World Journal of Science, Technology and Sustainable Development, Vol. 7, No. 1, 2010

THE NEED TO SCRUTINISE LATECOMER SOFTWARE DEVELOPMENT ACTIVITIES AND INVESTIGATE TECHNOLOGICAL CAPABILITIES

Rossitza Rousseva*

Delft University of Technology, The Netherlands

Abstract: Software development activities have been identified as a 'window of opportunity' for latecomer companies. Based on a critical literature review, this paper argues that studies are yet to scrutinise the exact nature and extent of the capabilities, which the latecomer companies have been able to develop. The main proposition advanced by this research is that the analyses need to investigate the technological capabilities, which the latecomer companies have been able to accumulate. This study outlines the specifics in analysing technological capabilities in latecomer software companies and improves our understanding about the complexity in developing software industries in latecomer context.

Keywords: software industry; latecomers; technological capabilities.

INTRODUCTION

In the last two decades a group of studies has been emphasising that the Information Technologies (ITs) open opportunities for leapfrogging by latecomer companies (Soete, 1985; Steinmueller, 2001). It has been observed that the availability of skilled human capital creates a base for development of IT industries by latecomer countries. The software sector is, in principle, a low-capital but knowledge and skill-intensive industry, and the international market for software is big and growing (OECD, 2004; Steinmueller, 2004). Due to their higher contribution to economic growth the development of software and other high-tech industries has the potential to foster economic development in latecomers (Kuznets, 1957).

A number of latecomers have attempted to develop software sectors in the last decade. Different countries followed different paths: development of the latecomer software sectors in some latecomers is foreign-led, in some it is indigenous-based; some latecomer software sectors are predominantly outsourcing-driven, some develop own software activities; some latecomer software sectors are export-driven, others remain domestic-oriented. The research is burgeoning following

*Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands; e-mail: r.i.rousseva@tudelft.nl

the recent expansion of software development activities undertaken by latecomers (Arora et al., 2001; Arora and Gambardella, 2005b; Carmel, 2003; Commander, 2005; Correa, 1996; Heeks and Nicholson, 2002; Tschang, 2003). Studies focus on different variables reflecting the development of latecomer software sectors, like the range of products and services offered, market orientation, models of development (e.g., outsourcing-driven, development of own products and services, domestic-oriented vs. export-driven, etc.), revenue, growth, skills and abilities, etc.

Despite the extensive studies on latecomer software development activities the focus has seldom been placed explicitly and systematically on the issue of technological capability, as the critical literature review in this paper reveals. The limited number of studies that did so have either not been well-recognised, or have some limitations. It is the aim of this paper to raise this issue and emphasise that if they are to assess the development of latecomer software industries, the studies need to investigate the technological capabilities, which the latecomer companies have been able to accumulate. This paper outlines the specifics in analysing technological capabilities in latecomer software industries.

The paper is structured as follows. Section 2 outlines the importance of capabilities as major drivers for development of latecomer software development activities. Section 3 gives an overview of the concept of technological capabilities building. Section 4 presents a critical review of the existing literature on capabilities in latecomer software industries. Section 5 describes the specifics in analysing technological capabilities in the latecomer software industries. Finally, Section 6 draws conclusions and outlines directions for further research.

CAPABILITIES: MAJOR DRIVERS FOR DEVELOPMENT OF LATECOMER SOFTWARE INDUSTRIES

The software sector is skill-intensive, and the availability of qualified technical personnel and capabilities for software engineering are critical for its development (Arora et al., 2001; Arora and Gambardella, 2005b; Athreye, 2005; Commander, 2005; Correa, 1996; Heeks, 1996; Heeks and Nicholson, 2002; Steinmueller, 2001). Critical importance is ascribed to computer engineering education. Availability of qualified human capital is, indisputably, a critical prerequisite for the development of the software sector.

Software production, by definition, is an innovative activity because it aims to produce new products or new ways of executing known tasks and functions (Torrisi, 1998). To undertake software activities, companies need the capabilities to innovate.[1] However, the development of innovation capabilities is neither automatic nor certain. The literature on technological capability reveals that innovation capabilities develop gradually towards the later stages of cumulative and gradual efforts aimed at increasing technological sophistication (Figueiredo, 2001). Applied to the software sector, this suggests that successful development of software activities requires accumulation of technological capabilities for software production in latecomer firms – this accumulation is not an automatic process activated merely by the presence of technically qualified human capital.[2] Therefore, the issue of technological capability development has to become a focal point both for companies engaged in software development activities, and countries that are aiming to develop their software sectors to harness their potential for fostering economic development.

Despite extensive study of software development activities in latecomers, there has

seldom been explicit and systematic focus on the issue of technological capability, as the literature review in this research reveals. The existing studies either focus on skills rather than capabilities or analyse the capability issue by focussing on certain capabilities without building a systematic framework for analysis of software technological capability and only one study develops a framework for software technological capability, which is valuable but has certain limitations. The current research places the issue of technological capabilities as the focal point of the analysis and develops a framework for analysing the development of technological capabilities in latecomer software companies. The following section gives a brief overview of the concept of the technological capabilities building. Section 4 provides a brief overview of the literature on capabilities in latecomer software sectors, followed by a section on the complexity of applying the concept of technological capability building to the case of the software sector.

