's body. The physical ways of performing work are not going to vanish: just as the theater did not disappear after the cinema turned acting bodies into moving pictures, the embodied migration of labor will perhaps never be replaced by virtual labor flows, as some services can be performed only with the body. Virtual and physical migrations are drastically different species in their mode of global integration, even if they tend to produce similar economic effects in terms of jobs and wages. They are as different as cyberspace is from physical space, despite court cases and metaphors to the contrary.

The rules of the road for virtual migration differ from those for physical migration such as body shopping. When a programmer in Delhi accesses a client's machine in New Jersey, we may define her presence in the United States as virtual presence or tele-presence, but clearly not as physical presence. In fact, computer networks—pathways of virtual migration—are so insensitive to geography that it is almost impossible to determine the exact physical location of the machine without extraneous knowledge. The network is technologically indifferent to physical location. Knowing the location of the client's machine may even be irrelevant to the programmer's task.

Network Interface Cards may have "physical addresses" and computers may be assigned IP addresses, but these are logical addresses that identify the machine on the network, not in real space. True, the network's design may be changed to add a provision for filtering and screening for specific geographic tags in packet headers. One could, for example, instruct the machine to refuse access to requests originating from the domain ".in" (India). But one should not think of the ".in" home page as a physical home, because a person may be sitting anywhere while managing the ".in" website. The body is never in cyberspace in any practical sense, and constraints are not physical but only logical and programmable. Indeed, companies tend to distribute their resources on many machines around the world that can be seamlessly accessed by the worker without knowledge of their exact physical location. There is no consistent homology between cyberspace and physical space (Mitchell 1996), except that both may be designed, integrated, and controlled through programming schemes (e.g., street planning schemes for physical space or software scripts for cyberspace). Cyberspace, even in its drastic difference, does connect and integrate physical worlds far apart, linking labor from one world to another.

There are two specific aspects of the virtual migration of software labor: spatial integration, which decouples work performance and the work site, and temporal integration, a real-time unification of different time zones, which underscores the importance of the temporal point of the revolving and rotating earth where one is located.

Spatial Integration

The programming schemes of virtual integration do not entail transporting the body from one place to another; instead, they keep the body in one place while performance travels to other locations. Virtual labor migration is not a single scheme of integration. It may range from the real-time work performed on mainframe computers and servers in the United States by a worker based in India to a distributed work design, allowing a firm to be geographically dispersed, without a central work station, among several sites throughout the world. Such a firm may divide up its work to take advantage of cheaper labor, diverse skills, and different time zones. Different modules of the system can be independently developed and electronically combined.

This programming labor is both an effect and cause of global integrations. On the one hand, virtual programming labor owes its origin to the transnational capitalist drive to harness cheaper skilled labor outside national borders by novel means; on the other hand, this form of labor—with universal and systems-level programming schemes—further integrates pockets of the world by tightening the interconnectedness of labor, capital, and administration in functional supply chains. One of the project managers in New Delhi described how the managers helped the Gap to develop a new information system to track its orders and connect its globally scattered vendors: "What Gap does is, like, all their clothes are produced in the Third World, Latin America, India, Bangladesh and all these countries. They have vendors in all these places, so purchase orders are created between these vendors and Gap, and you want to purchase so many goods of a certain style, cut, size, and this order is sent out to these vendors. So, the process of automation is purchase order creation, and then getting the goods back and things like that. We were involved in the development activity. Gap had given us a complete project. We cloned their environment on our own mainframe. We . . . developed the complete software, and then I was in the U.S. implementing it and making changes." Offshore programming labor thus is not merely a consequence of system imperatives. By creating software-based programming schemes that integrate various units around the world, it is a major force behind transnational system integration. It is responsible for creating integrative spaces that ground a variety of work flows and transactions related to banking, insurance, financial services, manufacturing, retail, distribution, communications, government, transport, and hotels.

With enough bandwidth, labor in India has been integrated into American corporate sites twenty thousand miles away, reducing an enormous spatial gulf to a matter of faster or slower transmission speeds. In 2004 there were more than eight hundred firms in India providing virtual labor to corporations in the United States and other countries. Nearly a decade ago the *Economist* (1996) reported that more than a hundred of America's top five hundred firms bought software services from firms in India. In 2001–02 about half a million workers were employed by the Indian IT industry, which is mostly oriented to foreign markets and has the majority of its clients in the United States. Over 200,000 workers were employed in global

software development, while 106,000 others provided a staggering array of related services, including customer service, banking, financial and credit analysis, back-office accounting and other operations, data entry, data conversion, transcription, translation, technical support, animation, engineering and design, website development and maintenance, remote education, electronic marketing, data search and integration, market research, documentation handling, and human resource services like employee benefits and payroll. The National Association of Software and Service Companies, or Nasscom, claims to be a "truly global trade body with around 850 members, of which nearly 150 are global companies from the US, UK, EU, Japan and China" operating from India. Most of these companies are global because the domestic market for software and services is very small compared to the offshore market, which offers major work contracts for Indian-based firms. Close to 60 percent of the revenues of the software and services market flows from foreign sources, jumping from $9.55 billion in 2002–3 to an estimated $12.2 billion in 2003–4 (Nasscom 2004).

We may understand the scheme of on-site labor that physically migrates from India through body shopping or otherwise as one mode of labor supply, and online labor that migrates from India virtually as another. Contrary to common categorization, I do not place the two in separate realms of migration and trade. In fact, both forms of labor are combined at times in carrying out a single project. A firm in India might send two or three systems analysts to the client's site in the United States for a short period, so that they might gain a first-hand understanding of the project and discuss systems design. These systems analysts then help to develop the project in India while remaining constantly in touch with their client, who can monitor the progress of the project and provide input. Once the project is over, one or two programmers fly back to the United States to test the system and oversee its installation.

A variety of transnational business strategies have evolved to integrate and govern dispersed labor. In many cases American companies have either entered into joint ventures with Indian companies or subcontracted part of their work to Indian companies while maintaining continuous electronic contact. A large number of Indian software professionals are connected to the mainframe computers of their American clients from a remote location,

completing a series of back-end jobs every day when the system in the United States is not as busy.[1] Some large insurance companies in the United States that receive a huge number of claims get their processing done through such a virtual mode of spatial integration. This ability to perform work at a place other than the location of the laboring body is becoming more and more common, not only in the context of third-party outsourcing but also among companies based in two or more locations.

When problems or glitches occur, however, the model does not always work smoothly. Many companies combine online labor with onsite support. One systems analyst cited an instance when his team supported Citibank operations: "Citibank had [changed] all their retail business; there were a lot of changes required in the programs already existing like day-to-day maintenance . . . One way is that they have their own people do it. The other way is how the work in their bank in Japan was done . . . There was a team of people working in India, and there was a project manager on site [in Japan]. I was the project manager, I would take work from the Japanese managers and I would send it offshore to India . . . So any changes, any production problems, anything will immediately come to [notice through] me."

