Chapter 2

LITERATURE REVIEW
2.1 Introduction

This chapter consists of 8 Sections. The structure of this chapter is presented in Section 2.1 and Section 2.2 presents definitions and evolution of Global Software Development (GSD) as a practice. Section 2.3 presents a Literature concept map, giving a pictorial view of the topics reviewed and the flow. Section 2.4 presents business drivers and benefits and Section 2.5 discusses the success factors of GSD and issues in realization of those success factors. Section 2.6 cites literature evidence on some of the Product Transfer strategies that alleviate GSD’s implementation issues. Section 2.7 cites the literature that talk on relationship across Product life cycle, GSD, Customer satisfaction and Sales enablement. Section 2.8 summarizes the chapter. Figure 2.1 presents the structure of this chapter.

![Figure 2.1. Structure of Chapter 2](image)

In this study, the objective of literature review is twofold: 1) to deep dive and understand GSD as a concept, practice and its benefits, success factors and challenges
and 2) to delineate key constructs to comprehend the transfer strategies and impacts of GSD.

2.2 GSD and Offshoring – Evolution and Definition

Globalization of IT industry has led to software development work being spread to various geographies including developing nations. Factors derived from various forces including ever changing customer tastes and preferences, features and characteristics of the products, business objectives, and changes of methodologies in software engineering and development are the drivers for globalization. With software products having evolved as modular components, it became a natural extension that the development work could more easily be distributed across various sites. This had motivated the software companies to go for outsourcing or subcontracting the work partially or fully as well as offshoring to their own subsidiaries in other countries. Lewin et al (2006) explains that the causes or triggers to the adoption of IT Offshoring are many and the primary one of them is the advances in internet and broadband technologies. They list that the Offshoring process grew rapidly in the IT/Software industry since the 1980s due to factors including higher demand for IT as whole due to factors like opportunities in Y2K projects, higher supply of manpower due to low cost, innovations in IT industry, increased investments in infrastructure and emergence of skilled labour in developing countries. Factors like huge supply of technically skilled manpower from developing and low cost countries like India, their liberalisation and improved business climate and the standardization of communication protocols, the increased interoperability of IT systems compounded and encouraged popular adoption of offshoring practise by IT industry globally.

As Edwards and Sridhar (2005), explain, business processes involving offshoring would be similar to a virtual project management environment in which the teams that
are located closer to a customer or the headquarters of the company would exchange information related to software development specifications back and forth with the remotely located software development facility at offshore locations. The practise involves globally distributed virtual teams that collaborate across geographies over multiple time zones and is known by multiple terms including Global Software Development (GSD), Offshoring and Distributed Software development among many others. Throughout this thesis, the term of GSD is commonly used in this context.

Davis et al (2006) define Offshoring as “Provision of organizational products and services from locations in other countries”. As per Aspray, Mayadas and Vardi (2006), there are about six major types of activities that are typically offshored in the field of Information Technology, in general.

- IT-Enabled Services (ITES) and Business Process Outsourcing (BPO)
- Telesales and telemarketing
- Manufacturing of parts or assemblies of electronic instruments including computers
- Consulting and management roles in IT including design, architecture and business strategy
- Software Programming, testing, and Support and
- Research and development in IT

As far as this study is concerned, the last three categories come close to the scope involving offshoring of software product development.

Literature cites many definitions of GSD. “Software development that is geographically, remotely or globally distributed with the aim of rationalizing the development process and products”, as Avram (2007) had defined. As per Sangwan, Bass,
Mullick, Paulish, D. J and Kazmeier (2006), GSD is defined as “development of software using teams from different geographic locations within a country or across countries”.

As per Steven and Anil (2006), Outsourcing would typically mean that the software development resources would be owned by a third party, while insourcing would mean they are owned by the manufacturer of the software. Their study explains the following four modes of GSD by combining these two concepts of outsourcing and insourcing together.

1) Captive Offshoring: This mode is used when companies choose a location in a region that it had not done business before and establishes the operation as a legal entity and hires local staff to perform.