BUILDING TECHNOLOGICAL CAPABILITIES: AN IMPERATIVE FOR LATECOMER COMPANIES

A number of scholars agree that, to be successful and sustainable over time, technological development in a latecomer or less-advanced context needs to involve technological capabilities building (Bell and Pavitt, 1993; Dutrenit, 2000; Figueiredo, 2001; Hobday, 1995a, 1995b; Hobday and Rush, 2007; Kim, 1997a, 1997b, 1998; Kim and Nelson, 2000; Lall, 1992; Marcelle, 2004; Rousseva, 2008; Scott-Kemmis and Chitravas, 2007). Technological capabilities building involve a deliberate process of learning and technology upgrading by the latecomer companies directed at the accumulation of knowledge and skills and their commercial application.

Technological capabilities can be defined as encompassing "the great variety of knowledge and skills which firms need so that they can acquire, assimilate, use, adapt, change and create technology" (Ernst et al., 1998, p.17). It should be noted that this is "a broad definition, which goes beyond engineering and technical know-how and includes organisational know-how" (Ernst et al., 1998, p.17).

A firm's technological capability is built upon multiple components that individually and collectively shape the rate and direction of the development of technological capability. Building technological capability is, in essence, a process of accumulation of a wide array of capabilities and subsequent deepening and broadening to achieve mastery over new technologies. While deepening the technological capabilities, the latecomer companies pass through subsequent stages of technological sophistication, which can be pictured as a 'technological ladder' (Hobday, 1995a, 1995b). In this sense, we can portray the process of technological capabilities building as a subsequent process of developing capabilities with a higher level of technological sophistication.

Research in technological capabilities building (Bell and Pavitt, 1993; Figueiredo, 2002; Hobday, 2000; Kim, 1997b) have emphasised that it is crucial to distinguish between production and innovation capabilities, as these reflect completely different sets of accumulated skills by the latecomer companies. Developing production capabilities involves accumulating skills and abilities to operate new technologies (e.g., to produce products and deliver services based on technologies produced elsewhere), while building innovation capabilities is a far more cumbersome task. To build innovation capabilities the latecomers need to deepen their knowledge and understanding

about the new technologies, to the extent that they will be able to change and modify the new technologies and, eventually, to introduce new technologies. In this sense, the innovation capabilities are qualitative different in nature as they signify that the latecomer companies have gone beyond the stage of replication and minor adaptations, and have become capable of introducing technological innovations on their own.

Innovation capabilities are difficult to develop and their development is far from automatic or certain (Bell and Pavitt, 1993; Dutrenit, 2000; Figueiredo, 2001; Hobday 1995a, 1995b; Hobday et al., 2004; Kim, 1997a, 1997b, 1998; Kim and Nelson, 2000; Lall, 1992; Marcelle, 2004; Rousseva, 2008; Tsekouras, 2006). Innovation capabilities develop on the basis of production capabilities, but go further as a result of active and purposeful accumulation of technological knowledge to go beyond production capacity and develop capabilities to change and modify technologies.

EXISTING LITERATURE ON CAPABILITIES IN LATECOMER SOFTWARE COMPANIES

The seminal works of Schware (1989, 1992), Correa (1996), and Heeks and Nicholson (2002) have outlined capabilities as a critical factor in enabling latecomer software companies to enter international markets. Some of the recent studies investigating the remarkable expansion of indigenous software development activities in a number of developing countries, like India, China and Brazil (see for example among many others, for all developing countries (Carmel, 2003; Arora and Gambardella, 2005b; Commander, 2005; Minevich and Richter, 2005), for India (Arora et al., 2001; Athreye 2005; Desai, 2005; Tschang, 2001), for China (Saxenian, 2005; Tschang and Xue, 2005), for China vs. India (Contractor, 2004; Tschang, 2003), for Brazil (Behrens, 2005; Botelho et al., 2005), etc) also have mentioned capabilities as an important driver in development of the latecomer software industries. However, most of these studies focus on software engineering capabilities without disentangling the capability issue. They are referencing capability but actually they analyse either skills for software engineering or for capabilities without placing them within the capability building framework. There are, however, several important exceptions, which are considered below.

In discussing the development of the Romanian software sector Grundey and Heeks (1998) employ a theoretical framework based on the concept of technological capabilities, and provide a taxonomy of software technological capability. This study is a valuable contribution, as it outlines different software production activities representing different stages on the technological ladder, which are required to perform more sophisticated software production. It outlines the activities underlying production and non-production software capabilities and provides a comprehensive analysis of the progression from simple software production, to software redesign, and skilled software production (Grundey and Heeks, 1998, p.11). It classifies software activities on seven levels: levels one and two include non-production activities, and level three and onwards outline the production activities.

