The Davids go Global - The Politics of High-Technology Industrial Development in Peripheral States
Ireland, Israel, and Taiwan

Preliminary First Chapter:

The Military as a Public Space – The Role of the IDF in the Israeli Software Innovation System

Dan Breznitz
Industrial Performance Center
and
Department of Political Science
Massachusetts Institute of Technology
Room E40-370
One Amherst Street, Cambridge, MA, 02139, USA
Telephone: 617-4522505
Fax: 617- 2537570
tbvb@mit.edu

<u>Extremely preliminary!!!
Appendixes and reference incomplete, for discussion only
not to be cited or circulated without permission!</u>

The author would like to thank Suzanne Berger, without her this research would have never occurred, to Richard Samuels for supplying the initial forum in which my thoughts evolved, to Mike Piore for the moral support in times of great need, to Morris Teubal for insightful remarks on the second part of this chapter and to the many interviewees who gave me hours of their time. A special thanks is owed to Y and G, may you forever prosper. The MacArthur Foundation through a grant given by the Center for International Studies in MIT enabled parts of the research described in the second part of this paper.
All mistakes are solely the author's.

Introduction

Whether "new economy", or old economy, contemporary political economy has a dismal view of the prospects for leapfrogging by backward societies. While each of the standard theories has a particular approach, they all unite in a pessimism about the possibilities of using government policies to leapfrog, i.e. to transform a relatively backward society to a front-rank competitor in technological innovation in a time-span of one generation. Thus, the question of what, if any, is the role of the state in developing a new economy in societies that have struggled in the old, is left unanswered. The neo-classical model, focusing on the market, does not offer any prescriptions beyond stable macroeconomic policy, open liberal markets, patience and luck. The revisionist or statist model, focusing on the role of the state in staging industrial and technological catch-up, continues to focus on borrowing and also assumes that such states can continue to rely on one-way trading relations with the industrial core. Moreover, it fails to explain how in recent years "weaker" states seem to excel while "stronger" states suffer. The market-friendly model sees a larger role for government but one confined to social and physical infrastructure, openness to trade, and macroeconomic stability. [1]

Given this analytic gloom, it is a surprise to find that many of the new Information Technologies (IT) are being developed by firms in peripheral states -- societies that were not known for their high-tech industries in the past. Riding what seems to be one of the greatest waves of industrial innovation ever, the Taiwanese, Israeli and Irish indigenous IT industries have become hotbeds of New Technology Based Firms (NTBF). Some of these firms have become global players over the last ten years. Remarkably, however, although all three countries are very successful by any measure, operate in the same markets, and developed during the same period, their IT corporations seem to embody different business strategy models when they try to penetrate global markets, and in the concentration of innovational activities on a particular position in the production chain of the same industry. These findings raise two sets of questions. First, since almost every state around the world has tried to stimulate the growth of indigenous IT industry, why and how did some states succeed, and others did not? Secondly, do these success stories point toward one or multiple models of economic development?

Answering these two questions would help us to elucidate the role of the state in industrial development, specifically in the initiation and sustaining of innovative industrial leapfrogging. This question is also intimately linked to the two central questions of the current globalization debate: does globalization indeed limit the ability of states and societies to shape their own economic future? [2] And does the globalization of markets force convergence to one political-economic model? [3] In this regard, IT is among the most globalized of industries.

This thesis aims to be a first step in covering our gap of understanding these issues by analyzing the development of the indigenous IT industry in three peripheral states, Ireland, Israel, and Taiwan. [4]

I propose that the state role is one of building institutions that induce and sustain the development of the IT industry. [5] I argue that differences we find between the three states, are shaped by two key variables: first by state policies to induce industrial production and to develop specific sets of skills, and second, by the particular pattern of state-industry relations that evolved as the industry grew and started to interact with the state and with other parts of the national system of innovation.

In the end of the 1950s Taiwan, Israel, and Ireland were poor, peripheral and technologically backward since independence, and had mixed histories of success in more traditional industries. However, by the beginning of the 1990s, the indigenous IT firms of all three leaped to the forefront of the global markets and become the drivers of their astonishing economic growth. [6] Firms from all three states overtook firms from the more advanced industrial states, and started to create new technologies and lead their markets in innovative product development. The puzzle is that these development trajectories do not fit our common understanding of industrial development as an organic growth process, where national economies pass through largely defined stages. Are these the exceptions that prove the norm? Or do these small states offer us insights for development today?

In the 1960s state-led initiatives to develop indigenous high-tech industries appeared in all three societies. At that point all three countries were very similar in most of their important economic variables, including size and population. All three had low skills intensity in their labor force (defined as the number of scientists and engineers in the population), a relatively low percentage of high school graduates, poor communication and physical infrastructure, and an overall reliance on agriculture. All three states followed similar infrastructure policies. Each improved its education systems, and all had an exceptionally high ratio of skill formation between 1960 and the 1980s. All three improved physical infrastructure, and after failed attempts at direct bureaucratic control created public telecommunication companies that vastly improved the infrastructure and line subscription penetration rates. Furthermore, all three have high rates of wireless and Internet penetration. Finally, unlike Japan and South Korea, they rely on small and medium size enterprise-led growth and not on big corporations.

Despite the similarities of their macro level and infrastructure policies, however, their micro, i.e. industry and firm level policies, were distinctly different. Interestingly enough, while from a purely economic view the problems faced by these countries were quite similar, very different concepts were prominent in the socio-economic policy dialogs. These different concepts led to very different industrial policy approaches. For most of the period since the late 1960s, Ireland conceptualized its problem as unemployment leading to massive immigration and devised a strategy that relied mainly on inward investment policies, with heavy investment in the supply side. Israel conceptualized its problem as one of creating a competitive advantage under conditions of scarce resources based on the perception that the needed state role is fixing market failure of underinvestment in R&D.[7] Accordingly, Israel devised policy that relied on inducing industrial R&D activities through government grants, with nominal neutrality in grant allocation, and project ideas originating solely from private industry. In Taiwan, the ruling party, the KMT, mistrusted big private industry and feared the rise of competing powers. Hence, it prevented the creation of big corporations, controlled the inflows and outflows of investment, and relied on state research agencies like ITRI and the III, to lead the R&D efforts and diffuse the results throughout the private industry.[8]

I conceive macro level and infrastructure policies as a necessary but not sufficient condition for innovative leapfrogging. I argue that it is the micro level institution building policies that affect technological innovation-led growth. Hence, the mix of similar efforts in regard to factor and infrastructure creation, and vastly different approaches toward formal and informal institution building, toward market creation, and toward the setting of the national innovation systems, offer an almost natural experiment with which to test the hypothesis about micro level institution building policies. In this way, the analysis attempts to recast the economic development debate.

The hypothesis that is developed and tested is that the state's main role is one of institution building. More specifically I present and test three sub-hypotheses. First, *interventions by the state on the micro level affect outcomes by creating different institutional settings*. Second, *different micro level policies lead to different institutional settings, hence to different outcomes*, and third, *different sets of micro policies can each lead to successful outcomes*. I contend that various state policies built specific sets of informal and formal institutions in each society.[9] These institutions spurred the creation of the first IT firms, which developed in interaction and discussion with the state and with the other parts of the national innovation system. However, past institutional setting shaped both the initial development of these corporations and the nature of the corporations' relationships and dialogs with their environment. Thus, different politics molded different processes of development and subsequent institution building in each country. Therefore, each IT industry was fashioned around a different political economy model, embodied by the different business strategies used by private firms and the different point of innovation on the same chain of production that each industry concentrate in. [10] For example, while both Israel and Taiwan have strong semi-conductors and electronics industries, and a very similar US patent per capita profile, most of the Taiwanese innovation deals with production and manufacturing, while most of the Israeli innovations deals with IC design intellectual propriety. [11]

This paper is divided as follow in the first part I define my concepts and lay the theoretical background. These enable me to identify the institutional variables I should use. In the second part I use these variables and theoretical insights on a small case study – the role of the IDF in the development and continuing sustainability of the Israeli software industry.

Definitions and theoretical setting

In this part of the paper I first define my theoretical concepts. After which I offer a short literature review and theoretical background that I use to flash out and define the institutional variables from the historical institutionalism, innovation, and economic growth literatures.

The focus of this thesis is *innovative industrial leapfrogging*. In order to define this concept I delineate it in two levels – first the meaning of "innovative industrial activity," secondly leapfrogging as compared to "normal" industrial development. I than theorize on what necessary capabilities a state should posses before it could start to innovate in the highest level.

The meaning of *innovative industrial leapfrogging*, as it is used in this thesis, pertain to the ability of a national economy not only to use technologies that were created somewhere else in order to produce better goods in lower or competing costs, but to the ability to create new technologies and products on the basis of these technologies. For example, while many poorer countries import textile technology from the industrial core and try to compete on cost basis. The first Israeli technology firms starting to export to the US in the early 1970s exported their own technologies and products.[12]

In order to simplify this concept we might think about it as a five stages development ladder.

1. An economy in the first stage has the ability to offer MNCs cheap low-skilled workers, which will work under supervision in *a foreign owned and managed* factory.
2. An economy in the second stage developed the ability to import final technology and use it in locally owned or managed factories to produce goods designed elsewhere. With its main competitive advantage still around lower costs for the factors of production, mainly labor.
3. An economy in the third stage has firms that have the ability not only to use and employ foreign technologies but also to design and produce, or upgrade, final products. However, most of the innovations in the economy consist of incremental improvements of existing products.
4. An economy in the fourth stage not only employ technologies developed in foreign countries, but also have the ability to use general propose technologies (GPT) to create completely new products and applications, and conduct R&D that would result in original improvement to the technology itself.[13]
5. An economy in this stage develops and researches various GPTs from scratch in addition to do everything that is done in stage four economies. For reasons that have to do with size, resources an the ability to appropriate the results of research, usually only the larger more developed economies, can successfully operate in this level in a constant and significant way.[14]

It is important to note that the stages are not exclusive, thus an economy might have firms operating in different stages of the ladder, we can however, defined the stage an economy is in, by the highest stage of development firms operate in.

This five stages model follows more or less closely the widely used definition of the three stages of Science and R&D, basic research, infrastructure research, and applicative research. Therefore, in order to spur an sustain an innovative industrial leapfrog, peripheral economies need to support a critical number of firms that operate in stage four, and enable throughout the economy affective research of infrastructural and applicative technologies, with strong abilities to conduct basic research, which should help it to stay in tune with the worldwide changes in technology and science.

In sum, for our proposes successful innovative industrial leapfrogging should cement in a critical number of firms in the national economy that operate in level four, with their main product being new technologies, and/or new application of General Propose Technologies.

We are now left with the definition of leapfrogging being inherently a relative term, i.e. one can only define leapfrogging if one defines the norm. According to most economic theorist, neo-liberal, statist, or market friendly. Economies usually evolve through largely defined stages, which conform to a great degree to our own ladder. In the first stage developed economies use their relative abundant supply of cheap labor to start producing cheap commodities products, they than slowly and usually not so surely through the time span of few generation move through the ladder of economic activities. With a move up the ladder achieved only after they gained full control in lower stage industries.