2) Global Outsourcing: This does not require the company the commitment to establish a captive centre but can contract with a service provider and get the development done.

3) Local Outsourcing: This method is employed to gain access to specialized skills or to meet temporary requirements. Contractors are hired on-site.

4) Centralized development: This is the traditional way of software product development where all the concerned resources are based at one-site, typically at the headquarters and are employees of the company.

The SITO Model - Sourcing of IT work Offshore, put together by Carmel et al (2006) discusses four stages of offshore IT sourcing:

- First stage: Offshore Bystander – Only domestic sourcing and no offshoring.

- Second stage: Offshore Experimenter – On an ad-hoc basis, initial experiments is done to try offshoring.

- Third stage: Proactive Cost Focus – Work that is add-on types and less critical to the core of the project are performed at offshore development centres and
gradually the mechanisms of management at offshores emerge. Cost cutting is the primary focus here.

- Fourth stage: Proactive Strategic Focus – The goal of offshoring is strategic and to gain competitive advantage. The offshore management process gains maturity. Core part of the work is offshored too.

As detailed by Contractor, Kumar, Kundu, and Pedersen (2010), “Outsourcing is a conscious subcontracting of part of the software development projects to contractors or vendors either within the same country or in overseas. This is different from offshoring in which part of the operations are moved necessarily to another country, either to the company’s own subsidiary set-up or to a vendor based abroad.”
Chakrabarty (2007) depicted this in a graphical representation as reproduced from his study shown in figure-2.2.

Carmel et al (2006) present that GSD is synonymous with ‘Offshore Sourcing’ as it would include outsourcing to an offshore third party service provider or a vendor in addition to insourcing within a global corporation. An insourcing facility is an overseas extension of the enterprise.

As per Rönkkö, Eloranta, Mustaniemi, Mutanen, and Kontio (2007), Distributed Software Development or GSD, as a practise is typically practised through two types of strategies - Offshore Insourcing and Offshore Outsourcing; this study would focus on offshore insourcing which is getting the work done by a wholly owned subsidiary of an MNC in a foreign country.

The following sections would quote the term GSD commonly in the context of Global Software product development which is synonymous with international product transfer and Off-shore insourcing or in general, Offshoring of Software products.

2.3 Literature Concept Map

The literature concept map shown in figure-2.3 depicts the structured approach followed in reviewing the literature and presents the sub-topics under which the literature was reviewed in this chapter. Literature on GSD’s definitions, history and evolution are reviewed first, followed by those that discuss the drivers, benefits, success factors, issues and challenges in the practice of GSD. Studies that review some of the strategies of GSD their influence on the issues of GSD as well as the product strategy parameters and also the two key types of software deployment are reviewed next. The Conceptual framework and hypotheses are presented in chapter 3.
2.4 The Business drivers and benefits of GSD

As listed by Stephan et al (2008), while the popular drivers of GSD were reduced development costs and access to skilled labour, other strategic drivers include (figure 2.4) ability to redesign business processes, pressure from competitive landscape, need to improve the service quality and strategy for business growth, while discussing on the survey findings of Offshoring Research Network (ORN).

Key factors and drivers for increased adoption of GSD by global software development companies, as per Herbsleb and Moitra (2001) are listed:

- Pressure to improve time-to-market
- Closer access to expanding and new markets
- Usage of skilled resources irrespective of their location
While the known benefits of GSDs are access to skilled manpower and savings in software development cost due to reduced cost of labour, as per Ågerfalk et al (2008), the benefits include other well-known benefits like easier access to new customers and markets and accelerated go-to-market possibility; internally, the benefits encompass at organizational, team and at process levels. The organizational level benefits include innovation, shared business practices, improved resource allocation realized by GSD. Benefits like improved modularization of tasks and team autonomy are realized at team and process levels.

Although cost savings and access to skilled manpower are the popular drivers, factors that expand the revenue base like competitive pressure and access to new markets also do figure in (Table 2.1) as strategic drivers of GSD as per a survey done by ORN cited by Lewin et al (2006).