This paper focuses only on software production activities in Grundey and Heeks's (1998) model. Level three represents production of copies of existing software products; level four includes adaptation, without production (e.g., creating a situation-specific application from a package); level five is simple software production (e.g.,

creating a new set of interfaces for users, creating a program to move data between applications, creating a small utility program, customising an existing program to user needs); level six involves software redesign (e.g., redesigning a program to meet local user needs, redesigning a program to meet regional/global user needs, minor process change such as modifying the software production process); level seven, the highest level in the classification, represents skilled software production: local product innovation (e.g., developing a new program to meet local user needs), international product innovation (i.e., developing a new program to meet regional/global user needs), major process change (i.e., redesigning the software production process), and process innovation (i.e., designing a completely new software production process).

Grundey and Heeks's (1998) model offers a comprehensive account of the wide variety of software activities. However, it includes both non-production and production activities. As the aim of the analysis of technological capabilities in this paper is to capture the level of technological sophistication of software production in companies, it is appropriate to focus on level five and upwards. These levels of the classification are incorporated into the framework for this research. The framework of technological capability building highlights that technological capabilities develop along the breadth and depth, i.e., they have different technological sophistication (e.g., routine, intermediate, advanced) and they incorporate a range of capabilities associated with each of these levels of technological sophistication (e.g., testing, engineering, design, R&D). Grundey and Heeks's model (from levels five onwards in particular) depicts the different levels of technological sophistication of the software technological capability that can be defined as a "software development technological ladder", but does not analyse the wide array of capabilities underlying these activities. In this sense, there is a need to explore the capabilities that allow latecomer companies to execute the activities referred to, and to build technological capability. In order to have more practical value, analyses should scrutinise the component elements of technological capabilities underlying the development of software technological capability. In Grundey and Heeks's study the theoretical framework is not tied directly to the empirical section, which explores, predominantly, the institutional foundations (and their transformation), and only briefly touches on the development of software activities in the latecomer software sector in Romania; thus, it does not provide a clear description of how to apply the proposed framework.

Tschang (2001) provides an extensive software development model. He employs a typology of software development activities in the software development lifecycle that corresponds to successive/different phases in the product lifecycle. The model outlines five major software development activities, which parallel four product development phases:

1 New product development phase

  1.1 *Conceptualisation*: requirements analysis, and design

  1.2 (*Initial*) *software engineering*: system analysis and software engineering, coding and programming, and testing

2 Installation phase

  2.1 *Customisation*

3 After sales phase

  3.1 *Maintenance*: operations and servicing

4 'Expiration' phase

4.1 Product code updating/versioning/ improvement.

Tschang focuses on the intersection between the software development lifecycle and the software product lifecycle. He also highlights that the identified activities in the software development cycle are associated with different value additions for the company. However, the model does not tackle the issue of the technological sophistication required for software development activities it identifies, neither does it investigate the links of these activities to outsourcing. Also, as the author acknowledges, further work is needed to break the model down into products and services, to determine different individual activities and skills needed for each type of activity and to distinguish different types of activities and firms.

Tschang identifies a set of skills that latecomer software companies need to develop (Tschang, 2001, pp.19–20). They are classified into two major groups: product development skills and business development skills. In Tschang's framework, there are four categories within product development skills:

1 basic technical skills such as coding and programming languages

2 system skills, including project management, requirements analysis and systems analysis

3 advanced or high technical skills, including mathematical abilities and other fundamental (scientific) knowledge used in science and innovative product development

4 innovative technical skills, which are the creative, interdisciplinary and other skills needed for new product innovation.

Under business development skills the author identifies two groups of skills:

- entrepreneurial skills, including various management and networking skills, e.g., sourcing of venture capital, managing a start up, forming alliances, etc.
- other conceptual skills, including new products requirements analysis, knowledge of market and customer needs, and innovative and creative abilities.

Tschang is helpful in identifying the capabilities that latecomer software companies need to muster and there are similarities and differences between his framework and the framework developed in this paper. The list of technical skills developed by Tschang provides a relevant account of the technical capabilities involved in software production but it is not specific in identifying advanced and innovation capabilities.

The list of business skills provided by Tschang (2001) is generic and does not take account of the specificity of technological development in a latecomer context, or the specific organisational capabilities mentioned by several studies of latecomer company development. From the technological and the organisational point of view, Tschang (2001) outlines skills rather than capabilities, an approach that has some conceptual limitations. This paper outlines the array of technological capabilities that reflect both the specifics of software production, and the works in the technological capability building and business literature. This study, therefore, attempts to provide more practically-oriented advice and a better representation of the array of capabilities needed by latecomer software companies in order to develop technological capability.

In a later study (Tschang, 2003), Tschang focuses explicitly on the capabilities of latecomer software sectors, examining the case of the Indian and Chinese sectors, and provides a list of items, namely, individual technical skills, process maturity, management

capability, technology, revenue model and product marketing capability. Despite being relevant to, and informative about, the state of development of a latecomer software sector, this list addresses individual technical capabilities and technology but does not distinguish between the technical capabilities needed for software production (e.g., capability for software engineering, design, etc.). It also leaves out a substantial number of important capabilities, e.g., capabilities required to monitor technological development and identify potential niches, capabilities for strategic thinking, linkage capabilities, capabilities to establish a dynamic organisational learning environment, etc. Further, from a technological capability point of view, this model provides a mix of capabilities (see above) and performance indicators (e.g., revenue model), without being exhaustive or clear about how these are derived (although most are indeed relevant).