This is the main paradigmatic assumption in both Adam Smith’s and Marx’s writing that they could focus only at England as the English industrial revolution can predict what would happen in the future to any industrializing state. That is also the story told by modern economic historian like Alexander Gerschenenkron and his followers, arguing that there is a relative advantage to backwardness they base it on the assumption that the backward states could, with the use of growing amount of capital, *utilize technologies developed elsewhere with a greater efficiency*.[15] This is also the story told in different variances about the miracles of South Korea and Japan from both side of the economic debate.[16] While they vehemently disagree about the cause of economic growth, one side championing the strategic management of the state the other granting it to the all powerful invisible hand of the market, both side describe how Japan and later Korea moved slowly but surely through series of more sophisticated industries, starting with textile, and moving on to shipbuilding, construction, automobile manufacturing, and consumer electronics, only in the end cementing with the ability of indigenous technological R&D.[17] Moreover, even when Japan and Korea did manage to move to the technological edge, for example in computing, they did that by first copying and reverse engineering technologies that were developed mostly in the US and only slowly through the use of incremental changes managed to pose a technological threat.[18]

Thus explicitly or implicitly, one of the major assumptions of economic development theory is that in order to succeed in the more technologically sophisticated industries states should first master the old industries. However, in the three cases which I follow in this research there seems to be a move in the time span of one or two generation, from agriculture economies with no history of industrialization to acclaimed success in the new IT industries, with only a rather mixed history of success in the old established industries.

In sum, I define innovative industrial leapfrogging, by the ability of an economy to move in the time span of one to two generations from a state of agriculture economies with very little industrialization to economies that rely mainly on new technology producing firms, without ever being able to successfully master the more traditional industries.

Thus, our first priority is try to understand what are the preconditions suggested in the literature for industrial innovation and identify the institutional variables which we should use when we analyze the three cases.

Literature review and institutional variables definition

Let us check what conditions and institutions were identified by studies from all three approaches as important for industrial success, and try to build some model with which we can try analyze empirical studies. There are three main bodies of literature that try to understand the preconditions for the development of industrial innovation and new technologies in any national settings.

The first, stemming from mainstream economics tries to adhere mostly to formal equations and quantative only research, is the body of literature known as growth economics and include both neo-liberal and endogenous growth theories.

The second and the third bodies of literature, stems from the same research agenda -- Institutionalism. Unlike the growth economics field which exclude any writings by none economists, Institutionalism is an interdisciplinary research agenda involving writers from the economics, political science, sociology anthropology and social psychology, and geography and regional studies.

The main claim of Institutionalism theorists is that the main social vehicles influencing decision-making, worldviews, policies, and technological and industrial capabilities are institutions. It claims that different institutions built in different times in different countries, influence the way people act and think, and the way they make decisions. Thus, political and economic institutes in various societies shape decisions and behaviors in their societies.[19] These institutions consist from formal ones like the rules regarding the transfer of land and property, or informal ones as norms of behavior, like who talks with whom and when. Accordingly, following North we will draw a distinction between organization and institutions. Organizations are groups of individuals bound by some common purpose to achieve objectives.

As these institutions are relatively stable through time, we can understand the way different countries behave by focusing our research on the development of their internal institutions. It is important to note that throughout this thesis I, unlike some institutional economists, argue that these institutions, these ideas about the relationship of the state, society, and the markets, are not only mediators of external influences and shocks, but are also the source of innovation, dynamism and change inside the system.[20]

There are two main level of analysis by which writers use institutionalism to try to understand industrial innovation and growth. One focused on the national settings and claim that the best unit of analysis with which to understand industrial innovation and growth is the national level.

The second approach has two strands that for various reasons that I will elaborate on later can be viewed as one, and claim that while the national level might explain quite a lot we should look either to the regional level or the industrial sector level if we want to understand growth.

There are indeed some interesting insights that we should talk away from the more formal economic growth theories, especially from the endogenous growth theories. The main being:

1. New innovations and technologies are a necessity for long-term economic growth.[21]
2. Starting from Kenneth Arrow's seminal paper (1961), through the research of Zvi Griliches and his students, to our days research n the introduction of major new technologies or GPT, growth economics has proved beyond doubt that there are a multitude of inherent market failures regarding both R&D activities and the diffusion of the results of these activities throughout the economy. These market failures stems from the three main attributes of new technological information: its indivisibility, inappropraiability, and the high uncertainties associated with

R&D. These reduce the chance that (a) the amount of R&D taken in a free market economy will be on par with the celebrated Pareto optimum, i.e. in a free market economy there will always be a tendency to underinvest. (b) As the research by Griliches and his student shows the diffusion of a new technology is uneven, and places where the other factors of production (e.g. labor and land) have lower costs, and or employ the old technologies more efficiently would tend to deploy the new technologies last. In sum, without specific sets of institutions and/or some government intervention we should expect to find that in a free market economies there would be (a) underinvestment in R&D, and (b) in places with lower wages and land prices (for our proposes read developing economies) the rate of deployment of new technologies, and diffusion of new information will be slower.[22]

3. Because of the nature of technology and innovation it evolve in jumps and discontinuities. Thus, we should expect to see different cycles of innovation and economic growth that start around new major technologies and their diffusion throughout their economies, which disrupt the whole economic system and create what Schumpeter called "Gales of Creative Destruction" followed by major slumps.[23]
4. Lately the new endogenous growth theorists broke with the neo-classical growth theories, and argue that technological innovation should not be treated as an exogenous residual to the system, but argue that technological progress is created through the action of agencies e.g. individuals and firms. Hence the ways in which different states are organized, and the changes in their human and physical capital, influence the rate of growth of different nations.[24]

Thus, even the more formal researcher urge us toward an institutional focused research agenda, and seem to hint that state actions in science and technology, and industrial, policies, are highly important and can have decidedly positive impact, especially in the less developed economies. However, growth economics and endogenous growth theories leave us with very little in the description of what these institutions should look like and how can states could achieve higher rates of technological innovation.

It is to institutional theory than that we have to turn to look for our variables. In the national level strand of institutional analysis there are few families of theories we should look at. One of which are these that argue for the existence and analyzing of what they term national systems of innovation.[25] The argument of these theories is twofold (1) that particular social systems consisting of various institutions, relationships of labor and management, government and business, the education systems, labor skills, and social norms and ideologies, either inhibit or induce innovation and a quicker defusing of new technologies throughout the economy. Consequently, enhancing or reducing the ability of the national industry to produce new products based on new technology in good enough quality to compete in world markets. (2) The best way to analyze these systems is on the national level, i.e. the distinct relations between actors in the business, finance, state, and educational arena in the national level, best explain the level of innovation throughout the industry and throughout that nation.[26]

Another line of argument in the national system of innovation agenda is based on the theory of research or technoeconomic paradigms (Giovanni, Dosi, 1982), these paradigms are diffused in the national level and effect R&D in a similar way that Kuhn (1962) argued that scientific paradigm effect the “work’ of science. Once you have a ruling paradigm of (1) what are good R&D activities and line of research, hence also of what would be neglected, coupled with (2) how R&D should be done, most R&D activities and efforts in the national system tend to fellow that paradigmatic approach. A third line of argument on the national level comes from Alexander Gerschenkron’s insights on the importance for any industrial leapfrogging, of ideology of progress through technological innovation.[27] Hence, we should be able to identify in any industrial, or industrial would be, nation not only an institutional system of innovation, but also a distinct research paradigm, and technological ideology, and using a comparative analysis understand how they influenced each nation's industrial growth.[28]

It is also important to note in this stage Piore and Sabel's important distinction between two paradigms of industrial production, innovation and R&D, and markets' structures that influenced the debate about industrial development. Piore and Sable distinguished between Fordist or Mass Production systems, and Flexible Production systems (1984). With the latter defined by a large amount of relatively smaller firms that work in close relationship and network with each other and with their suppliers, where information flows relatively freely from one firm to another, and where the system as a whole is highly innovative.

There are three main paradigmatic or level of analysis arguments, and one methodological argument against the national level of analysis view.[29] The methodological argument follows those lines: even if the national system line is correct, can we really define national system of innovation? Moreover, as technology and innovation move so fast can we really claim that our definition has any value?[30]

The first paradigmatic argument is more of a correction then a disagreement, and argues that there are national systems of innovation, but they do not and moreover cannot be viewed as promising overall industrial success, as each industrial/technological sector have a different optimal system of innovation or industrial governance.[31]

The second line of arguments more or less agrees with the basic lines of the national systems argument, but argues that those systems are best viewed as regional and not as national. The different between the regional systems inside countries might be as big as the differences between countries.[32]

The third line of argument contends that we should not view the national system as having overwhelming effect on the whole national industry. The differences between specific sectoral governance system, stemming as they are from the historical development of that specific sector in each country, and the worldwide technological and innovational level of that sector, can explain and predict industrial success, much more accurately then the national level alone. Moreover, as usually there is some correlation between industrial regions and industrial sectors, the latter views (regional and sectoral) can be used together in constructing a more accurate and comprehensive explanation of industrial successes or failures.[33] Let us check what conditions and institutions were identified by studies from all approaches as important for industrial success, and try to build some model with which we can try analyze empirical studies.

One important understanding that writers from all paradigms agree to is the importance of infrastructure; very little industrial success can be conducted without a proper infrastructure be it, physical, human, and legal. Thus we should give close attention to the development and differences between the infrastructures in all our cases, and try to understand how can a proper infrastructure be developed, maintained, and fine-tuned.

Secondly, starting with Schumpeter all the way through the writing of Gerschenkron to our day varieties of capitalism approach, it is argued that the organization of finance and the availability of different sorts of capital play a major role. Accordingly, we should look to the ways capital markets and financial institutions were arranged, and especially to changes they pass through in our different cases.

Thirdly, all writers attribute great importance to the flows of knowledge in the industry, be it technological knowledge, the know-how of R&D procedures, and good business making knowledge. These flows of knowledge should also occur in a way that enhance social or collective learning on which sustained constant innovation depends. Collective learning, unlike individual learning, is a process through which learning of individuals is influenced by joint learning of others, and the output of it are integrated new collective knowledge. Thus, collective learning still depends on the level, creativity, and skill of individual participants, but only through this collective endeavor these individuals gain knowledge not available outside the process. When that learning process is optimal, therefore cumulative and innovative, new opportunities for new technologies and innovations become apparent to insiders, while not so readily available to outsiders.[34] Hence, we should map and define the flows of information in each of our cases, find what roles does the state and other institutions play in facilitating these flows, and try to estimate how good are they compared with the past, and with other cases. These flows of knowledge are of course in turn influenced both by the institutional structure in the national and sectoral/regional levels, and the ruling technoeconomic or research paradigm that dictate who and how R&D is to be conducted.

Fourth, related to the three points above, the writers in the innovation systems, the industrial clusters, and institutional literatures, point toward the importance of market structure. Do the markets consist of many small and medium sized companies? Do few big ones control it? What are the relationships of the private companies with the public sphere? With the universities? Are there many local services and related industry firms that supply needed inputs to the industry?

The fifth variable that we should look at is the internal organization of firms. What are the roles and jobs of employees, what parts of the workforce provide inputs to the decisions making and innovative processes? What are the social and institutional origins of companies' founders and entrepreneurs? How mobile is the workforce? How skilled is the workforce?

These five points are related not only to the behavior of domestic institutions and organization in our two different level of analysis the national and sectoral/regional, but, especially with the greater internationalization of the world economy in later years, to the behavior and connections of the domestic to the world markets and multi-nationals. Thus, we should pay attention to variances in the behavior of multi-nationals, to the different connection and markets (physical, intellectual, and financial) that the domestic economy relates to. From where the domestic economy imports technology and finance, to where it exports them? What is the nationality of the MNCs? What are the size, composure, and direction of immigration e.g. low-skill immigration to Europe, or high-skill immigration to the US? What are the relationships and influences of the Diaspora on the domestic economy? To where, why, and which activities are relocated by domestic firms when they try to create a global presence?