---

![Percentage of Survey Responses Rating Driver as “Important” or “Very Important Across Functions](image-url)

Table 2.1 Strategic Drivers of Offshoring

<table>
<thead>
<tr>
<th>Strategic Drivers</th>
<th>% of respondents citing driver as important</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taking out cost</td>
<td>93%</td>
</tr>
<tr>
<td>Competitive pressure</td>
<td>69%</td>
</tr>
<tr>
<td>Improving service levels</td>
<td>56%</td>
</tr>
<tr>
<td>Accessing qualified personnel</td>
<td>55%</td>
</tr>
<tr>
<td>Changing rules of the game</td>
<td>41%</td>
</tr>
<tr>
<td>Industry practice</td>
<td>37%</td>
</tr>
<tr>
<td>Business process redesign</td>
<td>35%</td>
</tr>
<tr>
<td>Access to new markets</td>
<td>33%</td>
</tr>
<tr>
<td>Enhancing system redundancy</td>
<td>27%</td>
</tr>
</tbody>
</table>

% of respondents who answered 4 or 5, on 5 points Likert scale, to proposed strategic drivers of offshoring.

[Source: Lewin and Peeters, 2006]

The Product life cycle theory referred by Kotler et al (1990) associates the sales performance of products against time; with some of the strategic sales drivers listed in the previous sections such as speed to market, growth strategy and access to new markets directly related to Offshoring, one could associate GSD’s successful implementation to drivers of product life cycle.

2.5 Key Success Factors, issues and challenges of GSD

Although the practice of GSD projects these benefits, the extant literature also records the degree to which these benefits were realized. A multi-case study done by Conchúir et al (2006) at three MNCs that have extensive experience in GSD concludes that the benefits are neither clear-cut nor could be taken for granted. The study posits that the benefit of reduced development cost is only partially realized owing to high overhead in communication and coordination efforts. Also, it was noted that leveraging time-zone effectiveness did not find takers for development roles but only for roles like testing.
The systematic literature review undertaken by Šmite et al (2010) records that more than half of the reported cases in literature on GSD have reported problems; only about 35% of them reported successful practices or are success stories. As per a 2005 report by Deloitte Consulting, close to two-thirds of the companies that had originally resorted to outsourcing, had at one point, reversed their decision and insourced the work, since the originally projected advantages of outsourcing did not materialize (Deloitte Consulting, 2005).

As per Conchúir et al (2006), even the popularly projected benefits like reduced cost of development or leveraging different time zones have only been partially realized due to various challenges in implementing GSD.

As per a systematic literature review done by Sudhakar (2013), few of the top critical success factors for GSD are:

- Cultural factors
- Language factors
- Time zone factors
- Organizational factors
- Communication factors
- Knowledge transfer factors
- Project execution related factors and
- Team factors

In a case study done by Prikladnicki, Audy and Evaristo (2004) at a Brazilian Software Development Unit of a Multinational organization, the following critical success factors got listed in an effort to identify factors that enable successful operation in a GSD set up.

- Development process
- Team work
- Communication efficiency
- Knowledge management and training

In a combined interview and literature review study done by Lamersdorf and Münch (2010), communication, coordination, knowledge management and process maturity were identified as some of the key factors among others that influence GSD’s intended goals. Similarly, in a case study done by Damian and Zowghi (2002) focusing on Requirements Engineering function, Communication efficiency, Time zone difference and Knowledge management issues were listed as important challenges of GSD in addition to a few others.