To avoid problems of coordination, Indian firms developed what was known as a 75–25 model. While 75 percent of their work force remained in India, they established a small office in the United States, comprising the remaining 25 percent of the work force, to coordinate and mediate between the Indian team and the American client. One major software company in Bombay, which also had a small unit in the United States, provided twenty-four-hour information systems management for insurance claims processing to a major American insurance company by accessing its mainframe computer directly from India. One programmer, who moved briefly from Bombay to North Carolina while working for the same company, described this software work as follows: "The Bombay team can directly access the client's mainframe. Usually what we have is maintenance project, and we support [the firm's] insurance business for twenty-four hours . . . There are different groups in [the firm], and [we] support most of them . . . So, suppose someone is claiming [insurance] money from [the insurance company] due to some accident. He would go to [the firm's] agents, [who] would enter the data on [Customer Information Control System] screens, [inputs] like where

this accident happened, what's the cause, and other details of the accident. And when this information is entered on CICS screens and the daytime is over in the USA, that information is captured and is written to a file, which is the input for our nightly batch processing."

In terms of work performed, there is no critical difference between onshore labor engagements (North Carolina) and offshore ones (Mumbai), even if the first is characterized as migration and the second as trade. Software firms see on-site and online methods of labor supply as interchangeable "components" of the same transnational programming labor. Granted, the classification of onshore labor as "migration" and offshore labor as "trade" serves a function in the existing institutional setup of nation states; virtual labor remains a systemic innovation to negotiate national borders in a different way, using the same pool of Indian programmers.

In addition to constant support, maintenance, and enhancement of information systems, Indian software companies also work on independent software projects by cloning the client's systems environment—a unique feature of information technologies—and then redesigning and reengineering the system. The client could be a bank, an airline, or a manufacturer.

Another strategy of integration uses a third firm to connect through its common software platform a firm based in the United States to a firm based in India: "[This firm] helps companies source and deploy virtual skills to deliver work on time and on budget. Through [the firm's] technology, customers gain access to an outsourced workforce that is globally competitive, cost-effective, high-quality, fluid and flexible. Through [the firm's] proprietary technology platform, the geographic constraint is removed from the traditional, bricks-and-mortar staffing paradigm, allowing companies, for the first time, to deploy a virtual staff."

In some cases, American companies set up subsidiaries in India for various purposes, ranging from high-end research collaboration with American-based teams (a common pattern with Adobe and Microsoft) to low-end programming work to converting already documented tasks into software, as well as providing technical, financial, and customer services on line and on the phone (as is often done by the American International Group and GE Capital). Instead of using the binary terminology of "high-end" and "low-end" programming, I have elsewhere described the whole range of program-

ming skills along a spectrum from less saturated skills to more saturated skills (Aneesh 2001). I have proposed that the problem of skill should be analyzed in terms not of high or low skills but of the closure of play space in the structure of skills, which I term "skill saturation." This is a phenomenon characterized by an absolute predictability of procedure and outcome, resulting from an exhaustive ordering of various components of skills, and the elimination of all irregular spaces of work. On a spectrum from higher saturation to lower saturation, one could identify two kinds of programming: programming for a definite task, and programming for an imagined outcome. The first implies relatively saturated skills, as it consists chiefly of translating and codifying an already known task (such as an accounting or banking procedure) into a software package, where all the elements of the system are known and the creativity consists only in creating a simple and efficient program. Writing software for an imagined outcome—for example, designing a chip with an imagined higher processing power—is, on the other hand, a task for which no procedure has yet been identified, implying a relatively unsaturated skill. Software companies in India are engaged in both kinds of work. At the less saturated end, one CEO of an Indian subsidiary of an American company claimed that the kind of work done in India might be more advanced, dealing with more abstract, newer, imagined problems: "Well, we are part of [the firm's] engineering team. The work that gets done here is the same as is being . . . in the U.S. with a little bit of difference. The difference might be that the work we do here is slightly better quality than what gets done in the U.S., . . . [where] there is a bunch of work that's maintenance, it's maintaining old components and old tools. We would not take up too much of old things; we would start from scratch, newer things . . . So in that sense the kind of projects that we are getting here are slightly better quality than what people probably have on an average in the U.S."

The less saturated software engineering forms a smaller proportion of programming work carried out in India, as the majority of Indian companies are engaged in work involving more saturated skills. At the more saturated and routine end of the programming spectrum, one software professional in Noida, India, mentioned writing and customizing software for already existing tasks, like banking procedures: "We support your daily requirements for banking applications like daily branch opening, your account handling, your money transfers, everything, the routine tasks for which there's a need to build the software. It's very routine because most rules are documented. You just have to implement those business rules into software programs."

Toward the unsaturated end of the programming spectrum were some high-tech American and European firms (Adobe, Microsoft, Lucent, Siemens, and Nokia) that connected their spatially dispersed teams through data communication links, integrating their parts as if they were adjacent rooms in the same building where two teams might be working on the same project. "There are twenty people working in the U.S. and twenty people in India," said one CEO, describing the mechanisms of online collaboration between his team in India and its counterpart in the United States. "They are doing different things. But the mother ship is the same; it goes into the same product. So you are working on the same database, you are working on the same code. You are working on the same thing . . . we are sharing . . . a data server [and] we are working on those systems. Except for the fact that we are in India, we could be sitting across the room from those people and working."

The phrase "sitting across the room" was not a passive statement about global connectivity; rather, it was a continuing activity that joined the discourse of integration with the continuous production and refinement of corresponding equipment and apparatus. All major companies persistently invested in installing and upgrading data communication lines that allowed for virtual work flows. The emerging labor regime depends heavily on this mode of integration, which affords a new architecture for transnational strategies of organizational governance. More and more companies are investing in direct and dedicated communication links, as reflected in the following statement of the vice-president of another software firm in Noida, India: "Since we have a dedicated satellite, and we have paid for our lines, we are not dependent on any indirect channels; we have direct channels, if you want to speak to the U.S. I can put you through within half a minute; I don't even have to dial the [area] codes . . . so that's the kind of dedicated link I'm talking about. It's called the link line; that's how our server, our computers are connected." Globalization, in view of these concrete practices, does not come across as a passive force of history or an inevitable phenomenon happening behind our backs. Rather it is enacted, implemented, and brought about through

such integration schemes, resulting from the actions of high-level executives who consciously deploy technologies of integration. These technologies are constantly refined and upgraded.

Programming labor in India combines with other kinds of labor to provide services such as legal database support to large American law firms. Many Indian software firms armed with locally trained lawyers work closely, though remotely, with law firms in the United States to develop software-based databases of easily accessible legal information, equipped with common criteria to incorporate new documents. Similarly, in development of digital content such as animation, the content provider, while based in the United States, is able to continually review the project being developed in India. Other projects that employ the continuous integration of spatially separate units involve the development of web content, computer aided design and manufacturing (CAD-CAM), and saturated skills like medical transcriptions for American doctors who electronically send dictation through high-speed data communication lines. Since the late 1990s IT-enabled services in general, and call centers in particular, have added enormously to virtual labor flows across the national borders of India and the United States. Included in such virtual flows are customer service interactions through call centers, business process management, back-office operations, insurance claims processing, medical transcriptions, legal databases, payroll, human resource services, web site services, data digitization, online education, and digital content.