Two more variables that need further inquiry before we can begin our full-scale research are variables that relate to the national macro level politics. While we touched on politics in its broader definition when we look into the power structure and decision-making in the firms themselves, between the firms and the public sphere, and when we try to understand the flows of information and research paradigm in the system, we haven't asked ourselves anything about the way that the state shape its own policies and the ways it implant them. The first a major question to be asked is how decision makers conceptualized the question of economic development and the options open to their societies? It might be that looking back we can analyze the economic problems faced by these peripheral states as very similar, but it might not be the case that policy makers on all these states defined them in the same way. Thus, our first political variable should answer two questions. First how the economic development was conceived in each of our cases, and secondly how the solution was conceptualized. Thus, we should not try to find only whether there existed different economic ideology in our cases, but also what were the different conceptions of economic problems and solutions that interact with that ideology.

Secondly, we should analyze and compare the different development agencies that were created in our different cases, what was their mandate of operation, how well were they insulated from the political debate, what other organizations and individuals did they work with, where did they recruited their personnel, what did they try to create and how their goals and structure were changed over time? In short, we should try to compare the different technonationalism ideologies, state capabilities and goals in our different cases, and understand how they in turn influenced and were influenced by, the rest of the innovation system.

In short, surveying the literature we asserted that investment in R&D and technological diffusion have a tendency toward underinvestment under free market conditions, especially in peripheral state, therefore without some specific state policies we should not expect industrial innovative leapfrogging. Looking for the causes and conditions for sustained innovative system we hypothesize that social institutions that spur information sharing, collective learning, specific financial institutions, human and physical infrastructure, and specific business structure in the domestic level, coupled with as yet undefined relationship with the global, are the explanatory variables given.

The Role of the Military in the Israeli Software Industry

This part of the paper focuses on the role played by Israeli Military (the IDF) in creating and sustaining the highly innovative Israeli software industry. I will concentrate on the IDF's central computer unit (Mamram), and especially on the School for Computer Related Profession (hereafter: "the school") the main programming, software engineering, and computer users training unit in the IDF.

I decided to concentrate on the Mamram as a first case study for three reasons:

1. While the creation, development, and activities of the Mamram were and are highly important for the Israeli software industry as a whole, the Mamram was not created with that goal in mind. The Mamram was created in order to solve a specific perceived needs in computerization in the Israeli Defense Forces. Thus, the Mamram is a clean case to show the long term positive institutional affects of state intervention, without ever needing to attribute

to the state the kind of almost godlike prophetic strategic vision ascribed to it by statist theorists.

2. The same leaders and policy makers that decided on the creation of Mamram and other early age defense related high-tech agencies, are the same group of people that in the second half of the 1960s and all through the 1970s formulated and implanted Israel's science and technology and industrial policies.[35] It is therefore important to follow there early activities and achievements in High-Tech and R&D, if we want to understand why they later conceptualized Israel's economic future solely in the area of "science intensive" activities, in a time that Israel had a very low scientists and engineers per capita ratio.[36]
3. While there were and are no official recognition that one of the goals for the existence of the Mamram and the School, is the sustained development and upgrading of the Israeli software innovation system, there is an official and unofficial recognition that it is. Thus, some of its activities are devised around the unofficial goal of supplying a public good for the Israeli software industry. Is would be shown later this unofficial recognition is exactly what makes the dialog between The School and the industry so robust.

For these reasons a case study of the Mamram not only highlight an important part of the Israeli software innovation system and explain its development, but also enlighten us as to the long-term institutional affects the state and the continuing dialog between the state and industry has on the development of the IT industry in a clear cut case were the state didn't intend to have these positive influences. Moreover, this small case study would give us the first cut at understanding the development of the overall Israeli Science and Technology and Industrial policies from the 1960s onward.

The data for this part of the paper was gathered mainly through over 80 long open interviews (1-2 hours each) with Israeli IT industry leaders (founders of and CEO of companies, VCs, former chief scientists, etc'), of which, over a half were conducted with former officers of the school, the founders of software programming and software engineering in the IDF and the state apparatus, and the managers and cofounders of the leading private professional schools. The interviews were conducted in December 1999 and the summer of 2000, complementary data was gathered from archives, secondary sources, and state and industry associations' reports.

I contend that the military should not be viewed only as a mechanism for producing highly -trained professionals and a source for new technologies via various spin-off, but as one of the main nodes in the national innovation system that diffuse information, spur collective learning, and creates standards for the whole industry.[37] Viewed as an intrinsic part of the Israeli software industry the Mamram play six distinct important roles:

1. Software development and engineering standard setting: as the largest organization, one of the main customer of software products, and one of the main trainer of programmers and system analysts and designers, in Israel, the military through its decisions, help to solve the basic problem of collective action which is inherent to standard setting.[38]
2. Human capital - the IDF, unlike its American counterpart, has relatively limited resources and cash. This constraint together with the knowledge that its best people will stay only for the initial period of 5-6 years, and a reality in which it has no way of hiring mid-career professional to serve for long period of time from other institutions, results in very intensive investment in human capital, and an extremely large amount of responsibility for R&D given to very young personnel, compared with equivalent civilian organizations. These two factors coupled with the IDF training system based around Planning By Situation (PBS) approach of instruction material development, produces high quality and highly trained personnel prone toward, and experienced in, R&D product producing approach.

3. Due to the distinct timing of its creation the Mamram had two unique roles: (1) The IDF through the Mamram become the main stimulator for the diffusion of computer technologies and their application to create new software and hardware products throughout the Israeli economy.[39] (2) Since the Mamram was set up many years before computer science was an established Academic discipline, and due to the fact that it develop its own concept of software programming seen as profession aimed at creating operational systems, the school helped to create, and is the center of, a major paradigm of software development in Israel. I will term this paradigm, the applied or professional approach to software.[40] Consequently the school created a distinct approach of teaching software programming, and trained professional trainers in that approach. As these trainers later founded of private professional computer related profession schools, this paradigm experienced a fast diffusion and acceptance in the private market.[41] Moreover, the school is the only institution in Israel devoted to the creation, gathering, teaching, and diffusion of knowledge of software as a professional activity as such, i.e. not only toward revenue generating professional knowledge. Examples for the school activities in the defining of this research paradigm are the development and implantation of certain software engineering methods in Israel.
4. Network: Mamram graduates, throughout their own ranks, and with the contacts they create with the reserve personnel (i.e. civilian computing professionals serving annually for short period of time), create a dense network of knowledge, recruitment, and venture capital. This network not only gives them an advantage throughout life, it also enables them to tackle and solve R&D problem that are beyond their present knowledge or skills. As one of my responded said: "I am not afraid to take any consulting job in the field, because I know that in the worse case scenario I am at most four phone calls away from a world expert for any

software related problem."[42] The existence of such dense and large network gives the Israeli industry as a whole a distinct advantage.

5. Collective learning and diffusion of knowledge: the school creates three major settings of collective learning. First, through the school's activities in the creation and dissemination of software teaching and learning material, the school sponsors multiple activities of collective learning. These activities, due to the fact that the school is highly constrained in budget, but has the unique capability to bring as reserve personnel virtually any computer expert in Israel that it needs for short periods of time, are done through the creation of small project teams. These are composed from military and civilian experts from multitude of firms and academic institutions, working and sharing their knowledge together in a way that would have never been possible outside the school. Second, Israel's best experts for various parts of software R&D teach and produce written textbooks and instruction material only through the school's reserve duty. These experts, through the courses they give in the school, share their knowledge with their students and diffuse it throughout the Israeli software industry by way of their written instruction material. It important to note, that these people are in key development and managerial positions in private and public firms and would have never shared their knowledge in any systematic way without the school. Third, the reserves personnel themselves are constantly exposed to the knowledge gathered and created in the school, knowledge they take and utilized back in their firms. In addition, the school employs consulting firms that not only bring knowledge to the school but are also exposed to many of the school activities, which they than utilize to offer commercial activities in the private market. Lastly, all these are strengthen by the IDF's policy of not holding any property rights or patents on technologies developed under its sponsorship. Accordingly, the school

graduates freely transfer technology to the civilian market when they finish their service period.

6. The school and the private schools founded by its graduates, serves as an important point of contact for knowledge from foreign software-tools development companies like Oracle, Sun, or Microsoft to the Israeli system. Thus, before a new development-tool is released, the schools have often already acquired the knowledge from abroad and organize the courses to train professionals to use these tools.

This rest of the paper is divided as follows, first a short description of the success of the Israeli software industry and its uniqueness compared with the two other successful cases of India and Ireland, is given. A description of the history and working of Mamram and the school follows, with a short case study of one routine activity of the school -- the restructuring of the software design course -- one of the major software engineering courses.

The structure of Israel's software industry

The Israeli software industry is a phenomenal success story, annual total sales rose from 380$ millions in 1989 to over $3.5 billions in 1999, with an even more spectacular growth in exports, from $5 millions in 1984 to over $2 billions in 1999.[43] A very interesting feature of the Israeli industry is the large role that the local market played in its development. The local market was already well developed in 1984 with $370 millions in sales, while export sales only started with $5 millions. Local demand continued to develop, inducing tremendous growth in local software sales throughout the 1980s in 1990s. To clarify how the local market was already relatively large before the export boom began, it might be worthwhile to note that as late as 1993 the total sales of the indigenous Irish software industry as a whole were $368 millions. Four years before in 1989 the Israeli local sales were already over $315 million dollars, i.e.$438 millions in 1995 purchasing power terms.[44] Also unique in the peripheral states is the development of the whole software industry around small to medium size firms that develop their own products. India's and Ireland's software industries, for example, relied to a greater degree on MNCs, either as a regional HQ platform, in Ireland, or as "software factories" offering low cost good quality programming services, in India [45]. Accordingly, when we compare Israel to Ireland and India, we see that while the size of the Israeli software industry in 1993 to 1995 was vastly larger, Israel's exports were less than 30% of total sales, where in India's and Ireland's exports were 60%.[46]

Another feature of Israel's software industry business structure (in fact the structure of the Israeli high-tech industry as a whole) of indigenous firms with their own innovative products, is the number of Israeli high-tech firms listed on NASDAQ. In November 1999 ninety-six Israeli firms were listed, the highest number of firms listed on NASDAQ after the United State and Canada, only two Indian, and fourteen Irish (four of which with no connection to high-tech whatsoever) were listed.[47] In addition, Israeli firms are also listed on the German Neu-Market and the London Stock Exchange, and on the 3/26/2000 PictureVision, was the first Israeli firm to announce its decision to list a subsidiary in the new Japan's high-tech stock exchange. Another indication of the "Silicon Valleyness" of the Israeli industry is venture capital investments in Israel. The Amount of VC investments in Israel become the third in its size in the world, after Silicon Valley and Massachusetts, with VC investment in Israeli companies of 1.003$ Billions in 1999, and over $3 billions raised by Israeli VCs in 2000.[48] Venture Capital industry was virtually non-existent in Israel before 1992. There was only one small US-Israeli fund started in 1985, before the government funded the $100 millions Yozma initiative in 1992. This initiative lured American VC firms to invest regularly in Israel by securing 40% of the fund (i.e. 8$ millions) for ten $20 millions private American-Israeli co-managed VC funds, and another publicly controlled $20 millions fund.[49]

In sum, the Israeli software industry is not only a fast growing industry, it is also highly innovative as it is established around a "Silicon Valley" like structure consisting from a multitude of small to medium firms that developed original products. Historically this industry focused on a relatively large and sophisticated local demand. Thus, when we try to understand how the Israeli software industry achieved worldwide success we need to understand the specific domestic institutional history that propel and enable the market structure and technoeconomic paradigm of highly innovative small firms. In order to do that it might be worthwhile to sketch the history of computing in Israel trying to identify specific institutions and timing within software development was introduced, developed and understood.