In addition to Grundey and Heeks and Tschang, a fairly recent book by Arora and Gambardella (2005b) analyses the underpinnings of the successful development of the software sectors in several latecomer countries, among them India, China and Brazil. Alongside specific developments in individual countries, the study attempts to identify the driving forces of the development of latecomer software sectors. Capabilities emerge as important drivers underlying the success of these latecomers, as emphasised in the individual country chapters (see (Athreye, 2005)) in particular; also (Botelho et al., 2005; Tschang and Xue, 2005); and the conclusions in Arora and Gambardella (2005b).

Despite this recognition and highlighted importance, the studies in Arora and Gambardella (2005b) provide neither a detailed nor a unified framework for analysing capabilities (the study of Athreye analyses capabilities and is discussed below). In the individual chapters, the analysis of capabilities in Arora and Gambardella (2005b) is combined with many other factors affecting industry development, and it is the sources of the incubation of capabilities that are the focal points of the analysis, rather than the actual capabilities (with the notable exception of Athreye's contribution, which is discussed below). Similarly, despite emphasising the importance of firms' capabilities, the conclusions focus on the sources of firms' capabilities rather than on the capabilities themselves (Arora and Gambardella, 2005a). It is an advantage that this study has analysed the sources of capabilities, as it started unpacking the capability box. Nevertheless, an explicit framework considering the specifics of technological capabilities building in the software sector and a connection with the literature in the field of technological developments in latecomer contexts are both lacking in most of this study.

Within the collection edited by Arora and Gambardella (2005b) and Athreye (2005) devotes the greatest specific attention to capabilities development. Although it does not provide an analytical framework or systematically explore the issue of technological capabilities building, it does capture and portray the underlying idea of technological capability building in latecomer software companies. Exploring the development of the Indian software sector and the factors contributing to its successful development, Athreye (2005) observes that it is the evolutionary development of capabilities that underpins the Indian success. The study reveals that Indian companies entered the international market by providing basic programming skills, but that over time they developed capabilities for software process management and, in a few cases, expertise in specialised domains. Athreye concludes

by emphasising that the Indian model is a specific example; its success lies in the winning combination of developing different variants of the outsourced service model and evolving organisational capabilities for software process control and large-scale labour management (Athreye, 2005, p.36). In this sense, it can be perceived as a specific and exceptional case of a latecomer software sector development.

Athreye's focus on outsourcing of software products and services is just one of the paths open before the latecomer software companies, as a range of paths, including outsourcing or developing own products and services for domestic or international markets, are open to latecomers, as discussed below. In the case of Multinational Enterprise (MNE) outsourcing, capabilities building will be heavily influenced by learning spillovers from the MNE. Different paths may require different capabilities, which latecomer companies need to master, as this paper highlights. For example, outsourcing might require a set of skills that are limited and significantly narrower than the set of skills required for companies to produce their own products and services. In this sense, the question about technological capabilities in latecomer software companies is unresolved.

This critical review of the studies on capability building in latecomer software sectors highlights a major gap in our understanding about capability building in latecomer software sectors. Despite the recognition that capabilities are of critical importance for the development of latecomer software activities, a framework for analysing systematically the software technological capability and its component elements and the specifics of investigating software technological capability are still absent.

The following section attempts to fill the gaps and complete the research begun by the authors discussed above, with particular focus on the analysis of accumulation of software technological capabilities in the latecomers.

SPECIFICS IN ANALYSING TECHNOLOGICAL CAPABILITIES IN LATECOMER SOFTWARE INDUSTRIES

To have the capacity to investigate software technological capability the analysis has to incorporate the main ideas in the field of technological capability building.[3] Therefore, it has to investigate both the level of technological sophistication of innovation capabilities, which the latecomer companies have managed to develop, and the underlying capabilities. It also has to take into account the specifics of technological development in the latecomer context and the specifics in analysing the development of latecomer software development activities. This paper discusses the specifics of analysing technological capabilities in latecomer software industries: it focuses on the latter two points and touches upon the former two.

Exploring the technological capabilities in a latecomer software industry presents a challenge. So far, studies analysing the process of technological capabilities have been predominantly focused on the industrial sectors, studying the development of the electronics industry (Gee and Kuo, 1998; Hobday, 1995b; Kim, 1997b; Mytelka and Ernst, 1998), textiles (Gee and Kuo, 1998; Lall, 1987), steel industry (Figueiredo, 2001; Lall, 1987), telecommunications (Marcelle, 2004), and so forth. As the predominant part of the studies have been directed at exploring technological capabilities in industrial sectors, the analytical framework developed in the field so far reflects the specifics of the industrial sector as contrasted with the service sector and, additionally, specific features of industrial activities such

as photolithography in the context of electronic integrated circuits. A study investigating technological capabilities in a latecomer software industry needs to take into account the specifics of the software industry, which are discussed below.