Although IT-enabled services consist of mostly saturated skills, they provide a vivid example of virtual migration. Using dedicated, leased telecommunication lines, workers at remote call centers in India connect to the parent organization through voice links and online database access. This form of transnational, virtual labor flows is as direct as telecommuting within the boundaries of the national state. For example, when a bank customer in the United States calls the bank's customer service number, the call is immediately redirected to Indian locations without the customer's knowledge. The respondent in India taps into the customer's history and customer-owned product attributes, using the latest technological developments in global online information systems and web-enabled databases. Call centers in India handle complaints, technical queries, customer relations, account management, sales leads and follow-ups, telemarketing, credit and billing problems, market research, and database development. These services are used

by airlines, hospitals, law firms, utilities, and companies engaged in finance telecommunications, insurance, high technology, and manufacturing. Some companies set up their own subsidiaries to harness Indian labor. GE Capital has a subsidiary in Gurgaon, near New Delhi, that manages payroll accounting, call centers, mortgage-based loans, and insurance claims for consumers in the United States.

Lest I gave the impression that this kind of labor integration takes place only between two countries such as India and the United States, I should point out another transnational strategy that distributes software development to many sites around the world. Each site might work on a different stage of a product's life cycle; or the sites might divide responsibility along developmental lines, with one working on the application engineering, including specifications analysis, design, and integration and acceptance testing, and another site implementing system components; or the work might be distributed according to the qualifications of the team at each site. One Indian team employed by a subsidiary of an American firm worked on a project with five teams based in such faraway places as Tokyo, Singapore, Sydney, Beijing, and Arlington Heights. The global networking of labor has increasingly become the norm rather than the exception in software development.

Managed by the programming schemes of Internet protocols and applications, networks increasingly dominate socioeconomic life. "For the first time in history," Manuel Castells asserts, "the basic unit of economic organization is not a subject, be it individual (such as the entrepreneur, or entrepreneurial family) or collective (such as the capitalistic class, the corporation, the state) . . . the unit is the network, made up of a variety of subjects and organizations, relentlessly modified as networks adapt to supportive environments and market structures" (Castells 1996, 214).

The global mobility of labor increasingly depends on networks of this sort. As corporations evermore exist in direct networks with their suppliers and customers, it is less meaningful to talk about economic organization in terms of separate units rather than networked labor, information, and capital. The networking strategies have imparted enormous flexibility to the system, but to reap the full benefits of network flexibility, "the corporation," Castells (1996, 176) points out, "had to become a network itself and dynamize each element of its internal structure." Interorganizational computer

networks have attracted particular attention among organizational analysts (Benjamin, Delong, and Morton 1990; Benjamin, Rociiart, Morton, and Wyman 1984; Keen 1986; Meyer and Boone 1987; Zaheer and Venkatraman 1994), who have focused especially on electronic data interchange (EDI) in business-related transactions. However, as I have tried to show, EDI now covers real-time work flows, including exchanges related to design and manufacturing activities (Hart and Estrin 1991). Castells's attempt to understand the emerging economic regime in terms of networks has two advantages: first, it avoids the national-global dichotomy that has clouded most debates on globalization, as I pointed out in chapter 2. Second, the idea of networks enables an analysis based on flows, rather than isolated units, entities, and individuals, thereby bringing out the intrinsic interdependence of different economic processes and their relationship with the social world.

However, discussions of organizational networks tend to be economistic, focusing on possible efficiencies, competitive advantage, coordination, and relative transaction costs for corporations (Benjamin, Delong, and Morton 1990; Benjamin, Rockart, Morton, and Wyman 1984; Hart and Estrin 1991; Meyer and Boone 1987; Zaheer and Venkatraman 1994). The same is true of such conventional concepts as outsourcing and subcontracting. Inserting the practice of virtual labor migration into traditional schemata and concepts—such as outsourcing and subcontracting—makes it difficult to develop an understanding independent of corporate needs and motives. Surely, a business may need to "outsource" and "subcontract" work to reduce the size of its permanent work force or capitalize on globally dispersed low-cost labor. But any analysis based on functional choices made by corporations within the competitively structured field of capitalism tends to hide the politics of technologies. Public and corporate choices for particular technologies are also choices for particular social orders (Winner 1986), in this case a global social order with social effects going beyond corporate interests. A concrete description of the practice may reveal social consequences and adjustments that are not easily visible in analyses based solely on the perspectives of business needs and processes.

The language of "networks," despite its many advantages, also seems a little too neutral and objective, much as it does in the computer sciences, obscuring relations of power and governance in the emerging regime. To

discern the social consequences of growing system integration through networks, we need to closely look at adjustments made essential in social life by the velocity of electronic transmissions, which transform the space-time dimensions of social and work life. Recent breakthroughs in the transnational organization of labor are due to advances in the velocity of work flows, as well as to such developments as the programming of different organizational possibilities. These developments are not without their internal problems. Horizontal global integrations are beset with snags. They fail to work as often as they succeed.

Horizontal Disconnect

I asked the vice-president of a software company in New Delhi why on-site services had not completely given way to online services. "Because the management [in U.S. corporations] is many times lazy in providing complete systems specifications," he replied, a statement that is only partly true. Of course firms must precisely formulate requirements for transnationally developed software and agree upon modest quality standards, usually by entering into a contract; they must develop detailed specifications and project definitions, including methods, schedule, scope, and deliverables, clearly understood and agreed to by both parties. But there are many other reasons why the physical flow of migrants continues despite virtual labor integration. The instantaneous flow of labor from India to the United States simultaneously facilitates and hampers work organization along space and time dimensions. By decoupling work performance from the site of work, virtual integration overcomes distances, but only by losing the rich, multilayered, face-to-face interaction provided by place-based integration at work. Companies attempt to recover this loss of spatio-temporal immediacy by various mechanisms such as direct phone connections, video conferencing, electronic message boards, and instant messaging, as well as limited on-site visits. Indian software professionals are often flown to the United States for a brief initial meeting with the client, as it is not always possible for the client to formulate complete project specifications and communicate them online. Similarly, at the end of the project, despite the online access to completed software projects, senior software engineers typically return to the United States to ensure the success-

ful implementation of their projects. Even short business trips for emergency meetings can create problems related to visas, work restrictions, or different systems of social security. Schemes of integration based on the nation-state model come into conflict with global integrations. Physical migration thus must be planned in advance.

Physical migration is also required if knowledge of the project requires understanding other components of the system and the enterprise. "[When] you are developing a project, which interfaces to a lot of other projects within the U.S.," an informant noted, "those interfaces have to be studied there only . . . You can't just ship it here. If you ship it here . . . it's very difficult for the people to understand what the interfaces are; there [could be] ten different departments interfacing with this department." Often face-to-face communication is very important for a quick resolution of problems. Even when problems are resolvable by e-mail or phone, software companies in India depend on senior project managers who have been to the United States and understand American styles and speech. Virtual integration cannot instantly overcome sociocultural differences. Another software professional admitted that it was difficult for programmers sitting in India to understand what the client wanted, because of their limited knowledge of the United States in general. One responsibility of the senior project manager is often to translate phone and e-mail communications: "[In the U.S.] people can resolve things through meetings, through discussions, through lectures, through face-to-face interaction," a project manager explained. "[Here in India] communication does become a problem . . . the entry-level people are the people who haven't been to the U.S. or who haven't been abroad; they can't really relate . . . So you always require a senior person, a project manager who can relate to what the Americans mean."