The early history of computing in Israel

The official history of computing in Israel began before the creation of State of Israel as an independent national identity. In 1947 the Advisory committee of the Applied Mathematics Department of the Weitzmann Institute, consisting of Albert Einstein, Hans Kramer, Robert Oppenheimer, John Von Neumann and Abram Pais, recommended that the Institute build an electronic digital computer, making Israel the first not-yet-a-state to commit itself to computing.[50] However, the 1948 independence war and the continuing security threats quickly propelled the development of computing in Israel toward a different track. While academia, mainly the Weitzmann Institute of Science continued to develop three generations of scientific computers called the "Golems," very quickly the defense apparatus and the state bureaucracy become the torchbearers of software development in Israel.[51]

Probably the first unit in the Israeli defense complex to develop and use computers was RAFAEL (the Hebrew abbreviation of: Armament Development Authority). RAFAEL, the first, leading, and for many years almost the only body in Israel to conduct high-tech R&D, already started to use computers in the 1950s. In 1956 RAFAEL, then still a part of the IDF, developed an analog computer, one of the first analog computer ever made and developed in Israel. In 1959 a more sophisticated analog computer "Itzik" was developed in order to enable larger scale simulation. After more researchers were recruited in the end of the 1950s RAFAEL also developed few digital computers. In 1966 RAFAEL in a joint venture with a private company called Elron created Elbit. This high-tech company started by developing mini-computer, and become the largest Israeli high-tech defense company with worldwide activities.[52] More interesting for our research, is the fact that in 1959 RAFAEL together with the military intelligence, joined forces with the air force and logistics to call for the acquisition of a large-scale mainframe computer in the US.[53] In 1959 a joint committee of RAFAEL and the IDF to inquire to the needs for data processing the Israeli security forces, advised to buy a digital mainframe computer in the US. In 1960 Mamram ("Center for Computers and Mechanized Records") was created, a Philco computer was bought in the US and Mordechi Kikion was transferred from RAFAEL to become its first commander.[54]

The second major software programming effort was conducted in another important part of the state bureaucracy, the finance ministry.[55] In 1960 the deputy income tax commissioner Emmanuel Sharon, while studying economics in the Hebrew University, was told by a fellow student that the tax collectors, after giving the taxpayers hand calculated assessments, were checking these assessments using the computer of the central bureau of statistics.[56] Moreover, Sharon found out that these calculations showed that 15% of the assessments given to taxpayers were wrong. After checking this out, and being told more about the electronic computers from another student friend, Emmanuel Sharon decided that it might be more useful and efficient to computerize the tax assessment before handing them to the taxpayers. After a presentation of his ideas in the ministry, Sharon and the head of the ministry's SHAM ("Mechanized Processes Service") conducted a 43 day tour in Europe where they were introduced to the uses of electronic computers in ministries, municipalities, and public companies in France, Germany, Switzerland, the Netherlands, and the UK.[57]

After their return, a decision was made to buy an NCR computer. Due to the non-existence of software programmers in Israel, a special course very similar in organization to the Mamram's core programming course, was arranged.[58] The study material was constructed from translated books, NCR's material and some visiting lecturer's notes, in a method described by Sharon as, "we read the material at night and taught it in the morning".[59] The programmers were recruited mainly from students that pass an entrance exam modeled on assembler language programming language. The course graduates, together with Mamram's, later become the leaders of the Israeli software industry.[60]

In about the same time another larger computer unit was organized in a similar way in the state bureaucracy, the MALAM ("Center for Office Automation"), unlike SHAM the MALAM always worked in an IBM environment. The MLM was later privatized and become the Malam group, one of Israel's largest Value Added Reseller (VAR) and system integrators.

Another important role played out by the state institutions in that pioneering period, was the training of the first system analyzers by the Institute for Work Productivity, a course which was later copied by Mamram, and in later years in the beginning of the 1990's the creation of SEFER -- standards of software engineering, and in institutionalized flows of knowledge and personnel. Decisions that continue to influence the Israeli software industry in our days.

In sum, a short stroll through the history of computing in Israel, teach us that large-scale software training and development began inside the state apparatus, in order to solve specific administrative and R&D problems that required large-scale data processing and simulations. Consequently, a paradigm that sees software programming as an applied engineering profession aimed at developing complete software system, i.e. finished products, evolved as the ruling paradigm in Israel. Moreover, the specific needs and constraint of the state systems, the lack of any software professional and the decision to computerize the IDF that created a need to train large quantity of high-skilled people in short time, induced the development of sophisticated system of software programming teaching and instructing. In addition, the same constraints created a system of elite intellectual potential recruiting (more about that system below) giving computing as a whole, and software programming in particular, high professional status and prestige.

The Mamram and its school

An integral part of the establishment of the Mamram in the IDF was the creation of an internal training unit, the first such unit to be created in Israel. This unit become an independent unit in the second half of the 1990s and is now called "The School for Computer Related Profession" (The School). The training unit evolved somewhat throughout the years until it reached its present shape as The School, however, its basic procedures and operational methods remained the same. Let us begin our analysis of The School activities be first examining the career paths of software programmer in the IDF.

In the end of high school, usually at the age of 18 most Israeli young adults face a compulsory military service of two years for females and three years for males. However, if a person wishes to serve in special positions, or to get special extensive training they must stay for longer period of service (usually any period of time after the first compulsory period is called 'permanent' service and grant the soldier full salary benefits). One of these positions is the position of a software programmer. In order to enter The School's core programming course, the applicant need to agree to stay in the IDF for about three more years of service. A second way to secure a software development position in the IDF is to apply for the academic reserve, a service path where a candidate first finishes an academic degree and than serve for around two more years of service above the compulsory two or three years.[61] As a result, programmers in the IDF serve for around five and a half to six years.

To be accepted into The School's programmers' course, the candidate must pass a day of exams given to specific candidates after the first screening process of the IDF. These two screening processes ensure that those that start the course belong to the intellectual elite. The School usually recruits only people without perfect health profiles, as these are usually sent to combat units.[62] However, the school can choose up to six perfect health profile candidates annually, they must pass a second special day of exams. Those who pass these exams are exceptionally mathematically gifted (or in the language of the school's website, acknowledged geniuses).[63] There is no need of prior knowledge in software programming.

The course length is six months, consisting of around eight to nine hours per day of formal classes, and another six to seven hours of lab practices and exercises. The instructors are either reserve duty personnel (more about the reserve duty personnel later), who are usually industry professionals, academics, or former soldiers of the school, or the school personnel, which are soldiers who record especially high scores in the basic courses in the past and were given the opportunity to stay and become part of the school. In the past only around 50% of these accepted to the course finished it. Today, due to the growing demand in the IDF a higher percentage actually finish the course. The school trains around 300 programmers each year.

The school also trains system managers (both small and large), and to ensure the full absorption and optimal use of information systems throughout the military, the school trains around 500 application instructors each year, who go throughout the IDF and train users. Moreover, the school also sees as its responsibility to create sophisticated users out of the IDF commanders, and to train them to the level where they could relate, identify, and define requirements to the system analyzers and programmers in their own units.[64] To achieve that end, the school conducts each year several system analyzing courses for managers, and offer personnel training courses to the IDF's high commanders. From time to time, the school also opens a special course in educational and learnware software writings, to retain high level of these specialized skills.

The school also gives its programmers extensive advanced training throughout their service and offer on a constant basis, professional courses on specific platforms, systems, or languages (e.g. Oracle, Sun, Linux). A year after they finished the core-programming course, The School's graduates return for one-week basic software designing course. After another eighteen months of service they return to the school for an advanced design course of five weeks, the design courses will be described fully in the a short case-study that describe their last routine restructuring.

After the advanced software-designing course, the programmers career fellow specific professional path. These who specialize in infrastructure continues to take course that advance them as infrastructure specialists, and these specializing in application programming usually go to a year long (one day a week) system analyzer and designer course and project management courses. In order to take the last two courses an enrollment for another short period of service is usually required. In the past, soldiers were also offered professional courses toward diploma as a computer technician or engineer. Moreover, for these who wish to study for an undergraduate degree in computer science and technological management, a leave of one day a week is secured throughout the academic year. According the School's web page, a special program toward an undergraduate degree in an Israeli university will be offered in the near future.[65]

The typical advancement of Mamram's graduates is as fellows. After the core course the programmer usually operate as the junior member of a project team "the apprentice of the magician apprentice" in the description of one of my interviewees. After the basic design course they usually become deputy team leaders, and after the advanced design course (i.e. 2.5 years of experience) they either become team leaders or go to officers course. Thus by the age of 21, Mamram programmers are already well-developed programmers with vast experience. A former school official with a years of experience in the private industry illustrated this stage of the programmer development in these words “This 21 kid already worked in multiple projects, sometimes even in different units, he is an efficient and experienced programmer by that stage”. Or, in the word of another interviewee “There was no way I would have gotten to the same position I had in the IDF at the age of 20 in any civilian organization, no way, never!” Around one of four programmers also acquires a long (i.e. 1-2 years) experience as the manager of a full scale programming team before they leave the army. Around one of ten become section head, responsible for a specialized sub-unit with tight budget and long term projects management and control responsibilities.

Moreover, due to the fact that most of the IDF’s defined its programming needs for many years as specific software products that were developed either internally or externally with Mamram’s help, and that the computer units were always defined as service provider units, a very high level of attention is given to training these programmers to enable them to understand and define their customers’ and units’ needs. Thus a Mamram programmer, when leaving the army, already has several years of experience in analyzing and defining the needs of the ‘market’ they operates in, and in creating products to solve these needs.

In comparison with the equivalent computer units in the American army, the Mamram is financially limited and its commanders need to be very frugal, careful, and persistent in procurement activities. For Example, one of my interviewees recollected what he needed to do in order to buy a new and long overdue computer for The School.

> “It was unimaginable, I had negotiations with the one specific company from which we needed to buy the computer, the only good argument I had in my arsenal is that this computer would be used to train all the IDF personnel and they would than create demand for the company products. Being very persistent I lowered the price to 10% of the original price, the computer was essential for the unit, and still, the commander of the whole Mamram and I had to spend hours in special meetings. I think I spent tens of hours of work just to find the money.”

Another of my interviewees recollects similar memories from his time as a commander in one of the Navy's computer unit "In order to get a year and a half old computer, you had to do various kind of manipulations and politics."

Those budgetary limits have, however, few advantages in the training of Mamram's personnel. The first of these is that these constraints act as what Michael Porter termed *Selective Factor Disadvantage.*[66] Hence, this fiscal constraint encourage the R&D personnel to create and devise innovative ways to compensate for it, and supply incentives to acquire a far deeper knowledge and capabilities of any piece of equipment. A process that was described to by a former commander of the computer unit of semi-civilian manufacturing part of the IDF, now a CEO of a venture capital firm, as

> "The systems <hardware> in the army are relatively old, by the way this is a subject which is very interesting to think about its implications, as the level/class of improvisation <you do in order to supply workable solutions> needs to raise in a hysteric rates, since there are not a lot of financial resources. There is no such thing like what we usually do in the civilian market, if it is no longer working throw it away and buy a new one".[67]

In a follow up question the same interviewee who described his experience of buying a mini computer for The School answered,

> "How do you deal with it? <The severe budgetary limits> you deal with it by working around the clock, 14-15 hours of work a day. As a matter of fact, these who didn't work until the middle of the night soon found out that they developed a "bad stigma" for themselves. The other way of dealing with it is the quality of the personnel, not only the Mamram recruit whoever it wants, but you also have that deal with the reserve personnel, a thing without a parallel in the civilian life… some of them you will never be able to hire in the private market, they need, however, to serve their reserve duty and you have the whole industry to chose and pick from."