This research identifies two major features that are particularly relevant in analysing the accumulation of technological capabilities in latecomer software companies. These are, in particular,

1 degree of innovativeness
2 breadth and depth of technological capabilities.

First, the degree of innovativeness inbuilt in software technological capability may vary. As noted earlier software production is inherently an innovation activity (Torrisi, 1998). However, it must be underlined that the degree of innovativeness and the significance of novelty, which governs the *extent* of innovation capability needed, varies among different software projects, and this holds both for advanced and latecomer software companies. Software services involve certain innovative components, as they include innovativeness of design arising from the unique qualities of every software 'expression'. But even within software services the degree of innovativeness varies. Software services, such as re-coding legacy applications into more modern computer languages, data migration, or resolution of specific incompatibilities among similar systems, for example, involve a relatively small innovative component compared to software services associated with redesign. Creating software products involves an even greater, and often more significant innovative, component than software services, as it is associated with creating 'best of breed' software products and is comparable to frontier technological developments in that domain, which requires a high level of innovative capability. Therefore, development of software products represents the highest level of technological sophistication, while software services involve innovative component, which is lesser compared to software products and varies further according to the nature of software services. Hence it is important to consider the degree of innovativeness inbuilt in software production.

Before differentiating the degree of innovativeness in latecomer software production further, we should clarify what we mean by innovativeness. Innovativeness is usually measured with reference to novelty in the world market. Based on this logic, only products that are successful in international markets have high levels of innovativeness. However, a latecomer company may develop a product in the domestic market that targets local customers' needs and is innovative. Furthermore, innovativeness is also associated with the commercial value produced. It is generally believed that commercialisation in the international market has the potential to reap higher profits than commercialisation in the domestic market. But this may not always hold. For example, a company may attain greater commercial value by creating an innovative product that meets the needs of a large number of customers in the domestic market (e.g., payroll and tax record keeping systems reflecting local regulations) compared with a company that develops a niche product for a limited number of customers in the international markets. This duality creates problems in assessing innovativeness inbuilt in the software production. As the latecomer markets are usually less sophisticated compared to the international markets (unless the former are dominated by MNEs), we can assume that products and services offered by latecomer software companies in the domestic market are with less technologically sophisticated compared to products and services offered by latecomer

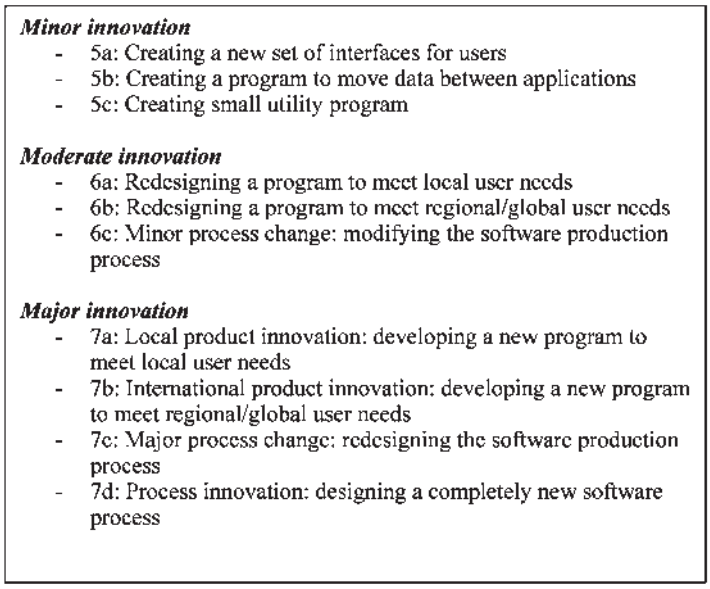

Figure 1 Software technological capability – software development technological ladder
Source: Adapted from Grundey and Heeks (1998)

software companies in the international markets; nevertheless, the analysis need to investigate the level of technological sophistication and the degree of innovativeness of the products and services offered in the domestic market.

In analysing the software activities undertaken by latecomer companies it is important to consider the degree of innovativeness they encompass and the degree of sophistication of innovation capabilities deployed. When studying the degree of innovative efforts associated with producing particular software products or services, it is necessary to distinguish between minor, moderate and major innovation, which, respectively, are associated with the capabilities for minor, moderate and major innovation. This can be viewed as the "software development technological ladder".

This distinction follows the classification of the software technological capability developed by Grundey and Heeks (1998) and corresponds to the activities classified in levels five to seven in their classification (Figure 1 in the Appendix). Simple software production, i.e., software activities such as creating a new set of interfaces, data migration, creating small utility programs and/or modifying existing programs to meet user needs, involves a small innovative component and signals the existence of capabilities for minor innovation. Software redesign activities, such as redesigning a program to meet local user or regional/global user needs (i.e., customisation and/or localisation), and minor process change (i.e., modifying the software production process), demonstrates the indicate capabilities for moderate innovation. Skilled software production activities, such as local product innovation (i.e., developing a new program to meet local user needs), international product innovation (i.e., developing a new program to meet regional/global user needs), major process change (i.e., redesigning the software

production process), and process innovation (i.e., designing a completely new software production process) suggest the capabilities for major innovation. The classification of the technological sophistication of the software development activities in Figure 1, i.e., the software technological capability or the software development technological ladder, in minor, moderate and major categories reflects the basic, intermediate and advanced categories in the framework of technological capability.