The major issues and challenges witnessed when GSD is rolled out as cited by various literature are very much in line with the above mentioned success factors. Herbsleb et al (2001) list some of the major dimensions of problems in GSD:

- Strategic issues like architecturally disjoint projects
- Inadequate communication
- Knowledge management
- Process and Project management issues

A case study done by Herbsleb, Paulish and Bass (2005) involving software development at nine different locations of Siemens listed communication, process maturity and domain knowledge as the causes that created various issues among few others. Similarly, in a large scale survey done by Begel et al (2008) with engineers at Microsoft Corporation it was found that communication and coordination reported the highest issues in GSD.
From these literature, it could be summarized that there emerges four key challenges or success factors for smooth enablement of GSD, namely:

1. Coordination
2. Communication
3. Process Maturity
4. Knowledge Management

A gap witnessed in most of the studies is that the GSD issues that are unique to software product development were not explicitly identified as compared to software services development.

2.6 Strategies to alleviate issues & challenges of GSD

GSD is a reality. In their large scale survey done at Microsoft Corporation, Begel et al (2008) observed that more than 50% of the respondents regularly collaborate with global stakeholders that are more than three time zones away. Studies have recommended solutions to address some of the issues in GSD implementation discussed above and pave way for GSD’s success.

There have been some case studies and published literature that analyse when, why, what, where and how the transfer of roles and products needs to be done in a structured and strategic manner so as to minimize and alleviate some of the issues and challenges involved in globalizing software product development. There are factors to consider in terms of product, process and people to identify the work that has to be transitioned as well as factors that influence the transfer process.

Processes involving effective communication, coordination and knowledge acquisition could help stakeholders across multi-sites to have shared understanding during Requirements Engineering phase of software development life cycle, as studied
by Damian (2007). The study further clarifies that defining a clear organization structure and creating roles and assigning the right responsibilities is one of the strategies to alleviate the challenges in GSD implementation.

Khanna (2017) suggests that providing authority and freedom to the management at the offshore locations, as one of the key approaches to the talent management challenges faced at these global locations. He also records that authority and responsibilities of the various roles at the global locations need to be in line with the strengths of these locations.

Similar observation was made by Puneet et al in a McKinsey report (2013), wherein it is emphasized that offshore development teams should front-end product development teams and that the offshore development centres should own and aim to manage end-to-end business processes.

Extending this point of placing the right roles at global locations (co-location) as one of the solutions in addressing the key challenges in a GSD set up is explored in the next section.

### 2.6.1 Co-located key roles at global locations with Product Engineering or Software Development

Prior to GSD’s evolution, roles that were associated with software development life cycle were based in one location and hence many of the issues caused due to distributed development did not apply in such conventional approach.

The practice of GSD involved moving Software Development/Engineering projects which had maximal benefits in terms of labour cost savings and availability of skilled human resources. Strategic and key roles such as Business Analysts, or Product Management typically involved lesser number of people as compared to Software Development and hence movement of those roles to offshore was neither pursued actively
nor did follow any structured or planned approach. To an extent, they were also considered as corporate roles and were kept closer to the headquarters. Most GSD implementations focused offshoring the software development roles in isolation without these adjacent roles and it possibly led to eroding the merits of co-located software making and led to many of the issues listed earlier.

In the following sections, literature that relate to co-locating key roles at global locations along with Product Engineering/ development have been reviewed. Such an approach would lead to having the business benefits of GSD and yet be able to combine the merits of co-located business roles and functions.

Boland and Fitzgerald (2004) discusses in a case study notes that communication efficiency came down when the project was shifted from a co-located scenario to a distributed during a case study.

Battin et al (2001) discuss various issues in GSD including communication and coordination and one of the solutions recommended therein is about transitioning the complete subsystems to the offshore global location with full life cycle responsibility, instead of in-parts. Inhibited communication and coordination issues witnessed during GSD led to reduced productivity and increased intervals between product releases, as per Mockus et al (2001) when talking about distributing roles and activities across sites without co-locating them.

As per Carmel et al (2006), while narrating the four stages of Offshoring of IT in SITO Model (Sourcing of IT work offshore) names the fourth and final stage in the maturity cycle as proactive strategic focus. As per the model, at this fourth stage, global sourcing centres would be performing core corporate activities including new product development, innovations leading to global growth and these offshore centres would house people with domain expertise.
Lane et al (2008) analyse the impact on project deliverables with respect to placement of various roles in a GSD set up. They cite that when Product Designers, Business Analysts, Development and Support personnel were centrally placed to be accessed by remote stakeholders, it led to coordination issues.