The other more mundane affect lies in the experience gained by the Mamram management personnel. Not only do they learn how to manage their units under constraints, with tight budget and deadlines, they also have the unique experience of working with a very high quality workforce, and managing by the ages of 21-23 the most experienced personnel of the private industry.[68] This shared experience also helps to explain the highly egalitarian work place behavior of most Israeli IT firms.

The applied software paradigm – the affects of the Planning By Situation model

As mentioned beforehand, the Mamram with several other state agencies were the first to train software developers, created a specific paradigm of software development -- the applied, or professional software paradigm. This paradigm sees software development as an engineering profession whose aim is the development of products, systems, and solutions that answer to the growing need of a modernizing market, military, or civilian. As the first Academic Computer Science departments were not opened until 1969 and at first offered only graduate degree, this paradigm became de-facto the hegemonic paradigm in the Israeli Industry. Since The School became the major training institutions of software professionals and the main center for instruction material and course creation for many years, it also became the center of this paradigm. The paradigm was than diffused and reinforced throughout the Israeli software industry both by Mamram graduates that grow to be industry leaders and by the many private schools for computer related profession that evolved in Israel since the beginning of the 1980s. Former officers of The School and Mamram graduates founded all leading private schools.

In the 1980s the IDF developed a special approach for the creation and development of courses and instruction material, the Planning By Situations (PBS) model. The PBS model was quickly diffused to the private market through The School's personnel opening private training center after finishing their service. The PBS is critical in the strengthening and locking-in of the applied or professional approach to software paradigm throughout the industry. In the heart of this engineering like approach to training, lie the assumption that any training that the IDF and similar organizations give, is given to achieve one goal: the creation of professionals that would operate in a specific role in the organization. In other words, the PBS is a pragmatic holistic approach to the creation and teaching of discrete bodies of professional knowledge.

The building of a course by the PBS model start by analyzing the responsibilities of a specific profession; what are the duties of the professionals working in that role? What do they need to operate? What do they need to produce? And what accompanying skills, e.g. ability to work in teams, they should acquire? In a similar way to any engineering system analyzing project, the course developers analyze both (a) the activities of professionals in the field and define the 'current state description,' termed in the PBS model as 'analyzing the working process components', and; (b) the profession components (the required skills the specific profession necessitate). After the profession's definitions are analyzed, the course developers delineate the course skeleton. The backbone of the PBS training model is constant practice, the model demand that this practice should be conducted in environment that simulate as closely to possible the real working environment. Thus, the course skeleton consists from a continuum of "real-life simulation exercises" that are termed "stairs."

The definition of the course's progression begins with the last exercise that is called the "summarizing exercise." The exercise simulate a situation that is as similar as possible to the real life required operation of the graduate, the exercise should require from the student to prove satisfying level of control of all necessary skills of the profession. For example, for many years an appropriate summarizing exercise of Mamram's core programming course was a four days team mission of designing and writing an interpreter.[69]

After defining the summarizing exercise, the rest of the exercises are defined as a continuum of progressively more complex tests of knowledge in all defined skills that climate in the summarizing exercise. The exercises are called stairs, therefore, as each requires the student to prove higher operating level in all the defined skills while they progresses closer to acquiring the profession. After defining all the stairs the course developers define the sub-profession that the profession consist of, and develop the specific classes, and decide the order in which they should be acquired.

An example should highlight the major difference between the PBS to regular software training models. If for example, an institute or a private individual want to specialize in web-programming the courses that would be usually offered are courses in specific programming languages and tools, from which they should pick and choose and fill the gaps on the wire, e.g. Java, Dreamweaver, etc'… A training institute which develop a course modeled on PBS would try to understand what a "web programmer" profession is, and offer a tailored mix of specific technology courses with some added accompanied skills classes, e.g. reading design requirement documents, or the principal of good distributed database designing, and offer this package together as one course, which give a professional diploma. The result is that while software professional in other countries that want to specialize in web based programming, might need to go to four or five course both in specific programming language and in specific disciplines (if they can even find them) knowing that large percentage of what they learn might be irrelevant and either too narrow or too broad to help them in their daily work, their Israeli counterpart working in an environment where software is viewed as a profession can usually find a complete course in specific sub-specialization. That institutional environment is self-reinforcing, as IT professionals, institutions and schools are looking for, developing, and paying for professional 'packaged' courses.

This approach has naturally enough some weaknesses compared to academic education, especially in regard to depth in particular subjects, and knowledge of complex mathematics. However, graduates of Mamram's core course are professionals trained to work as developers, work in and with specific methodologies of development, have a common professional language, acquire extensive knowledge in software engineering, and most importantly see software writing as a profession whose main goal is to produce solutions for practical problems, and are highly trained to do accomplish that specific goal.[70] Moreover, as this approach and the system of private schools that thrive on it, exist side by side with the Israeli research universities and academic institutions that are world renown as center for research and education in mathematics, computer science, and electric engineering, it seems as these two approaches can mutually develop and enhance the level of software programming in a national setting.

The affects of Mamram and the PBS model on the private market are widespread. Israel has a broad range of competing private professional schools for computer related professions that were founded or managed by former officers of The School. All of whom brought PBS model with them from the IDF, and are now finding that it give them an advantage when trying to move their activities to a global scale. The Israeli software professionals can usually have more options in finding more professionals oriented complete sub-discipline courses than their counter parts worldwide.[71] Furthermore, the students of these courses gain widely accepted professional diplomas, some of which, like system analyzers, are now a recognized profession with its own association. This recognition has far-reaching affects, for example insurance companies tailor specific policies and benefits for system analyzers, and demand private firms to employ them in order to get various kinds of professional liabilities insurance.

The founders of these schools readily attribute their success to the training and course development methods they were taught in The School. A founder and managing director of one of the leading school recount how he and his partners diffused the PBS to the private market.

> "We took a lot of success of the training models and methods of The School, and as a matter of fact we copied them into the private market, both in the creativity in course creation, and teaching and working methods… we took a lot from The School and we are very proud in it, I can show you that in all of our presentation we present ourselves as Mamram graduates, it give us an enormous competitive edge."

The specific influence of the PBS method as the locking-in mechanism of the applied or approach to software paradigm in the Israeli market and the affect that paradigm has on the whole concept of 'software programming' in Israel, is also apparent when these schools try to widen the activities to a more global scale. According to the same interviewee.

> "We almost never add any value to the official courses that the software and hardware vendors create. We are adding value when we create packages to offer specific "professions" courses or specific professional specialization courses. Those we create with the PBS model. We now start to work in Europe, where we work with local schools and the major American vendors like Microsoft or CA, we found out that the PBS and the packaging of courses approach is unique, usually the courses offered worldwide are at two extremes either highly academic, or very, very narrow…. The concept of professional software training in most of the <continental> European states is somewhat backward compared to Israel; it is probably very disappointing to the European but very interesting as a market opportunity for us. Most of the PBS packaging things we do here are not known there and they are very keen to get them. For example, CA Europe wants us to export everything we done and develop in Israel, to Europe"

In short, The School with the private professional schools founded by its graduates, was crucial in the creation and diffusion of a specific paradigm of software development, concentrated around the concepts of software programming as a profession whose role is the creation of software solutions and products. This paradigm was strengthen by the diffusing of the IDF's PBS training model to the private industry and is now the at the base of a mutually reinforcing institutional system, with the results being first, constant high level of professional education to IT professional in Israel. Secondly, the creation of an extensive body of knowledge and social conception that sees software development as engineering discipline whose goal is the creation of new products and systems.

The transparent borders between military and industry – the role of reserve personnel

One of the most important ways in which The School operate as a major node of information creation, sharing, and dissemination throughout the Israeli software industry, is via its extensive use of reserve personnel and its unique exploitation of their *civilian technological* *knowledge.* This section explore the role of the reserve personnel in The School and the way that their participant in its activities transform it from an isolated military space into a "public space" that provide essential public good.

Reserve duty is one of the prominent facts of life for any Israeli, especially males. Each Israeli citizen that served in the IDF is required to serve for up to 30-40 days a year in reserve duty. Men serve usually until the age of 50+. Women are usually exempt, except these that have specific skills and training, these are usually called to reserve duty until the age of 26.

For most people, reserve duty is done with an organic unit, which train and serve in some front line or guard duty for a preordained couple of weeks each year. Relatively smaller amounts of people are retained in the more technical units and are recalled to serve as technicians or developers to continue the same role they had before. Even smaller amounts of people are recalled to training units to be trainers in the army courses. Of these training units, The School, being the only large scale technical and technological training unit in the IDF, is unique in the ways its use and recruit its reserve personnel.

The reserve personnel of The School are usually industry experts, while many of these have served in The School in the past, many more are recruited from the Academia and industry. They are given three main responsibilities:

1. Teaching – the reserve personnel are usually responsible for the more complex or industry unique material in the basic courses, and are almost solely responsible for the advanced courses, e.g. system analysis, design, or project management. Where at least 97% of the classes are taught by reserve personnel.
2. Developing and upgrading The School curriculum – reserve personnel serve in the steering committees of The School courses, and make up most of the teams that develop a upgrade the programming and advance courses.

3. Creation and writing of instruction and reference material – one of The School main activities is the creation and writing of reference and instruction material. This is highly important as The School is the central organization, if not the only one writing original advanced reference material in Hebrew. Most of this writing is done either by or with the help of the reserve personnel. Moreover, that creates a unique environment for the processing, sharing, and dissemination of invaluable professional and tacit knowledge. Since the people who write this reference material are industry professionals that would have never had the opportunity to write and share their knowledge in their daily life. In addition, due to the fact that serving in the reserve duty is seen as a citizen duty and is conduct in a public space of a national service environment, and as there is usually a delay of at least one year until this material find its way to the civilian market, the level of trust and information sharing is exceptionally high.

In addition to those activities, we should note that without The School many of these people would have never been involved in any teaching or other knowledge acquiring activities whatsoever. Thus the School prevent knowledge from getting "lost", or being stuck in specific bottle necks, and help diffuse it throughout the system.

The relative prominence of the reserve personnel in The School is evident if one look at the manpower available to The School. The School has it its disposal about one hundred regular full time personnel, and around four hundred reserve personnel. Thus, The School has about 20,000 days of reserve duty, which are equivalent to one hundred work-years. In short, the same amount of time it has from its regular serving personnel, but given by a much more professional and experienced sources. Therefore, it is not surprising that the managing of the reserve personnel is highly important, in the words of one of my interviewees, "very shortly after you are nominated to a section head you understand that your first priority is managing your reserve personnel, they give you the same amount of time as your regular soldiers, but there is no comparison in their professional level and quality."

The reserve personnel serve as a two-way conduit, from the industry and academia to The School and hence the IDF, and from The School to the industry (see diagram in appendix 1). Moreover, the school serves as a point of contact and network to other industry professional, and between the reserve personnel, The School's regular duty personnel, and their students creating a strong multi-cohort network, which is rare among other teaching institutions. In addition, The School reserve personnel are privy to all of the material written and created in The School, which they bring back with them to their civilian life. A testament to the strength of these information flows can be found in the answer of a cofounder and managing director of one of the private schools when asked about the source of knowledge and materials:

> "When we left the IDF, we had a very strong influence of The School, but now you have to remember that my partner, that rewrote our professional conversion course, was in the exact same time very active in the steering committee of the upgrading project of The School's basic programming course, and was leading its creation. So from where exactly the knowledge arrives and to where does it flow? I have no way to tell you."

The same was true all along the 1980s, as another interviewee who lad the rewriting of the system analyzes course both in the IDF and the civilian market was happy to recount:

> "I can tell you that there was a sort of a positive conspiracy between The School and ourselves – we write material for instructions for the IDF but in exchange we also use them for our interests -- sort of a gentlemen agreement that we and The School take material form one another. There was a lot of interaction, oh yes there certainly was a lot of interaction."