A proportion of activities such as re-coding, data migration, resolving incompatibility, etc., can be expected to account for a significant share of the software services offered by latecomer companies. On the other hand, the presence of more innovative activities, such as the creation of packages or sophisticated customised services, despite their small share in latecomer software developments, signals the existence of potentially significant innovation capabilities in latecomers. For example, if many latecomer software development activities are directed at offering services in the domestic markets and there is also a growing share of outsourced services for international markets, this indicates the existence of capabilities for minor and, eventually, moderate innovation; India has specialised in offering software services in the international market and, if we apply the classification of the degree of innovativeness to the range of software development activities that Indian software companies offer according to the literature (Arora et al., 2001; Athreye, 2005; Desai, 2005), this reveals the existence of capabilities for minor, moderate and, in a limited number of cases, major innovation. China and Brazil have developed software products and services for their domestic markets (Behrens, 2005; Botelho et al., 2005; Saxenian and Quan, 2005; Tschang and Xue, 2005), which suggest capabilities for moderate and major innovation. Further, in-depth case study and comparison-based analyses are needed to reveal the achievements and problems involved in the development of technological capabilities in the latecomers. These should be done by comparing companies within a single country and comparing companies in different countries, to capture company – and context-specific issues.

The second specific for analysing technological capabilities in latecomer software companies, concerns the breadth and depth of capabilities. The discussion about degree of innovativeness inbuilt into software technological capability implies breadth and depth of the technological capability. Capabilities develop sequentially and higher technological sophistication usually entails deeper (i.e., more sophisticated) capabilities and a wider range of (i.e., broader) capabilities, as highlighted in technological capability building literature.

Provided that the nature of innovation is similar across sectors, the capabilities literature suggests that major innovation requires the execution of a greater variety of, and also more complex, software development activities compared to the capabilities required for moderate and minor innovation. For example, creating a product innovation involves broader and far more complex capabilities than software redesign or simple software production. Similarly, capabilities for moderate innovation entail a wider variety and more complex software activities compared to the capabilities required for minor innovation. Thus, software redesign (e.g., redesigning a program to meet local or global user needs or minor process change), for example, requires greater capabilities than the simple software production of new interfaces, small utility programs or programs for data migration, or modifications to existing

programs, etc. Therefore, a higher degree of innovativeness entails broader and more complex capabilities, i.e., the breadth and depth of capabilities increases with innovativeness.

CONCLUSIONS AND DIRECTIONS FOR FURTHER RESEARCH

Despite the recent burgeoning research on development of latecomer software industries, critical questions remain unanswered. Based on a critical literature review, this paper argues that studies are yet to scrutinise the exact nature and extent of the capabilities which the latecomer companies have been able to develop. The main proposition advanced by this research is that if they are to assess the latecomer software development activities, the studies need to investigate the technological capabilities which the latecomer companies have been able to accumulate. This study outlines the specifics in analysing technological capabilities in latecomer software industries and thus improves our understanding about the complexity in developing software industries in latecomer context. The current enquiry lays the foundations of the analysis of technological capabilities in latecomer software industries and it will be further coupled with a separate paper disentangling a wide array of capabilities, which the latecomer software companies need to muster to be able to develop software activities based on indigenous resources (Rousseva, 2007). These two papers provide examples to support the ideas they develop. Nevertheless, further in-depth, case study and comparison-based analyses are needed to reveal the achievements and problems in the development of technological capabilities in latecomer software companies. These should be done by comparing different companies in one country and comparing companies in different countries to capture both company– and context-specific issues.

ACKNOWLEDGEMENTS

This research is the result of a research project 'TechCapaBuild' supported by the Marie Curie Individual Fellowship within the 6th Framework Programme of the EU. I express my gratitude to Ed Steinmueller and an anonymous referee for their comments on an earlier version of this paper. I am solely responsible for omissions and the usual disclaimer applies.

BIOGRAPHY

Rossitza Rousseva is an Assistant Professor in the Delft University of Technology, The Netherlands, and was a researcher at UNU MERIT. She has a background in technology and innovation management, and science and technology policy. Her areas of expertise are innovation management, knowledge management, capability development, strategic intent, ICT sector, with special focus on EU and new member states, and emerging economies. Her research specialisation is in development of software industries with special focus on capability accumulation and innovation activities in latecomer software companies.

REFERENCES

Arora, A., Arunachalam, V., Asundi, J. and Fernandes, R. (2001) 'The Indian software services industry', *Research Policy*, Vol. 30, No. 8, pp.1267–1287.

Arora, A. and Gambardella, A. (2005a) *From Underdogs to Tigers. The Rise and Growth of the Software Industry in Brazil, China, India, Ireland and Israel*, Oxford University Press, Oxford.