It is not surprising, as mentioned earlier, that many Indian software companies also set up smaller offices in the United States for effective coordination and communication between their developers in India and the clients in the United States. "There is something to be said about face-to-face communication," another programmer emphasized, "being in face-to-face communication, and being part of the team physically is very different from being just online." Face-to-face encounters, especially informal corridor talk at work, are surely of a different order. Karin Knorr Cetina and Urs Bruegger

(2002) have introduced an appropriate distinction between embodied presence and response presence. Embodied presence is always face to face; but response presence describes situations in which the participants are capable of responding to one another and common objects in real time without being physically present in the same place. Being interactionally present through the screen does not mean that one can discuss things over a cup of coffee with colleagues across the globe or share a hearty laugh with the team on the other side of the video screen; audiovisual links fail to carry jokes over to the other side. At current technological levels, the screen does tend to split the teams, making it improbable for a single social space—or at least the usual social space—to emerge. In the absence of what Alfred Schutz (1973) described as the "interlocking of the glances" and the "thousand-faceted mirroring of each other," many sociocultural differences, including different first languages among developers in remote locations, are exacerbated in audiovisual gatherings. But we must also not forget that the screen allows a kind of gathering not possible in face-to-face situations. Before the advent of software platforms, it was not possible for so many workers, managers, and developers to watch the same screen and make changes to the same database. In the realm of financial integration, Karin Knorr Cetina and Urs Bruegger (2002) rightly assert that only the screen's emergence made it possible for traders to simultaneously watch the market.

Virtual migration presents a problem of synchrony across sociocultural and physical divides. In the heat of continued success, one must not forget that the Indian software industry also hides many failed projects and abandoned transnational business relationships. Heeks, Krishna, Nicholson, and Sahay (2001) define quite accurately the question of success or failure in global software relationships as "synching or sinking," defining how congruence must be achieved along six dimensions (the initials of which form the mnemonic "cockpit"): coordination and control systems, objectives and values, capabilities, processes, information, and technology. Organizational relationships across the globe must synchronize themselves along these dimensions or they will sink. Clearly, the synchronization must be programmed by structuring work with clear specifications; it must also be performed by making attitudes, values, and organizational behavior work in unison. To develop successful software systems together, the client in the United

States must transfer some of its knowledge and technology to the subcontractor in India, as well as synchronize its management systems and organizational culture. Studying the relationship between the American company Global and its Indian counterpart Shiva, Heeks, Krishna, Nicholson, and Sahay (2001, 56) found that Shiva had a relatively personalized and subjective management culture while Global's stressed objectivity and accountability. "It took enormous efforts before the Shiva project leader would produce a standardized monthly progress report, and Shiva staff refused to participate in Global's employee satisfaction survey." Evidently sociocultural differences could not be easily erased.

When problems occurred because of sociocultural differences, the solutions increasingly took a technical turn. The introduction of such external quality standards as the Capability Maturity Model (CMM), developed by the Software Engineering Institute at Carnegie-Mellon University, and the ISO 9000 series, developed by the International Organization for Standardization, served the purpose of "objectivity" over the years. These standards were used by software firms in India as a label for their quality and performance. There were also other ways to program some sociocultural differences out of the system, though not all. While it may be difficult to extract a programmer's performance report from an Indian project manager, it is increasingly possible to program some functions of monitoring into the system itself. For example, software development platforms have acquired a defect-tracking function that can identify the number of errors made per thousand lines of code.

Despite the development of external standards and the coding of governance structures into software platforms, there have been numerous barriers, both social and physical in nature, that have hindered systemic integration. A temporal lag in data communication links does not help the cause of synchrony. One CEO complained of the slow and overpriced connections provided by the Indian state as he discussed his plans to install, as far as the law permitted, his own fiberoptic lines, which were vital for the virtual integration of remote sites: "Our work—because we are completely integrated—requires something like 100 Mbps or 200 Mbps; what we get is 256K at an exorbitant price. And because it's going through a satellite, there is a delay. There is a ping time of 300 or 400 milliseconds,[2] which is too high for our

kind of work. So, what we want is go all the way on fiberoptics, so that we can ping at 7 or 8 milliseconds. That'll improve the response of a lot of our systems." The system has reached a stage where a delay of 300–400 milliseconds, hardly noticeable to previous generations, has become crucial in the programming of globalization. Quite like technologies of material mobility, virtual migration must contend with the question of time and speed as the forever receding frontier of spatial conquest.

The question of speed is not new. "[The] history of capitalism has been characterized by a speed-up in the pace of life, while so overcoming spatial barriers that the world sometimes seems to collapse inwards upon us" (Harvey 1989, 240). David Harvey (1989) employs the concept of "time-space compression" to describe how faster modes of transport have drastically reduced the size of the world by annihilating space through time. The experience of time-space compression is not merely a physical experience in the form, for instance, of jet lag; it also translates into accelerating turnover time in production and a parallel quickening in exchange and consumption, with the associated rise of temporariness in all spheres of life. At the speed of light, the electromagnetic waves of virtual mobility take the time-space compression to another level. As distances become transparent, a new form of space compression reduces the need for movement from one place to the other, and time compression makes more work available at any one point of time. This integration is different from the material integration of space attained through air transport, for example. The difference is not only that between the flow of physical bodies and bodies of code; it is also one of speed. Being a logical entity, code rides on electromagnetic waves close to the velocity of light, connecting spaces in "real time." High as it may seem, the speed at which data flows is impeded by many factors (such as data traffic and the low quality of cables), resulting in brief but annoying delays that seem foregrounded in functional schemes of integration; by the time delays are reduced for one type of dataflow (such as text), other forms of data (such as images or video) start competing for the increased bandwidth. We must remember that virtual migration does not merely suffer from the problem of delay; it constitutes the very possibility of delay. In the absence of virtual integration, complaints about slow connections in India would obviously not be raised.

The drive to increase the speed and bandwidth of online transmissions leads to a related dimension: time. This is another underexamined subject in labor and globalization studies: the programmed integration of workers' lived time. The distinction between spatial and temporal integration is mostly analytical, as they are two aspects of the same phenomenon. In step with the relativity of space-time dimensions, the sheer velocity of dataflow integrates spatially dispersed places while also bringing days and nights of different temporal horizons within a single framework. A global regime of "real time" increasingly presides over the previously secluded temporal pockets of life. Discourses of the "local" become possible only against an overarching global canopy.

Temporal Integration: Follow the Sun

The speed of electronic flows brings different time zones together and connects them in real time. Work is integrated across geographies, aided by the logic of programming schemes, including information protocols that facilitate electronic flows through adaptive routing. These protocols periodically reevaluate the fastest route between two points in the network, taking stock of the current traffic in the network, broken routes, and other problems. Guided by a routing algorithm, electronic packets hop from node to node, casting the net of real time over the globe.