This observation is in line with the empirical study done by Botzenhardt et al, (2011) which establishes that cross-functional integration across Product Management and Product development roles positively affect the product release’s success. Similarly, Yan (2004) talks of software maintenance support offered from offshore locations helps improving overall efficiency. The study was done based on a case study at an actual project.

When discussing on benefits of Co-location, literature do not restrict its applicability to Software industry but also extend to other hi-tech industries as well. A case study done by Terwiesch et al (2001) connected to data storage on the international product transfer from USA to an Asian country, talk on the role of co-located key functions at offshore locations enabling addressing communication and co-ordination issues.

Drawing from the above discussion, it could be argued that transitioning key roles such as Product Management and Product Support in addition to and relation to Software Development/ Engineering could address some of the key challenges like communication, coordination issues, knowledge management and process maturity involved in GSD. Literature quote that successful offshoring and GSD could influence the business drivers of the software products and this in turn could influence the drivers of product life cycle.

There is a research gap that needs to be highlighted. Not much research is available to study the drivers and effects when roles such as Product Management or
Product Support were offshored as compared to what was published on Offshoring of Software Product Engineering or Delivery roles.

2.6.2 Product Maturity stage as a key factor of considerations during product transfer decisions of GSD

The basis on which products are chosen to be transferred to global locations, could play a vital role in having these issues and challenges of GSD reduced. This section reviews published literature on the role of maturity stage of the product being an important factor of consideration in ‘product transfer’ decisions during globalization and in influencing factors that drive product life cycle. A detailed case study was done at Ericsson’s Offshore-insourcing operations by Smite et al (2013) on strategies that facilitated software product transfers. The study included products that were done with as well as in the process of transfers across Sweden and a couple of locations in Asia. They summarized that product maturity, complexity and market pressure are three key factors that influence the product transfer process. They also cite that complex development activities may be inappropriate when development is done at multiple locations with the follow-the-sun approach. This could be mapped to the maturity stages of the product since it is typically at the earlier part of the life cycle when the development work would be more complex.

Smite et al (2013) emphasize that products so transferred need to have sufficient time left in their life cycle to be able to ramp up, justify the costs incurred in transferring and thus allow reaching economic benefits. They conclude in their study that, “the degree of realization of projected benefits of GSD is influenced by software life cycle’s stage at which the offshore insourcing is rolled out”. This would translate to the fact that the maturity stage of a product is a key factor in realization of GSD’s benefits.
Inhibited communication and coordination issues witnessed during GSD in turn lead to reduced productivity and increased intervals between product releases, as per Mockus et al (2001). Therefore, it is important to identify the right candidates or entities/chunks of work for transfer across global locations. Their study lists the criteria based on which these candidates are chosen for transfer as: 1) Transfer by functionality, 2) Transfer by localization 3) Transfer by development stage and 4) Transfer by maintenance stage – which in a way is synonymous to product maturity stage.

Cormican et al (2009) in their analysis on the role of technology transfer across global locations in extending product life cycle, list the maturity stage of product life cycle as a key factor among others that include nature of technology, cultural differences and geographic distance between locations.

These literature establish that when business decisions are made on identifying products to be transferred to global locations during GSD, irrespective of popular drivers like cost and skill gap oriented considerations, the maturity stage of the product has an important role in establishing the stability and smoothness of the transition process and deriving the success and benefits of GSD.

The study of Cormican et al (2009) relates this attribution of product maturity stage as a key factor during product transfer decisions to the possibility that, less mature a product is, less could be the availability of information on design and development and hence could it calls for higher efforts during transfer.

Herbsleb and Mockus (2003) cite that stable projects are more advantageous for GSD as they would call for less changes during requirements management phase and hence coordination issues could be less. In this context, complex projects could be more or less mapped to products at their initial phase of the life cycle.
Hahn et al (2011) approach the explanation of why matured products are preferred to be offshored from a different perspective. They quote that offshoring of service activities involving lower level of skills and lower information sensitivities would occur earlier than those with higher levels; this could arguably be mapped to Matured versus Growth stage products as well, in terms of product life cycle stages.