A few examples might be appropriate to illustrate the uniqueness of The School. One of The School's officers told me how he specifically encouraged is reserve personnel to create references books on the basis of their tacit knowledge:

> "For example I had a reserve instructor that was expert on the outside discipline of software engineering, I encouraged him to write a few classes on the subject. It is not exactly the main

> stream of the knowledge, but there are in these class notes, a huge amount of professional tricks and capabilities that you cannot learn from any other source, only a person who had done it many, many times had developed these capabilities. For example, when, how, and why to write a log, it is not really related to mainstream software engineering, but when you actually go and write a system, you find that this knowledge is invaluable. This guy never wrote before, and would have never found any other place that would let him write this kind of instruction material."

Another contrast is between the behavior of these industry experts when they do agree to teach in civilian course, and when they teach in The School doing their reserve duty. While almost none of them agree to write or develop instruction material of their specific knowledge for the civilian schools, and while they hardly participate in the creation and maintenance of the private courses material and instruction, they are actively participating and sharing their knowledge in the School's environment.[72]

> "When I go to The School to teach software engineering course, about 99% of the people are reserve personnel, they talk with one another, make sure that they teach the material in a coherent way, develop new instruction material, lecture to one another about their specific fields, and I do the same. But when I go to a civilian school and they organize a team meeting, I keep my mouth shut, because I know that this material is worth its weight in gold, I do not develop any written material… There is no way they can pay me enough to write these things, and I tell you, anyone who is worth his salt behave in the same way."

These symbiotic relationships go in both directions -- reserve personnel that developed and taught a course in the IDF get all the relevant material to that course for their own use. Student that participate in courses and lost their material can usually get another copy even after the left the IDF. Moreover, due to the fact that The School officials are aware that these reserve personnel are going over and above the call of duty to help The School, they are very willing to help them in their requests. An example is given by one of The School's section heads,

> "One of my reserve people that also work in a civilian professional school called me and told me that he they cannot write a specific set of classes, so I immediately gave him a booklet of 120 pages with questions, examples, and instructions wrote by our best instructors. This is a guy that when we couldn't develop something and he heard about it he force three of the industry leaders to volunteer for a whole day to come to help us to understand a specific very complex process. So of course I helped him when he needed my help."

Another interesting effect of the involvement of industry leaders in The School's activities is the positive influence that they have on the motivation and behavior of The School's regular personnel. In almost every interview I heard a variation of the following recollection of a former officer of The School:

> "Soldiers have a huge motivation to be good in the IDF. Because if you are good in the IDF, you get a vast exposure throughout the industry, and you will get very good offers even before you leave the army, and you should remember life only begin at 24. It happen to me, one example was an offer to work in the most interesting team of a leading software company accompanied with the most competitive terms offered in the market, I was not even 24 and this was just one offer of many."

In short, the extensive use of reserve personnel in The School blur the borders between military and public space. Moreover, as The School not only creates an environment of information sharing, gathering, creating, and disseminating, but also act as a node where information flows not only in and out but between industrial firms and academic organization, it grant the Israeli software innovation system a vital and unique public service.

The School's contribution to the Israel software industry – a case study of the design course

This section would conclude our analysis of The school's role in the continuing vitality of the Israeli software system of innovation, by presenting a detailed case study of the routine upgrading of one of The School's advance courses -- Software Design -- the software engineering course given to the IDF programmers about two and half years after they finish the basic course and start working as programmers in the IDF.[73]

The use of the software design course is appropriate for several reasons. First, it is an advance course, but not one of the most advanced. Thus, not only every programmer that have been through the basic Mamram programming course is given this course, but its upgrading was not done as a result of an overall industry decision, but was an internal affair of The School. Secondly, the new format of the course is unique throughout the industry, and the course is probably one of the only such courses given in the world, especially in the scope given in the IDF. Therefore, due to the fact that this course teaches specific software engineering and designing methods that as a result become the standard throughout the IDF, and are now diffused to the private industry, the upgrading project was a de-facto standard setting act for the whole industry.

Software designing and engineering, like any other product design problems, is an unsolved discipline.[74] In its boarder definition software engineering is the domain between the unquantified, undefined, and amorphous reality, and the translation of its reality to defined logical structures that can be transformed to programmable source code.

There are two phases to an ideal software engineering process, first, the system analysis phase. In this stage system the analyzers sits with customers and in the end write product definition and the accompanied requirements. Ultimately this document answers what the customers really want and the two sides, the software programming side represented by the system analyst, and the market, represented by the customers, know exactly what the product should do and what needs it answers.

The second phase is software designing, in this stage the designer take the requirements and definition document and translate it to specific software units, these are defined up to the level of classes, objects, etc'. The designer also defines how all these units relate and change information with one another, in order to get the functionality defined by the system analyst. For example, the system analyst should decide what is the needed reliability of that system, e.g. 99.99%; the software designer should define how via the use of different software units and specific algorithms, the system would actually achieve the needed 99.99% reliability requirement.

One should be aware, however, that due to organizational and time constraints of most software houses, rarely do they create full high-level and low-level design for all of their projects. Moreover, while the design method and standard termed UML (Unified Modeling Language) offered by Rational, are slowly gaining a wide audience, there are still many different designing techniques and methods in use.

The upgrading project of The School design course started during the second half of 1996. At that time a new officer become responsible for software engineering in the “advanced training” department of The School, and decided to start a parallel project to the PBS based upgrading project of the core-programming course that was already well advanced. In order to do that, he started to organize a group of reserve personnel, and through various resources like The School’s library, his own research projects, and outside information agencies like the Gartner Group, started to assemble knowledge about software designing. At the time the group that is now known as Rational and advances UML was in its inception phase as three unrelated advocates of software designing. After few months of research the responsible officer made a decision to modeled the new course around their proposed standard.

It is important to note that from all accounts both the responsible officer and the team of reserve personnel that he formed around the project, knew from that early phase that that their decisions would affect the whole industry,

> “We decided that we will teach the state of Israel how to design software, it was obvious that only in the IDF one could do such a thing. If you want to know how to properly design software and to solve complex problems, it’s a long process that take a long time to organize and cost a lot of money. Moreover, not everybody would understand that immediately, you must also change the organizational culture, and send a critical mass of programmers to this course. Thus, only in the IDF it could be done… we thought it would succeed because the IDF is a place where 90% pf the people leave after six years, and they than become the leaders of the industry. Moreover, every year you infuse the industry with another cohort of your graduates.”

The consciousness that the decisions made by Mamram affect the whole system and the fact that the national goal is taken as one consideration throughout the Mamram’s decision-making apparatus is evident in an interview with a former overall commander of the Mamram,

> “I saw my role as the commander of the Mamram in the national aspect. In addition to the primary and pure military goals and aims of Mamram, another goal that Mamram has is to take part in the building of the human infrastructure of Israel. This is a role which is highly important

> due to the fact that the universities do not train people in the practical side of software programming in the same way that Mamram does, and do not train them on the technological edge in the way the IDF does as a derivative of our security needs. I didn't always have the support for that from other parts of the IDF, the planning bureau was all against it, and rightly so from its strict financial point of view, but we did it all the same. What we mainly do is (1) to push the use of new technologies, (2) standard setting and method using decision that diffuse throughout the industry, and; (3) the building of infrastructure technologies and the passing of them on to the whole industry. We stuffed The School consciously with the best manpower available and used a lot of reserve personnel, which was good for both sides, the industry and us, and created a lot of information flows... The School is the main way to fulfill our national duty, the building of human infrastructure, a person that learn all these things in the IDF goes to the private industry with all this knowledge and skills in his head."

The involvement and intimate relationship between The School and the industry was apparent as early is the planning and requirements definition phases of the project. In accordance to the PBS model, the first stage was to define the designers' role. Due to the fact that the IDF didn't have designers working at the time, the first step in the PBS model -- analyzing the working process components -- was done solely in the industry and Academia. The team conducted four months of research and observation of the state of affairs in the private market and academia, and about 60 interviews with the leading experts on software design.

After finishing the research phase a bigger team that again consisted almost solely on reserve personnel, started actual development. The project team was divided to five teams of reserve personnel around a specific sub-discipline, e.g. User Interface, Databases. Each team prepared the specific classes, stairs, and exercises of that sub-discipline in accordance to the level and background knowledge the student will need to acquire using the PBS model. The teams met on scheduled meeting to show one another their progression, debated needed changes, and ran quality test assurance. Feedback from the industry was also sought after on a regular basis. The project was managed by the initiating officer, which was assisted with some secretarial help by one compulsory duty soldier, the rest was done by the reserve personnel. It must be important to note, however, that the initiating officer was not only involved in all the team works, the defining of the course rational, and the overall strategy, but also in securing the agreement of the Mamram and the various computer units of the IDF to the different changes in that course and to the need to send all of their programmers after two and a half years for five weeks back to The School.

On his role, and on the IDF reaction to his activities in this project, the initiating officer, now a manager of a private firm recount,

> “Think about it, where would you find such a big organization and bureaucratic system that would let a 23 years old kid to manage a real method project, which is above his evident capabilities? I was in such a system that was only too happy that I do it for it. In that the IDF is unique.”

When course was developed it was presented to the Mamram commanding officers, and in 1997 the first pilot course was given. After which, a few minor changes were done, the course was standardized and is now given on a regular basis. Being and advance course, more than 95% of the classes are always given by reserve personnel. Moreover, all of the results of the projects were given to all of these involved in the project and were quickly diffuse throughout the industry.

One of the project members described the participating of the private industry in the course’ building project as an overall effort, and the diffusion of the information of the collective learning to the Israeli industry as through,

> “All the reserve personnel put their practical knowledge into this collective effort and got all the information created and gathered in the project. Almost all of the big companies in Israel sent “their representatives” to the course… I worked with one private consulting firm, which we brought to consult us and I was surprised to find out that they learn more that what we learn from them, in later versions of their product I saw parts of our project… I know that the reserve people took this information and assimilated the new design process in the private industry. I saw some specific classes in two private schools, I know that the industry is using our methods already, and I know that one specific private university based a whole course around our material… our graduates also come to us to tell us how they now understand what a mess the private firms they work with operate in, and they try to establish some order, and if they wouldn’t the next graduate will, I tell you, we fulfilled our patriotic duty ”

In short, The School, through its various activities and through its involvement of industry leaders as its reserve personnel, become an important node of information creating, gathering, and diffusion to the whole software industry in Israel. Moreover, The School becomes a unique quasi public-space where important collective learning activities are conducted on a regular basis and important network activities, and standard settings are done. In sum, The IDF as a whole and The School in particular should not be viewed only as a factor creation institution, and as a source for spin-off technologies, but as an intimate part of the Israeli innovation system, and as a provider of an essential public good.

Conclusion

While the scope of our case study of the IDF's Mamram and the School for Computer Related Professions is indeed to small and narrow to fully develop our arguments, it does suggest that the institutional variables, identified in the first part of the paper, indeed serve the dual purposes of focusing our empirical analysis, and helping us understand the development of and role of specific institutions and organizations in the Israeli software industry.

As our small empirical study showed the effect of the state institutions and organization on the development and sustained success of the Israeli software industry and innovation system are immense. Although the state of Israel did not create the Mamram and its School in order to shape and sustain the Israeli software innovation system, one could not underestimated the affect that these decisions had on the creation and development of the industry in Israeli, and on its continuous success.