New organizational structures emerge. The global twenty-four-hour office was always the hidden possibility and agenda of all programs of globalization. Now it is a reality with which practices of business and labor management must contend. From the perspective of corporate governance, the new arrangement allows work organization in two time zones to be sequentially patterned for competitive advantage—this is called the follow-the-sun approach. "Basically [when] it's night in the U.S., it's early morning here," a programmer in India explained. "At the end of their day [the Americans] just have to [compile] their problems and the changes they want us to do, and we can fix them in our normal working hours, fix them just in time, and it will be there next morning when they come to their office." The corporation's work in the United States does not stop when its employees are asleep at night, as Indian professionals keep advancing the project during their daytime. Many

processing and maintenance jobs fit this model. A few American insurance companies, using this pattern, have their back-end tasks completed in India. A programmer who worked at the time in the United States to coordinate work flow between the client and his software firm in India explained, "So, at 10:00 [p.m.] here, which will be around 8:30 in Bombay, in the morning, our daily batch cycles run . . . The claims that are entered in the day [in the United States] . . . will be processed in the nightly batch cycles in Bombay. We actually have about sixty jobs running one after the other, which update the table information."

In cases like this one the time-zone difference is an asset for the corporations involved: by the time offices close in the United States and night descends, software workers in India start working on the back-end tasks during their daytime. When the CICS (computer information control system) is not in use in the United States, Indian workers can provide solutions and complete them online. When the office opens in the morning in the United States, a lot of back-end work has already been completed, thus creating a virtual twenty-four-hour office for the American client. However, this new timing of organizational flows must follow the day-and-night pattern strictly if it is to function properly, as the team in India must finish all the tasks during their daytime. "Some of the files, which [the Bombay team and U.S. team] use, are common," the programmer further explained, "so unless and until these files are closed, we cannot start our cycles. The CICS has to be down [before the Bombay team can start working]. Around 10:00 p.m. [Eastern Standard Time] the CICS goes down, no information can be entered after that, so our batch cycle [in Bombay] can run. And if the batch cycle is not successfully finished within time, or if it gets delayed due to some reason, then there will be a problem, because these people [in the United States] won't be able to enter the information [in the morning]. So, it's very critical to resolve everything [before they open their office in the United States]."

The organization of time thus again receives a major overhaul. While the principle of the timetable has been essentially negative in its conventional form, forbidding idleness with moral injunctions against wasting time, the modern conception of time has been that of a positive economy, following "the principle of a theoretically ever-growing use of time; exhaustion rather than use; it is a question of extracting, from time, ever more available mo-

ments, as if time, in its very fragmentation, were inexhaustible" (Foucault 1979, 154). To the problem of exhaustion has been added a use of time which is not available, or at least not available to a territorially co-located community. While speculative options trading may be a good example of going beyond the presently available time, my contention is simpler. Earlier, when workers went to sleep in the United States, their productive time ceased to exist. Now, technical developments associated with virtual migration make available the productive time of workers who are not part of the territorial community, using the time that was not available to them earlier.

The temporal sequencing of work across continents is not restricted to cases of third-party outsourcing. Some companies have started to follow the same model of twenty-four-hour work, for their clients in the United States and also for themselves, by dividing their work groups and assignments between India and the United States. "So this is the model we follow . . . We have a dedicated satellite," the vice-president of a company proudly claimed, "It means you are reaching a level where you are able to provide twenty-four-hour support. Time-wise, when those twenty guys are working there, we are sleeping, when we are working they are sleeping, but then they are all connected with the mother computer, the main computer, they go to the server, pick it up from where it was left off." In certain software development projects, the twenty-four-hour work schedule may allow accelerated detection of problems or a distributed daily test-and-fix cycle.

This kind of temporal integration is also common to American companies that receive an enormous number of queries for technical support. They find this model especially productive in two senses. First, tech support provided by e-mail from programmers sitting in India is obviously cheaper than support from programmers in the United States, and second, the answers to those queries can be received within twenty-four hours, given the time-zone advantage. "They give out contracts for a lot of things. So tech support was one of those," explained one programmer, who moved from India to the United States under a body-shopping arrangement and was later absorbed by a major American company as a regular employee. "Working from India was more monetarily [cost-effective]; you can hire more than double the number of people for the same amount. It also worked for India, because the various data are reported during the day [and] we can get it to them in the nighttime,

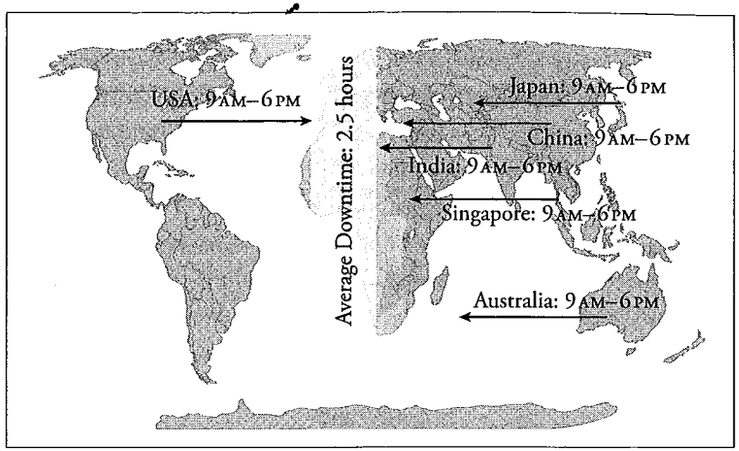

FIGURE 3. *The follow-the-sun approach to temporal integration.*

that is, nighttime here [in the United States], and people get their answers within twenty-four hours."

That technical expertise is physically located in India does not compromise the quality of work, as the software industry has developed fixed standards for governing and overseeing quality. In addition to imposing external quality standards (such as the ISO 9000 series) on general programming work—a practice that has recently been linked with deskilling and diminished job growth (Prasad 1998)—programmers in the tech support area must follow quality standards and adhere to the stipulated turnaround time for solving problems, as noted by another programmer: "We work under certain guidelines and certain principles and quality standards. All these things are documented in the contract or the agreement, so we are required to meet certain service levels. For example, the turn around time for this problem is two hours, but the other kind of problem may take four hours. It's all very well documented . . . Our effort is to meet whatever standards, whatever deliverable standards are there." A visual depiction of the twenty-four-hour office is shown in figure 3.

In this follow-the-sun approach, work hops from site to site daily. In software development, developers in California complete the day's work by passing their code to their colleagues in India who will shortly be pulling up their chairs at work. The Indian developers in their turn pass their work back to

California before diving into the rush-hour traffic in Delhi or Bangalore. This is surely not as easy as it sounds. Coordination is a complex business, requiring each party to wrap up its work in a way that it is readable and understandable by the other party, which can then quickly proceed to add value without losing the entire day deciphering what the other side did. If the virtual integration of work spans more than two sites, the problem of coordination multiplies in complexity. To overcome the problem, some software development tactics involve low coupling among relatively independent modules to be developed by different sites. Apart from regular coordination through liaisons who fly from various countries to the headquarters of the company to learn the system-level requirements and communicate them back to their colleagues at home, this form of transnational integration is heavily dependent on the governance schemes embedded in the architecture of defined interfaces and their concise semantics for the network element behaviors, along with low coupling, allowing teams that are geographically far apart to have relative independence.