Arora, A. and Gambardella, A.e. (2005b) *From Underdogs to Tigers. The Rise and Growth*

of the Software Industry in Brazil, China, India, Ireland and Israel, Oxford University Press, Oxford.

Athreye, S. (2005) 'The Indian software industry', in Arora, A. and Gambardella, A. (Eds.): *From Underdogs to Tigers. The Rise and Growth of the Software Industry in Brazil, China, India, Ireland and Israel*, Oxford University Press, Oxford, pp.7–40.

Behrens, A. (2005) 'Brazil', in Commander, S.E. (Ed.): *The Software Industry in Emerging Markets*, Edward Elgar, Cheltenham, pp.189–220.

Bell, M. and Pavitt, K. (1993) 'Technological accumulation and industrial growth: contrasts between developed and developing countries', *Industrial and Corporate Change*, Vol. 2, No. 2, pp.157–210.

Botelho, A.J., Stefanuto, G. and Veloso, F. (2005) 'The Brazilian software industry', in Arora, A. and Gambardella, A.e. (Eds.): *From Underdogs to Tigers. The Rise and Growth of the Software Industry in Brazil, China, India, Ireland and Israel*, Oxford University Press, Oxford, pp.99–130.

Carmel, E. (2003) 'Taxonomy of new software exporting nations', *Electronic Journal on Information Systems in Developing Countries*, Vol. 13, No. 2, pp.1–6.

Commander, S.e. (2005) *The Software Industry in Emerging Markets*, Edward Elgar, Cheltenham.

Contractor, F.a.K.S. (2004) 'The role of export-driven entrepreneurship in economic development: a comparison of software exports from India, China and Taiwan', *Technological Change and Forecasting*, Vol. 71, No. 8, pp.799–822.

Correa, C. (1996) 'Strategies for software exports from developing countries', *World Development*, Vol. 24, No. 1, pp.171–182.

Desai, A. (2005) 'India', in Commander, S.E. (Ed.): *The Software Industry in Emerging Markets*, Edward Elgar, Cheltenham, pp.32–72.

Dutrenit, G. (2000) *Learning and Knowledge Management in the Firm. From Knowledge Accumulation to Strategic Capabilities*, Edward Elgar, Cheltenham.

Ernst, D., Ganiatsos, T. and Mytelka, L. (1998) *Technological Capabilities and Export Success in Asia*, London, Routledge.

Figueiredo, P. (2001) *Technological Learning and Competitive Performance*, Edward Elgar, Cheltenham.

Figueiredo, P.N. (2002) 'Does technological learning pay off? Inter-firm differences in technological capability-accumulation paths and operational performance improvement', *Research Policy*, Vol. 31, No. 1, pp.73–94.

Gee, S. and Kuo, W-J. (1998) 'Export success and technological capability. Textiles and electronics in Taiwan province of China', in Ernst, D., Ganiatsos, T. and Mytelka, L. (Eds.): *Technological Capabilities and Export Success in Asia*, Routledge, London, pp.46–86.

Grundey, M. and Heeks, R. (1998) *Romania's Hardware and Software Industry: Building IT Policy and Capabilities in a Transition Economy*, Development Informatics Working Paper Series, University of Manchester.

Heeks, R. (1996) *India's Software Industry. State Policy, Liberalisation and Industrial Development*, SAGE Publishers, Thousand Oaks.

Heeks, R. and Nicholson, B. (2002) *Software Export Success Factors Strategies in Developing and Transition Economies*, Development Informatics Working Paper Series, Manchester.

Hobday, M. (1995a) 'East Asian latecomer firms: learning the technology of electronics', *World Development*, Vol. 23, No. 7, pp.1172–1193.

Hobday, M. (1995b) *Innovation in East Asia: The Challenge to Japan*, Edward Elgar, Cheltenham.

Hobday, M. (2000) *East vs. Southeast Innovation Systems: Comparing OEM- and TNC-led Growth in Electronics*, Cambridge University Press, Cambridge.

Hobday, M., Rush, H. and Bessant, J. (2004) 'Approaching the innovation frontier in Korea: the transition phase to leadership', *Research Policy*, Vol. 33, pp.1433–1457.

Hobday, M. and Rush, H. (2007) 'Upgrading the technological capabilities of foreign

transnational subsidiaries in developing countries: the case of electronics in Thailand', *Research Policy*, Vol. 36, No. 9, pp.1335–1356.

Kim, L. (1997a) 'The dynamics of samsung's technological learning in semiconductors', *California Management Review*, Vol. 39, No. 3, pp.86–100.

Kim, L. (1997b) *Imitation to Innovation: The Dynamics of Korea's Technological Learning*, Harvard Business School Press, Boston.

Kim, L. (1998) 'Crisis construction and organisational learning: capability building in catching up in Hyunday Motor', *Organisation Science*, Vol. 9, No. 4, pp.506–521.

Kim, L. and Nelson, R.e. (2000) *Technology, Learning and Innovation. Experiences of Newly Industrialized Economies*, Cambridge University Press, Cambridge.