In relation to infrastructure, it is obvious that The School since its inception was prominent in three areas, first, the early creation of skilled human infrastructure and early incorporation of IT in Israel. Second, the creation of a specific paradigm of software development, and the subsequent creation of supportive network of private schools, all of which teach and develop software programming as a profession, and offer software developers in Israel a rich environment of continuous training. Third, the teaching and training of a large number of developers and users, which are subsequently exposed to cutting edge technologies and released to the private market at a very young age.

In regard to knowledge and information flows, and collective learning, The School plays a central role, enhancing the information flows and networks throughout the industry. Moreover, without The School activities and use of reserve personnel knowledge, information bottlenecks would be more prominent, and less collective learning would occur, in the Israeli software system of innovation. While it is not so obvious how the Mamram and The School influence the market structure in Israel, it is obvious that The School create a public space for networks between various companies, and between them and academics. Moreover, the fact that the IDF do not try to obtain intellectual property rights on knowledge developed in its units; create a constant spill over of technologies to the private market.

The internal organization of The School and the software programming in the Mamram and the IDF, is unique in the fact that it not only let, but also force, young personnel to take managerial and critical development responsibilities in major complex multi-million dollars projects.[75] The Mamram projects are also run under a tight budget that drives its personnel to achieve higher level of utilization of their equipment and their development abilities. In addition, due to the fact that most of the Mamram's personnel leave the IDF after six years and the IDF has no way of recruiting experienced mid-career personnel to replace them, The Mamram heavily invest in training and selection of the Mamram's manpower.

Thus, while it is obvious that a large amount of further research needs to be conducted on the Israeli IT industry and the Israeli state technology policy before any conclusions could be reached, and before we can hope to stabilize our theoretical model. Our tentatively developed model has helped us to understand the major and complex role played by the IDF's computer units in the creation, development and sustainability of the Israeli software industry and innovation system.

[1] Example of research in the neo-classical approach are Solow (1957), Young (1995), and Hughes (1988), in the statist approach, Johnson (1982), and Amsden (1989), Wade (1990), Evans (1995), for the market friendly approach see the World Bank's The *Asian Miracle* (1993), and some of the endogenous growth theories based research, for example Romer (1993), and Pack (1993).

[2] For basic arguments about globalization see Ohmae, 1990, Greider, 1997. On the adverse effects of globalization on the advance economies see Rodrick (1997), and Slaughter and Swagel, (1997). For example of arguments about the diminishing abilities of states to effect economic policy see, Goodman, and Pauly (1993), Garret. (1997), Freiden (1991), Frieden and Rogowski (1996).

[3] See Berger and Dore (1996).

[4] I define peripheral states as countries that by 1970s, the beginning of the new wave of globalization, while not belonging to the developed industrial core, e.g. Japan, the US, Western Europe, were more advanced than the developing world.

[5] For studies on institutional theory in the social sciences see Peters (1999), North (1990), Dimaggio and Powell (1991), Hall and Taylor (1990), Hodgson (1995).

[6] For example, the share of High-Tech (mostly IT) of total Israeli exports in 1998 was 71% in 2000 IT accounted for 74% of the growth in industrial production in Israel and in the last decade it accounted for 39% (source: *Israel's economic Overview*. 1998. Ministry of Trade and Industry; *Development of the Information Communication Technologies in the Last Decade*. 2001. Central Bureau of Statistics - Israel). The share of software alone (excluding electronics) of total Irish export 1999 was 34% (Source: Ireland vital statistics, IDA, 2000), and the share of IT of total Taiwanese exports in 1998 was more than 42% (Source: Institute for Information Industry, January 2000).

[7] For the economic theory underlying this ideas see Kenneth Arrow's seminal paper (1961).

[8] On Israel economic and industrial development see Teubal (1987, 1993), Steinberg (1994), Bregman, Fuss, and Regev (1998), Bruno (1989), and Ben-Porath (1986). On Ireland see O'Hagan (1995, 2000), O' Gr'ada (1997), O'Riain, (1997a, 1997b, 1999), Breathnach, (1998), Sweeney (2000), and O'Sulivan (2000). On Taiwan, see Gold (1986), Wade (1990), Noble (1998), Pack (1993), Field (1995), Fuller, at el. (2000).

[9] On the importance of research paradigms see Dosi , (1982) and Samuels (1994), on the importance of institutional setting of the innovation system and market structure see, Piore, Lester, and Malek (1994) Lazonick (1990, 1991), Nelson (1993), Mowery (1996), Bengt-Ake (1992), Kitschelt (1991), and footnote 9 below.

[10] For argument about the effect of history on institution building and the effects of different economic settings on innovation see footnote 8 above and Piore & Sabel (1984), Lazonick & O'Sullivan (1997), Locke (1995), Hall (1986), Saxenian (1994), Gourevitch (1986), Wilks, & Wright. (1987), Hollingsworth, Scmitter, and Streeck (1994), Daniels (1993), Zysman (1994).

[11] For some recent use of patent data in to measure innovational activity see Trajtenberg (1999), and Michael Porter and Scott Stern (2000) for more about global production chains and the de-verticalizition of the industrial landscape see, Timothy Sturgeon, and Ji-Ren Li (2001), Hitt, Keats, and DeMarie, (1999), Gary Gereffi (1996), and Powell (1990).

[12] The first Israeli High-Tech firm to be listed on NASDAQ in 1972 was Elscint, which developed and manufactured computerized nuclear medical imaging devices.

[13] General Propose Technologies are also called Major Technologies, Infrastructure Technologies or Disruptive Technologies, these are technologies that can be used to revolutionize most if not all areas of the economy

[14] That does not go to say that small peripheral economies do not successfully conclude research in basic science, however their overall affect in the creation of globally important GPTs is usually very small.

[15] Gerschenenkron, Alexander. 1962. Economic Backwardness in Historical Perspective; A Book of Essays. Cambridge: Harvard University Press. For some examples of research that followed his theoretical arguments in Europe see T. Kemp, ed., Industrialization in Nineteenth century Europe. 2nd edition. London: Longman. Zamagni, Vera. 1993. The Economic history of Italy: 1860-1990. Oxford: Clarendon Press. Toniolo, Gianni. 1990. An Economic History of Liberal Italy: 1850-1918. London: Routledge. Clough, Shepard, B. 1964. The Economic History of Italy. New York: Columbia University Press.

[16] For research on the economic history of Japan from both the Mieji period and after WWII see Yoshihahra, Kunio. 1994. Japanese Economic Development 3rd edition. New York; Oxford University Press. Sumiya, S, Koji, T. 1979. An Outline of Japanese Economic History 1603-1940. Tokyo: University of Tokyo Press. W. Lockwood, ed. The State and Economic Enterprise in Japan: Essays in the Political Economy of Growth. Princeton: Princeton University Press. Macpherson, W. J. 1987. The Economic Development of Japan c.1868-1941. London: Macmillan Education Ltd. Lockwood, William, W. 1968. The Economic Development of Japan: Growth and Structural Change 1868-1938 (expanded edition). London: Oxford University Press.

[17] The most famous proponent of the statist approach are Chalmer Johnson's research about the Japanese miracle after WWII (1982) and Alice Amsden's story of South Korea (1989). For some market friendly and neo-classical arguments

see Helen Hughes (1988), The World Bank's east Asian Miracle (1993) and Alwyn Young (1995) or Krugman's further development of Young's ideas (1995)

[18] See for example Marie Anchordoguy, *Computers INC: Japan's Challenge to IBM* (1989).

[19] Path dependency…

[20] On institutional theory see….

[21] For the seminal paper that established this finding see Robert Solow (1957). Theoretically if capital and labor grow in exactly the same rate the rate of return will never diminish even without technological progress (and that is why their ratio in the equation is represented as K/L), however it is highly unlikely that they will continue to rise in exactly the same rate for any length of time.

[22] See, Elhanan Helpman. ed. 1998. *General Purpose Technologies and Economic Growth.* Cambridge, MA: MIT Press. Arrow, K. 1961. "Economic Welfare and the allocation of Resources for Invention" In R. Nelson, ed. *The Rate and Direction of Inventive Activity*. Princeton: Princeton University Press. Gene Grossman and Elhanan Helpman. 1994. "Endogenous Innovation in the Theory of Growth". *JEP.* Vol. 8. Romer, M. Paul. 1994. "The Origins of Endogenous Growth," *JEP.* Vol. 8. (1):3-22. Lucas, Robert. 1988. "On the Mechanics of Economic Development." *Journal of Monetary Economics,* Vol. 22: 3-42. Paul Romer. 1989. "Capital Accumulation in the Theory of Long-Run Growth." In, Barro, Robert, ed. *Modern Business Cycle Theory.* Cambridge, MA: Harvard University Press.

[23] Schumpeter, Joseph. 1983. *The Theory of Economic Development*. London: Transaction.

[24] See footnote 20 above.

[25] The national innovation systems school of thought can be seen to have two branches that cooperate with each other, for a good collection of essays which describe the national systems of different countries with a more "European" flavor see: *National Systems of Innovation* edited by Bengt-Ake Lundvall (London: Pinter Publishers, 1992), for a good "American' flavored collection of essays see: *National Innovation Systems* edited by Richard Nelson (Oxford: Oxford University Press, 1993).

[26] On the basis of these assumptions we have various studies that try to define national systems of innovation and explain the industrial rise and fall of these countries using them. For example Christopher Freeman in his book *Technology, Policy, and Economic Performance: lessons from Japan* (1987) defines the Japanese national system of innovation in the beginning of his book and move on to use this definition to explain way Japan achieved industrial success vis a vis Europe and the US. Another example is William Lazonick's *Business Organization and the Myth of the Free Market* (1990), where he argues that the differences in industrial success of Japan, United State, and Britain are caused by the different national way that markets and firms are organized both internally and externally, i.e. the way that power relations, human capital, education and information flows, and decision making are constructed inside the firms, and the relations between these firms, and between firms and other institutions of the markets.

[27] Alexander Gerschenkron. *Economic Backwardness in Historical Perspective* (Cambridge: Harvard University Press, 1962).

[28] Richard Samuels merge these two approaches to analyze the particular Japanese system of utilizing military R&D paradigm of dual use technologies for the benefits of the whole Japanese industrial system in his book *Rich Nation Strong Army* (1994). Building on these but putting forward a more Polanyi (1946) based argument that any national economy is a manifestation of a specific society, John Zysamn argues that the national level is the correct level to understand innovation systems (1994).

[29] It is important to note that the methodological concern could be applied to all the others research approaches as well.

[30] Examples for that inability are the case of the Israeli system of innovation described and lamented on by Morris Teubal in the 1993 *National Innovation System*. Were he argues that the national and political official in Israel prove that they cannot handle any systematic innovational policy, and as such Israel has a very poor national system of innovation. Less then a year after this paper was written the national system of venture capital and industrial incubation for the high-tech sector, created by the Israeli government acted as the trigger of the following worldwide success of Israeli high-tech (the other contributing factors were the national system of innovation with the intimate historical and contemporary relations between the army, the universities and the business networks). These government actions are now viewed as the ideal type for state intervention aimed at spurring the creation of local private venture capital and had been since then phased out (The OECD Observer, 1998). Another example is the argument put forward in the same volume by Mowery and Rosenberg about the US system of innovation and the present and future-declining rule of start-ups and NTBFs (New Technology Based Firms) in the US system of innovations. For a more sectoral example we can look at the basic assumption behind Michael Cusumano's two books on the software development industry where in the first he claims that the Japanese software factory approach will win the day, while in the next the Microsoft approach is looked upon as the optimal. (*The Software Factory: Origins and Popularity in Japan*, 1988*; Microsoft's Secrets*, 1995).