However, the twenty-four-hour office model does not work in all cases. Some projects need constant interaction and lack precise specifications and standards to permit the temporal independence and neat delineation of twelve hours for each side.

Temporal Dissonance

In the event of problems and glitches, the time-zone difference hinders instant communication with the client, delaying the quick resolution of problems. Desirable as they may seem, strategies to integrate different time zones and twenty-four-hour office systems cannot universally meet all corporate needs, especially those dependent on face-to-face interactions. To overcome this problem, some companies open a branch outside the United States in the same time zone as an American site, for example in the Caribbean, to avoid higher wage costs without losing temporal proximity. As one informant who worked for a major financial enterprise described the situation: "[This company] had people working in India on projects. But . . . the time lag, time difference, [was a problem], they are sleeping and you are working, and you cannot really talk to them at the same time. [The work] was done through a [software] consulting company . . . [that] hired people in India; they worked there, and sent back some code. But it didn't work. So instead, what the consulting company came up with was that they moved them to Barbados." "So, they started working in the same time zone?" I asked. "Yes, the same time zone and same weather [as in India]," the informant responded, "And you don't have to pay them [as much as you do in the United States]; you can pay the same amount [as you do in India]."

Innovations in the governmental integration of transnational labor are constantly geared toward overcoming problems of space and time. Different models are devised for different kinds of work, costs being the constant factor in all decisions. The reframing of space and time may not always be easy or even successful, but the effort to govern and harness cheaper labor is always present. One CEO of a subsidiary of an American company in Noida, India, expressed his frustration over the time-zone difference between New Delhi and Seattle, which unlike cities on the East Coast of the United States offered no overlap with India in work hours: "In our case the problem is that it's exactly twelve and a half hours difference. So there are no common office hours. For example, if the office there started at our 4 p.m. or 5 p.m., I would be so much more happy. Because right now every meeting is an effort. On their part or on our part, both sides. Because we have to stay in office until 10 or 10:30 p.m. to have a reasonable meeting, and they need to start early, and vice versa. So that's an extra effort . . . When we worked with [our office in] Germany, it was so much easier. You know we would stay in our office till 4:00 p.m. and we would call them at 2:00 p.m., but just the U.S. West Coast, it's very hard." During daylight savings time in the United States the situation improves, but the West Coast and Delhi still do not have any temporal overlap, and this puts a heavy demand on programmers, project managers, and executives to work late at night or early in the morning so that they can find a temporal space for virtually getting together. As shown in figure 4, the Indian side must stay a couple of hours after 6 p.m. for a virtual conference with New York, and three hours more to connect with Seattle.

We can easily see that when it is time to close offices in Delhi at 6 p.m., New York is about to open its offices, and that by coming to work early in New York or leaving later in Delhi employees can create a temporal overlap to solve problems. On the West Coast, temporal distance puts more intense

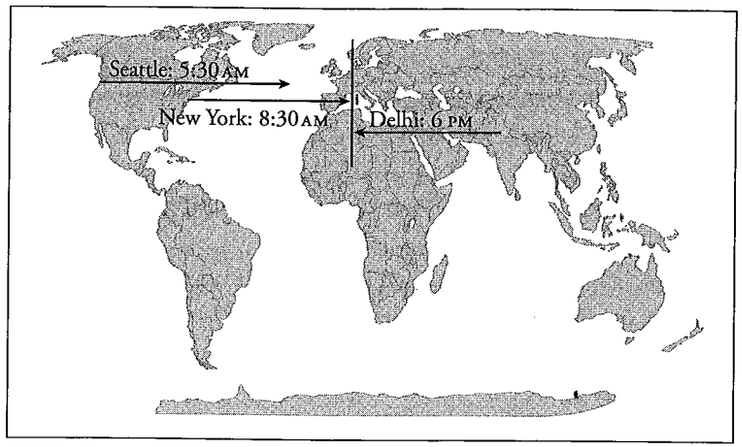

FIGURE 4. *Temporal distance between India and the United States.*

demands on employees. Just as physical migration emerged as a problem after the rise of strict national borders, the temporal location of a place became an obstacle to overcome after the real-time integration of continents. Each new solution to global integration comes with new needs and problems. Programming work, despite a high degree of specification and standardization, still requires, like other kinds of work, meetings and communication to resolve issues of software development. The frustration of the CEO quoted above was directed not at the whole labor regime but only at problems of communication raised in his kind of work (online R&D collaboration), in which the time difference between the two places was often an impediment. He clarified: "You get advantages also. You need to decide between the model that would take advantage of the situation and the model that's a liability. In our kind of model where we are working very closely, we need to talk to a lot of people. That becomes a liability . . . That's why I said if the difference was just eight hours, it would work beautifully. Because then you are getting the advantage and you are not suffering because of the time zone difference. Because there are times when I am up at 4 a.m. [or] at midnight to attend to the phone calls, which would not happen if the time zone difference were not there."

He later made it clearer that if his American office were on the East Coast, there would be enough overlap between office hours for the system to work perfectly. What was needed was merely two hours of overlap for daily communication, so that the people in New Delhi would not have to wake up at odd hours. Clearly, systemic flexibility is often achieved by demanding immense flexibility from workers themselves, who need to adjust their waking and sleeping hours according to the demands of virtual integration.[3] The transnationally connected organizations also come into conflict with national holidays. Even when there is an overlap of hours between two sites, the countries tend to have different off-days and religious or national holidays. While the United States celebrates its Independence Day on 4 July, India observes its Independence Day on 15 August and Republic Day on 26 January. India also has many more official holidays than the United States because of its diverse religions (Hinduism, Islam, Sikhism, Buddhism, etc.) and other cultural traditions. In fact, in the late 1990s there was a discussion in the Indian Parliament about the country's large number of holidays—three times as many as the United States—and the consequent loss of productivity. The national order of things comes into conflict with the emerging real-time regime of transnational integration. The temporal dissonance is a contentious issue when setting deadlines or phone or video meetings.

In the current round of integration through information technologies, the conquest of space is simultaneously less sweeping and more drastic than earlier conquests by transport technologies. It is less sweeping because the body stays put, while symbols and signals zip across continents. On the other hand, the sheer speed of instantaneous electronic transmissions conquers space more drastically. For the first time, the obstinate resistance of space is overcome in a way that makes different times zones collide daily, as groups based in India and the United States enter into each other's time zone. As technologies of transnational governance are able to break into the local social times of people directly, they also occasion a break with their local contexts, as I discuss below.