Mockus et al (2001) explain the reason of why products considered for transfer is typically based on maintenance or maturity stages. Products in maintenance phase would typically have less scope for addition of new features and hence could be transferred to sites that are not involved in developing new functionality. Therefore, their perspective is, resources that are capable of developing new functionality are freed up from maintaining it, in the study.

Battin et al (2001) discuss various issues in GSD including communication and coordination and one of the few solutions recommended therein talk about transitioning the complete subsystems to the offshore global location with full life cycle responsibility, instead of in parts. This could be related to mature products that could be given the full life cycle management control at offshore locations.

Going beyond IT industry, consideration to transfer products at their matured stage also applies in manufacturing industry. A case study conducted by Alex et al (2011) in a Pump manufacturing company in Denmark to explore and identify the most optimal point to offshore, conclude that companies that offshore at pre-matured phases could face high and unexpected implementation costs and therefore recommend that offshoring happens when the production is at an optimal point in terms of maturity and control. While hard core-manufacturing and software product development follow two completely different production cycles with different market dynamics, it could be yet
compared and seen the relevance between products’ maturity stage and their transfer to
global locations.

Drawing from the above discussion, the important theme linking all the papers cited in this section is that product maturity stage is one of the key factors considered while choosing products to be transferred to global location during GSD. It makes the product transfer process less painful and facilitates addressing some of the key challenges and issues of GSD. As discussed earlier, successful offshoring and GSD could influence the business drivers of the software products and this in turn could influence the drivers of product life cycle.

However, the literature does not discuss on the results of products’ transfer on parameters like customer satisfaction, sales and its enablers and the enhanced ability to add new product characteristics at each stages of their maturity stages. These would need further research and empirical validation which is one of the main themes or objectives of this study.

2.6.3 **Software type based on deployment – ‘On-Premises’ Vs ‘SaaS’**

Since the last decade or so, with the evolution of Cloud technology, ‘SaaS’ or On-demand way of software deployment is becoming increasingly popular. The software industry is witnessing a shift from the ‘On-Premises’, the traditional model of software deployment to the ‘SaaS’ type of software. The technology of Cloud computing has enabled consumers to use software applications hosted in the cloud infrastructure by the providers in an on-demand, as a service, model.

The history of Enterprise software evolved from custom-built software to packaged business applications through increasing service orientation and now, cloud computing and ‘SaaS’ seem to be the next big steps.
‘SaaS’ is evolving in to a compelling alternative to the ‘On-Premises’ type of software. As per Boillat and Legner (2013), when software is hosted and offered as a service as in ‘SaaS’, software vendors are expected to modify the pricing models that reflects new type of revenue streams associated with recurring nature of smaller magnitude than a fixed, one-time type of revenues as associated with the traditional, ‘On-Premises’ type products. The cost-model, therefore for the ‘SaaS’ products, would have marginal costs that would vary since the sales cycle is longer than the traditional model and also since the expected levels of high availability demands higher levels of real-time support. In addition, software vendors also enable and offer a development platform that is hosted and available for deploying add-on type of applications usable through on-line stores. This type of add-on apps evolves in to a combined ‘SaaS’ as well as ‘PaaS’ (Platform as a Service) offerings or multisided platforms. All this has completely disrupted the business model of software product companies and has also made them to review and revisit the traditional cost model and the way the software is developed.