Kuznets, S. (1957) *Modern Economic Growth: Rate, Structure and Spread*, Yale University Press, New Haven.

Lall, S. (1987) *Learning to Industrialize. The Acquisition of Technological Capability by India*, Macmillan Press, London.

Lall, S. (1992) Technological capabilities and industrialization', *World Development*, Vol. 20, No. 2, pp.165–186.

Marcelle, G. (2004) *Technological Learning. A Strategic Imperative for Firms in the Developing World*, Edward Elgar, Cheltenham.

Minevich, M. and Richter, F-J. (2005) *Global Outsourcing Report 2005*, Going Global Ventures and HORASIS, New York and Geneva.

Mytelka, L. and Ernst, D. (1998) 'Catching up, keeping up and getting ahead. The Korean model under pressure', in Ernst, D., Ganiatsos, T. and Mytelka, L. (Eds.): *Technological Capabilities and Export Success in Asia*, Routledge, London, pp.325–336.

OECD (2004) *Information Technology Outlook*, OECD, Paris.

Rousseva, R. (2007) *Approach For Analysing Capabilities in Latecomer Software Companies*, UNU MERIT Working Papers (2007-035, November 2007), p.60.

Rousseva, R. (2008) 'Identifying technological capabilities with different degree of coherence: the challenge to achieve high technological sophistication in latecomer software companies (based on the Bulgarian case)', *Technological Forecasting and Social Change*, Vol. 75, No. 7, pp.1007–1031.

Saxenian, A. and Quan, X. (2005) 'China', in Commander, S. (Ed.): *The Software Industry in Emerging Markets*, Edward Elgar, Cheltenham, pp.73–132.

Saxenian, A.a.Q.X. (2005) 'China', in Commander, S.E. (Ed.): *The Software Industry in Emerging Markets*, Edward Elgar, Cheltenham, pp.73–132.

Schware, R. (1989) *The World Software Industry and Software Engineering. Opportunities and Constraints for Newly Industrialized Economies*, World Bank Technical Paper No. 35/1989, Washington.

Schware, R. (1992) 'Software industry entry strategies for developing countries: a walking on two legs proposition', *World Development*, Vol. 20, No. 2, pp.143–156.

Scott-Kemmis, D. and Chitravas, C. (2007) 'Revisiting the learning and capability concepts –building learning systems in thai auto component firms', *Asian Journal of Technology Innovation*, Vol. 15, No. 2, pp.67–84.

Soete, L. (1985) 'International diffusion of technology, industrial development and technological leapfrogging', *World Development*, Vol. 13, No. 3, pp.409–422.

Steinmueller, E. (2001) 'ICTs and the possibilities for leapfrogging by developing countries', *International Labour Review*, Vol. 140, No. 2, pp.193–210.

Steinmueller, E. (2004) 'The European software sectoral system of innovation', in Malerba, F.E. (Ed.): *Sectoral Systems of Innovation: Concepts, Issues and Analyses of Six Major Sectors in Europe*, Cambridge University Press, Cambridge, pp.193–242.

Torrisi, S. (1998) *Industrial Organisation and Innovation. An International Study of the Software Industry*, Edward Elgar, Cheltenham.

Tschang, T. (2001) *The Basic Characteristics of Skills and Organisational Capabilities in the Indian Software Industry*, ADB Institute Working Paper Series No. 23/2001, Singapore.

Tschang, T. (2003) *China's Software Industry and Its Implications for India*, OECD Working Paper Series, No. 45/2003, Paris.

Tschang, T. and Xue, L. (2005) 'The Chinese software industry', in Arora, A. and Gambardella, A.E. (Eds.): *From Underdogs to Tigers. The Rise and Growth of the Software Industry in Brazil, China, India, Ireland and Israel*, Oxford University Press, Oxford, pp.131–170.

Tsekouras, G. (2006) 'Gaining competitive advantage through knowledge integration in a European industrialising economy', *Int. J. Technology Management*, Vol. 36, Nos. 1–3, pp.126–147.

NOTES

[1] It should be noted that the degree of innovativeness inbuilt in software production varies and this point is explicitly discussed in outlining the specifics involved in analysing software technological capability (Section 5).

[2] Studies of technological development in latecomers reveal that to be able to develop innovation capabilities the latecomers have to engage in a deliberate effort of technological capability building. It is not impossible that a brilliant software solution might be developed by a 'lone inventor', but this is more likely in advanced-context companies than latecomers. Studies on technological capabilities in latecomer software companies are limited, and no such cases have been identified.

[3] The current enquiry adopts the following main propositions from the literature on technological capability building:

- latecomer technological development has specific features
- every sector has sector-specific features and trajectories of technological development
- technological capability is comprised of a wide array of component capabilities and expertise
- technological capabilities develop gradually by passing through subsequent stages of increase in technological sophistication of the accumulated capabilities
- analyses have to investigate both the level of technological sophistication of the technological capabilities (e.g., basic, intermediate, advanced), and the underlying component elements
- the accumulation of technological capability requires accumulation of technological and organisational capabilities
- learning and capability development efforts in the company are a major driver for innovation and technological upgrade.