Globally Yours: Reconfiguring the Lifeworld

At about eleven o'clock in the morning I reached the office of a major software firm in Gurgaon, a city that together with New Delhi and Noida forms the northern hub of software firms in India. It was April and the dreaded, rather

early, north Indian summer had just set in. I had driven about twenty miles from New Delhi to interview the firm's senior manager. The air-conditioned interiors of the office resembled a corporate office in the United States—plush, orderly, clean, and carpeted. I always experience a culture shock at the contrast between the quiet interiors of major software firms and the outer din and disarray of Indian cities. Even from the glass opening in the door that divided the lobby from the large open office space, I could see nicely partitioned cubicles with computer monitors and phone sets. While being led to the senior manager's office, I was struck by a rather odd and curious sight: the office and all its cubicles were totally empty. During the meeting, which was cordial and informative, I could not help asking the manager about the absence of employees in this workplace. "Oh, they will start coming in at six in the evening," he responded nonchalantly. He explained that the empty part of the workplace that I saw was a call center, and workers were expected in the evening to start working during American office hours. Obviously the sparkling office space inside and the chronically unfinished construction work outside reflected not merely a contrast of two places, India and the United States; the two spaces were in different time zones. Transnational governance was fast overcoming the earlier CEO's complaint about not having enough overlap with office hours in the United States, especially the West Coast. Many firms had already adapted—by working at night—to serve their clientele in the United States during their daytime, and there was no shortage of people looking for jobs that required them to adjust and synchronize their lives with regular consumer or work hours in the United States.

Technologies of virtual mobility are increasingly made to penetrate into local times, and thereby reconfigure local contexts and the social times of people's lives. Globalization emerges as a series of different effects in different places (Guillén 2001). As physical space becomes drastically compressed by these technologies, this freshly achieved flexibility of labor supply demands certain life adjustments. People who performed nightly labor lived in two worlds. One happened during their daytime (during the major part of which they slept), marked by local languages, friendships, and relations; the other, at night, was marked by the English language as well as interactions with people living in the United States. The holidays enjoyed by the worker were those celebrated in the United States. This integration into a transnational system of labor effects a break with local mechanisms of social integration and social solidarity, by introducing what Eviatar Zerubavel (1981) calls *temporal asymmetry* as opposed to *temporal symmetry*—a temporal coordination that makes communal life possible by organizing activities in the same time frame. The importance of temporal symmetry is clear: "a most common method of punishing monks was to segregate their activities temporally from those of the rest of their community, or by having them eat their meals three hours after them, rather than together with them" (Zerubavel 1981, 65). Night work in software companies is putting workers out of phase with their own society. There is a reason why night work has another name: the graveyard shift. In this graveyard of social life (or social death?), many workers seem dissatisfied and treat their jobs as a temporary phenomenon in their lives. By placing people in another time zone, programming schemes of integration enable a systematic unhinging of people's "present" time from their concrete presence here and now, for the sake of their transpresence in another space (Virilio 1993). Spatio-temporal integration is at once spatio-temporal alienation. There is a high attrition rate among these workers. But demand for their services is growing at a high annual rate.

In the last few years, consumers in the United States have gradually become aware that when they call their bank's customer service department, the phone might ring in another continent. They may or may not know that approval for a home mortgage was conveyed to them by a person sitting in Delhi, that their medical claims were assessed by someone in Gurgaon, or that calls chasing credit-card debtors in the United States are made from Bangalore. Globalization has finally dawned on popular consciousness in a personal way. Americans are increasingly aware that that customer service agents—with friendly voices, American pseudonyms, and some training in acquiring an American accent—are telecommuting at night from India to their living rooms. Sounding "American" is important. Inauthenticity—or cultural simulation—is the very basis of authentic performance. Call-center workers in Gurgaon may tell American customers that they are calling from an American city to put these customers at ease, since their private finances could be a matter of conversation. On their cubicles are posted large nameplates with such common American names as "Victor Smith." Some call centers do not allow the workers to use their real names at all in this

FIGURE 5. *India vs. America. Cartoon by Khalil Bendib for* CorpWatch *(in Kumar and Verghese 2004). Reprinted with permission.*

Americanized space, lest they forget their American name while talking with customers.

Economics and culture merge in capital's effort to create a world after its own image. Workers watch Hollywood films to understand American diction and undergo speech therapy to sound "American" while also receiving at times, for the sake of ambience, coke and pizza on weekends. Software firms in India are now also getting part of their work force trained in American history and geography. Language plays an important part in the success of Indian software firms, as the considerable prevalence of English in India gives them an advantage over their competitors from non-English-speaking countries like China. English, a legacy of colonialism that is still used only by a small share of the population, has increased markedly since the British left India. Interestingly, colonialism was the first serious wave of transnational governance, marked by trade expansion, the administrative integration of colonies, and the spread of a few European languages, English clearly preeminent among them. The programming of postcolonial globalization also needs certain unifying mechanisms, including linguistic homogeny, to tame existing incongruities and chaos. English again comes in handy. A number of institutes opened in India to train the Indian tongue in the ways of American speech, or at least neutralize the thickness of difference. It is not the success

of the night labor regime that merits attention; what is of importance is the very endeavor to erase from view the disjuncture of different worlds, different time zones, different subjectivities, languages, and accents.

With a feminist twist, the *Wall Street Journal* claims that the "revolution" brought about by call centers is not merely economic but also cultural, challenging age-old patriarchal values that prevented women from working at night: "As outsourced jobs pour into India, they are bringing much more than money to this nation of one billion. They are also creating a young, affluent class absorbing Western attitudes at the office, far from parental supervision. The independence of these twenty-somethings is helping to unravel time-honored social mores in India where young people are expected to marry someone their parents choose and live with an extended family. The idea of women working at night was unthinkable until recently" (Slater 2004). The journalistic claims of affluence as well as women's liberation must obviously be taken with a pinch of salt. I would not go so far as to describe call-center workers as affluent. Unless these young workers lived with their parents, they could only hope to rent a decent apartment at the starting salary of Rs. 8,000–10,000 in places like Delhi and Bombay. Salary raises are tightly linked with the number of telephone calls that one takes so that the targets set for one's team can be met, and the number can reach the incredibly high number of 250–300 calls a night. Still, one cannot call such practices oppressive in the regular sense of the term. The jobs are sources of income for this group of relatively young workers. Call-center work may also reflect a fascination with things American and may not be immediately perceived as oppressive or imperialistic by the workers. True, there was no desire on the part of the workers interviewed to continue in their jobs forever. Most of them left within a year. Even as a researcher, I could not spend the whole night observing work. Permanent night shifts, exceedingly high call targets, and such health problems as insomnia, weight loss, stress, and declining eyesight were some of the reasons why employees did not stay on their jobs for long.