[31] Two proponents of this argument are Herbert Kitschelt and Michael Porter. Kitschelt in his 1991 paper (*Industrial Governance Structures, Innovation Strategies, and the Case of Japan: Sectoral or Cross-national Comparative Analysis?)* argues that different industrial endowment of specific countries give them a specific edge in particular

industries that are based on particular level of technology. Porter in *The Competitive Advantage of Nations* (1990), present his diamond model of industrial clustering, which essentially contend that particular sets of factor inputs and industrial structure on the national level, either gives or prevent particular industries from having a competitive advantage in world markets.

[32] In *Regional Advantage*, AnnaLee Saxenian takes this stand when she analyzes the history of two of America's most successful high-tech clusters, Route 128 and Silicon Valley (1994). In her book she argues that the reasons that brought Silicon Valley in California to worldwide success, while Route 128 in Massachusetts suffered from prolonged relative decline since the 1980's, stems from different kind of regional system of innovation, which comes from different culture and corporate governance in the two regions. A somewhat similar logic, i.e. that the regional level is more illuminating then the national is brought forth by Richard Locke in *Remaking the Italian Economy* (1995).

[33] The logic of the sector as the unit of analysis stands behind some of the papers in J.R Hollingsworth, P. Schmitter, and W, Streek, Eds. *Governing Capitalist Economies: Performance and Control of Economic Sectors* (1994), another book that calls forth for this kind of research is Stephen Wilks and Maurice Wrights's *Comparative Government-Industry Relations* (1987).

[34] For a fuller description and definition of innovative collective learning see O'Sullivan, 1996, Teubal, *R&D and Technology Policy in NICs as Learning Processes*, World Development. 1996. Vol. 24. (3) 449-460, Tuebal and Justman (1996), Piore, Lester, and Malek (199X), Saxenian, (1994), and Piore and Sabel (1984).

[35] For example, the Katzir (Kachalsky) brothers who formulated the first S&T and industrial policy that reformulated the role the chief scientist, were the early leaders of "The Science Corps" in the early days of the IDF, professors in the Weitzmann institute, and prominent figures in REFAEL in its first years, the first Chief Scientist of the Ministry of Trade and Industry, Itzhak Yaakov (Yatza), last role before joining the civil service was the IDF's chief scientist and the head of the IDF's R&D unit at the time of Mamram's inception. (Yatza had a famous battle with the special Government Committee on Air-Conditioning, which couldn't understand why Yatza insist that a machine (i.e. the computer) needs air-conditioning, see Yediot Aheronot 5/4/2001).

[36] As late as 1968 the number of scientists and engineers in the Israeli industry was 3,400 or 1.3% of total employment. (Source: Ministry of Trade and Industry as reported in Teubal, (1993))

[37] "Spin-on" refers to the process in which technologies developed for the civilian market are transferred to the military market, in contrast to the "spin-off" process, in which technologies are transferred in the opposite direction. For more information about spin-on, spin-off, and dual-use technologies, see Samuels (1994).

[38] On the problem of collective action see Mancur Olson *The Logic of Collective Action* (Cambridge: Harvard university Press, 1965).

[39] As early as 1973 Pinchas Zussman , the than economic advisor to the Ministry of Defense, argued that the Mamram and the Computerization is the best example of the IDF critical role in the adaption and application of new technologies to create new products in the Israeli economy. (Zussman, Pinchas, and Tolkowski, Dan. 1973. *The Defense Establishment and its Contribution to Technological Progress*. Jerusalem: Van-Lir Institute (Hebrew)).

[40] On the importance and effects of research Paradigm see Richard, J. Samuels (1994), and Giovanni, Dosi, (1982)

[41] All the founders of the major professional computing schools in Israel (a major industry in of itself) are former instructors of the school, serve in the school on their reserve duty, recruit large quantity of their instructors from the school graduates, and base their all instructing approach of technology training around the school's approach.

[42] Interview with former official of The School.

[43] Source: *The Israeli Association of Software Houses*, followed with a telephone interview with Amnon Leibowitz of the IASH on the 12/22/99.

[44] Source: *US Statistical Abstract: The National Data Book*. 1999. (1984 PPI-CPI = 0.961, 1993 PPI-CPI = 0.692).

[45] See Mary O'Sullivan 1999 and Se'an O'riain 1997, and 1998. For a more comparative study see Jacob, J. Seid. Barriers to the Growth of the Hong Kong Software Industry. MIT: Master Thesis, 1998.

[46] Sources: O'Riain, Sean. 1997. Pp. 183, *The Israeli Association of Software Houses.*

[47] Source: NASDAQ news 11/22/1999, www.nasdaq.com.

[48] Source: *Israel's economic Overview.* 1998. Ministry of Trade and Industry, and the Giza Group: Israel Venture Capital survey, 01/15/2000. Even more astonishing is the fact that for Europe as a whole the whole VC capital in 1999 was only 10$ billions, including investments that would have not been considered VC if they were taken in Israel (Source: *The Economist*). It important to note that over 50% of Israeli start-up finance doesn't come from Israeli VC funds. Thus the 3$ billion raised by Israeli VCs in 2000 represent only a fraction of the future investment earmarked for Israeli companies.

[49] The Yozma initiative was planned specifically to create and spur a private professional VC industry in Israel, and indeed according to plan starting in 1995 all the Yozma funds were complately privatized. (Sources: Interviews with Yigal Erlich former Chief Scientists in MOTI and president and CEO of Yozma and the Israeli VC association, and Boaz Goldchmidt Executive VP of Yozma, Manual Trajtenberg, *R&D policy in Israel*. 2000).

[50] Araiv, G. Goodman, Seymour, E. 1994. "Israel: Of Swords and Software plowshares." *Communications of the ACM.* 37(6): 17-21.
[51] It should be noted, however, that the Golems were highly advanced worldwide. The building of first electronic computer in the US was finished in 1952, the first Golem, which was built on a similar scale to the first US computer was finished in 1954.
[52] In 1983 Elbit was one of the first Israeli firms to be listed on NASDAQ, three of Elbit companies, Elbit Medical imaging, Elbit Ltd., and Elbit Vision system limited are now listed on NADAQ.
[53] It is important to remember that the head of RAFAEL even after it was separated from the IDF was a member of the chief of stuff forum ("Forum MATCAL") the leading discussion and decision body in the Israeli security apparatus, a telling evidence to the importance given to indigenous R&D throughout the history of the Israeli state.
[54] Some of the data regarding these decisions is still classified, however, large parts of it are now published. For more details see the memoirs of Munya Mardor RAFAEL's first CEO, *RAFAEL: In the Paths of R&D for the Defense of Israel.* 1981, and *In the Beginning: 40 Years of Computing in Israel.* 1999.
[55] An interesting and in retrospect highly amusing, were the conclusions of the 1960 ministry of finance committee for the assessment of computer uses in the Israeli national economy. Headed by Professor Dostrovsky from the Weitzmann institute, the committee concluded that two Philco computers should be enough for the needs of the whole national economy in the years to come. These conclusions were sponsored by the ministry of defense that hoped that someone would buy a second Philco to be used as a backup to the one in Mamram. Fortunately enough no one paid any attention to these conclusions, not the least the committee's own secretary Emmanuel Sharon, which in the same time was busy organizing the computerization of the finance ministry.
[56] Emmanuel Sharon later finished his studies for a Ph.D. in economics in Berkeley, taught economics for few years in the Hebrew University, left to become the manager-director of the finance ministry, the CEO of CDC Israel, and the Chairperson of bank Hapoalim, the largest Israeli bank.
[57] E. Sharon and Y. Naamen. October, 1961. *Electronic Computers in Taxes and Administrative Services: A Report of a European Survey.* The State of Israel: the chamber of the state's revenues.
[58] The decision to buy an NCR was made after they correctly analyze the potential of a new information storage media, the hard disk. IBM in these years vehemently fought against this new innovation and declared it to be "inconsistent with the good working of a computer" (interview with E. Sharon 12/19/1999).
[59] Interview with Emmanuel Sharon, ibid.
[60] The first designated computer-science program in the universities opened in the Hebrew University as part of the mathematics institute only in 1969. Indeed by 1960 all the universities already had mainframe computers, and the Weitzmann institute was the first Israeli institution with an electronic computer as early as the 1950s. Nevertheless, none of the academic institutions in Israel neither taught programming and computer science as an academic discipline or thought about using these computer for anything else beside scientific calculations for some years after the IDF, the state bureaucracy, and some of the large public companies and banks were already training personal and using computers on a day to day basis. Another important effect of these early efforts was the accumulation and development of Assembler and machine language skills that requires a more intimate understanding of the working and capabilities of computers than the more modern programming languages.
[61] Due to the recent surge in the use of computers throughout the IDF, and the limited amount of programmers supplied by the school, several units in the IDF, mainly the intelligence and the air force, started to circumvent the official programmers recruiting paths and started to offer what they term "PC kids", i.e. high-schoolers with extensive knowledge of programming special shortcuts service of only three years, with almost no formal training.
[62] There is almost no correlation between being healthy and having a perfect health profile. Most of the population doesn't posses a perfect health profile, as for example if one needs glasses, suffer from any minor allergies, or was unfortunate enough to break too many bones in his body as a child; one's profile is no longer perfect.
[63] It is important to note that in the last two decades the requirements of the computer science department in the universities also ensure that only these who score exceptionally high on both the Israeli SAT and the national matriculation exams in the end of high-school are admitted, i.e. in both recruitment paths the IDF accept as programmers only these who belong to the national intellectual elite.
[64] On the important of a common language to product development and design process see the writings of Piore, Lester, Kofman, and Malek (1994) were they offer a new interpretive approach to product design.
[65] The School's website (Hebrew) is: http://www.idf.il/hebrew/organization/comp_school/info.stm
[66] Michael Porter, 1990. The *Competitive Advantage of Nations*. New York; Free Press. Pp. 81.
[67] It is a very interesting fact that 'improvisation' is a distinct and special word in Hebrew (Le-Alter) with ancient origin in Aramaic, with distinct modern meanings, and without any negative meanings whatsoever. Moreover, Ailtor (the capability to improvise) is considered as one of the most highly rated characteristics of the Israeli R&D personal, not

only as a commonsense knowledge of all Israelis, but as an important factor mentioned by all of my interviewees, and by the popular press.

[68] This experience is far from being ideal, however, a 23 years old that successfully managed with all its problems a mixed team of very young soldiers and few world-renowned professors in computer science and mathematics and professionals, with the average age of 45 and social network with her superiors, had probably gained invaluable managerial experience for the rest of her career, an experience that no private firm is able to offer.

[69] An interpreter is a programming language, which take the instructions that the programmer write and translating them into machine code that enable the computer to execute them.

[70] As attest by the high market value they have both in Israel and the US.

[71] For example in December 1999 John Bryce Training, one of the leading schools ran twenty six parallel year long professional diploma granting courses in system analyzing and designing, each of these usually has twenty to twenty five students.

[72] All of the private schools write original material. However, only their in-house instructors usually write it, and they are highly limited by profit consideration.

[73] As mentioned earlier the upgrading project was conducted on the two design courses, however as the first one is only a one-week introduction to the field of design, we will concentrate mainly on the second one.

[74] For an example of how Microsoft couldn't fix its software design issues and instead uses on the fly designing techniques see Michael Cusumano, *Microsoft's Secretes* (1995). For more information on product design see Michael Piore at el. *The Organization of Product Development* (1996).

[75] While I haven't dwell on these here this approach of letting and even forcing technical personal to take major responsibilities in complex projects is prominent throughout the IDF R&D units and other defense R&D organizations like REFAEL, as the logic of the IDF's Talpiot project aiming to create highly skilled R&D officers through special training of gifted 18 years old attest to.