Before one starts complaining about exploitive work conditions in India, and the absence of a level playing field for American workers, one must realize that night work is not a new or a mere transnational phenomenon. The colonization of night, to borrow a phrase from Murray Melbin (1987), has been a growing phenomenon within the United States for a long time. Re-

cently Harriet Presser (2003), in a disturbing look at the pervasiveness of nonstandard work schedules in the United States, pointed out that two out of five employed Americans worked mostly at nonstandard times—in the evening, at night, on a rotating shift, or during the weekend. In this overlooked story about a silent, swift, and invisible social fragmentation, spouses are not together at home in the evening or at night, and parents are often not home with their children. People's integration into the twenty-four-hour economic system is also a temporal unhinging of family life. While the rhetoric of a 24/7 economy conjures up images of a fast-paced, seamlessly integrated engine that serves its consumers around the clock, the social cost of economic growth does not enter into the discussion. In virtual migration, the ever-growing subordination of the social to the economic acquires global dimensions, as this surrender of social life to the economic system can literally take place across the globe.

The intellectual left is faced with a dilemma. On the one hand, the systemic colonization of night creates a new form of inequality—work-time inequality—extending its imperial march into life's very foundations and altering circadian rhythms, which are connected with body temperature, hormone levels, and that expendable comfort—sleep. Night work may be associated with higher rates of cardiovascular disease, breast cancer, gastrointestinal disorders, miscarriage, preterm birth, and low birth weight (Boggild and Knutsson 1999; Schernhammer, Laden, Speizer, Willett, Hunter, Kawachi, and Coldtiz 2001; U.S. Congress 1991). Yet on the other hand it is difficult to assume an easy anti-capitalistic stance, because the jobs and capital generated by the software firms do help the local economy. To opt out of transnational capitalism might worsen job prospects; it might reduce the chance to have any say in projects of globalization and the direction they might take. To stand out of contemporary imaginations of world society or global membership, as discussed by James Ferguson (2002) in the context of sub-Saharan Africa, is not a real option. Indeed, it is wrong to frame the question as one of choice. Disengagement is not an option either on the part of software firms or their employees in India. The world economic system—expansionist and powerful by nature—continues to reach all pockets of the world by either favorable or unfavorable engagement. Disengagement is merely a form of unfavorable engagement. It is quite like an attempt by a

student to resist the grading system by not submitting the final assignment, i.e., by not participating. But the system still engages her, if unfavorably, by giving her a failing grade. If she continues with her attempt at disengagement, she may even end up homeless with no credentials or employability, as the system reduces other support alternatives. Disengagement is impossible in another sense as well. Contrary to my slightly flawed analogy of the student, there is no single subject in the context of globalization who can exercise the choice to disengage in the first place. Saturated as our political talk may seem with language that equates nations with individuals (for example when we refer to "rogue nations"), India is not an individual subject who can disengage from globalization. The global regime has produced a new Indian transnational class whose interests are in harmony with programs of greater integration. This class in turn mobilizes through employment all local others in programs of globalization. If there are any rules still lingering from earlier times, any vestiges of the national order in conflict with the new global regime of labor, they are removed in favor of a more productive, competitive, and integrated system.

Some old rules in India are being erased to make way for the new ones. In April 2000 the governor of Tamil Nadu exempted the software industries in the state from the chapter II provisions of the Tamil Nadu Shops and Establishments Act of 1947. In effect, the rule on opening and closing hours of the shop would no longer apply to software outfits. Local restrictions designed to protect the social world from the encroachments of night work must yield to the new global labor regime. National legislation must be brought in line with global programming schemes. The Factories Act of 1948 is in for an overhaul. The act stated that no woman shall be required to or allowed to work in any factory except between the hours of 6 a.m. and 7 p.m., with the proviso that the hours may be varied as long as no woman works between 10 p.m. and 5 a.m. The Factories (Amendment) Bill 2003 seeks to remove this restriction imposed by section 66 of the Factories Act of 1948 on women's night employment. To allay the fears of the Central Trade Union Organisations that the "work environment in the country was not favourable for night work of women" and that there was a danger of harassment and exploitation, the bill allows "employment of women workers between 7 p.m. and 6 a.m. only if the occupiers of the factories make adequate safeguards as regards occu-

pational safety and health, equal opportunity for women workers, adequate protection of their dignity, honour and safety and their transportation from the factory premises to the nearest point of residence" (Factories (Amendment) Bill 2003). The bill also mentions that "many Women Organisations have filed Writ Petitions in the various High Courts seeking directions for amending the Act to provide for night work by women on the ground that the existing provisions of the Act are discriminatory." Women in India are finally gaining the "freedom" to work at night.

The adjustment to a global labor order was begun earlier by the labor ministry itself. In a statement to the International Labour Organisation (ILO), it provided the rationale behind lifting the ban on the night shift: "Ministry of commerce has sent a proposal to the ministry of Labour that the female workers should be allowed to work in the third shift in the export processing zones to stimulate India's exports. These zones are set up as enclaves, separated from the domestic tariff area by fiscal barriers. These are intended to provide an internationally competitive duty free environment for export production at low cost. By allowing women workers to work in the third shift, it will not only help to utilise the installed capacity but will also be cost effective in the competitive international market. This would also lead to increased employment opportunities for women. Ministry of commerce is also of the view that productivity and turnover of women is much higher than that of men in the field of electronics" (Sindhu 2003). In the landmark case of *R. Vasantha v. Union of India*, the High Court of Madras in 2001 ruled in favor of a woman petitioner who contended that the Factories Act, in violation of the Constitution, discriminated against women by restricting their lawful employment. The judge agreed that denying the night shift to women amounted to discrimination on the basis of sex and gender, depriving them of chances of fair employment and equal opportunity. Calling the legislation an instance of "romantic paternalism," the judge found in it traces of the stereotype that women's role should be confined primarily to the family. The law was a way to restrict women to household activities and thus for men to retain economic superiority. Similarly, the Report of the Second National Commission on Labour maintained that "on the question of night work for women there need not be any restriction on this if the number of women workers in a shift in an establishment is not less than five, and if the management is able to provide satisfactory arrangements for their transport, safety and rest after or before shift hours" (Ministry of Labour 2002).

The above is a good description of a familiar feminist dilemma. In the first half of the twentieth century, the issue of special protective legislation for women created a split within the global women's movement, as discussed by Elisabeth Prügl (1999) in the context of homework. While equal-rights feminists condemned the legislation for its discriminatory effects, union women defended it, arguing that women's working conditions were at times objectively different from those of men. In the late 1920s this led to a split of the International Alliance of Women (IAW). Although both defenders and enemies of protective legislation preferred equal employment status for women, those who favored protective legislation regarded women as actual and potential mothers, recognizing motherhood as a disadvantage in the labor market. Equal rights feminists, on the other hand, defined women as free individuals in the liberal sense and thus contested the association of womanhood with weakness and dependence.

A proper critique should perhaps not be framed in terms of women's freedom to choose their hours of work, for surely everyone must have equal opportunities. The issue might be framed instead along the lines of a struggle for an eight-hour workday—that is, not necessarily in terms of the equal distribution of misery (night work for all) but of its reduction for both genders. The critique must uncover the effects of various programming schemes that bring about greater system integration, and the ways in which concrete social and personal lives are subordinated to system imperatives. Legislative changes are important to understand, but we must also focus on how and what governance mechanisms are coded into the programming schemes of global integration. In the next chapter, I attempt to narrow my discussion by discussing only software-based programming schemes—code—along the twin dimensions of organization and integration.