As a result, the cost pressure experienced by the ‘SaaS’ vendors is higher as compared to their competition in the traditional, ‘On-Premises’ software market, since they have to compete in terms of both technology and features. As per Stuckenberg et al (2011), the ‘SaaS’ software vendors ought to manage operational costs including additional costs to maintain quality, provide software support on a continuous basis and sustain the longer sales cycle. They also say that to be able to manage these cost pressures and also to leverage the potentials of ‘SaaS’ type of deployability, there required a certain minimal level or scale of operations for the ‘SaaS’ vendors. In the case of ‘SaaS’ products, the risk of investment is higher than that in the case of traditional license base ‘On-premises’ products. As Cusumano (2010) says, the “In the sense of standardising services
for a more efficient delivery, ‘productization’ of services, is of equal importance as the ‘servitization’ of products”.

Hence, GSD is an important element of the software making process of many global software product companies who are in the process embracing the clients’ adoption of ‘SaaS’ type of products as an alternative to the ‘On-Premises’ types. In this background, it is important to study how GSD and the various transfer strategies are impacted in the case of ‘SaaS’ type of product making as compared to the traditional ‘On-Premises’ type.

As per Tarif Rahman (2018), ‘SaaS’ product development is different as compared to that of ‘On-Premises’ from two perspectives:

- Revenue considerations are different
- Expectations regarding customizations to be done on the software
- The need to build open and flexible products
- Higher frequency of software releases
- Expected higher speed of development
- Roadmap flexibility

With all the cost pressures that was discussed above, (“Classic Informatics”, 2015) outsourcing ‘SaaS’ product development makes a lot of economic sense also since ‘SaaS’ product development is quite different as compared to the traditional ‘On-Premises’ type. Agile type of development methodology that is very relevant when it comes to ‘SaaS’ type of product development and a globally distributed development team obviously warrants wisely invested time, energy and product development budget.

With GSD being an inevitable element of the cost structure of enterprise software products and in spite of most of them also being in the market of having the cloud-hosted or ‘SaaS’ type of software products, there is not much of literature or studies in the public
domain that have gone in to detail of how GSD could contribute to ‘SaaS’ type of product development, more so with added cost pressures they have as cited in the earlier section. There is no empirical research done to evaluate if the type of software plays any role in addressing the challenges involved in rolling out GSD or its strategies and also on the influence of transfer strategies of GSD on strategic parameters like sales enablement or customer satisfaction. This literature gap has led to a research question of evaluating the influence of the software type on the impacts of two key transfer strategies of GSD, in this study.

2.7 Relationship between GSD, Sales Enablement, Customer Satisfaction and Product life cycle

Product Life cycle is a theory or concept that maps the life horizon of a product in reference to the demand over time right from initial design or development stage till obsolescence. The cycle is plotted as a curve that is mapped in to four distinct stages starting from introduction, growth, maturity and decline. The International Product Life Cycle Theory, authored by Vernon (Vernon, 1992) to explain the cycle that products go through when marketed internationally, describes how a product matures and declines as a result of globalization.

It is primarily considered a marketing theory connecting time with the sales volume, revenue or profits which are typically driven by both market oriented forces that are external to the organization and are also driven by managerial decisions that are internal to the organization.

Cormican et al (2009) reciprocate an observation in their analysis on the role of technology transfer across global locations in extending product life cycle and list ‘maturity stage’ of product life cycle as a key factor among others. It is in line with the
research focus of establishing maturity stage based transfer strategy could in turn result in extending the product life cycle.

The Product Life cycle theory referred by Kotler et al (1990) typically relates the sales of products over time. As quoted by Stephan et al (2008) on the Offshoring Research Network (ORN) Survey findings, growth strategy, competitive pressure, improved service levels and business process redesign were the strategic drivers of GSD in addition to reduced labour cost and access to skilled resources. Massini et al (2010) record that drivers of offshoring include competitive pressure, growth strategy, and ability to cater to new geographies and accelerated go-to-market feasibility. With these strategic sales drivers directly related to offshoring, one could associate GSD’s successful implementation to the internal or organizational drivers of product life cycle. As per Kazemi et al (2011), extending the ‘product-availability’ time during the maturity stage would increase the sales of the product. Higher sales would mean extended product life cycle.

Also, Meenaghan and O'Sullivan (1986) explain that Product life cycle's length and shape are not just based on the unit sales as depicted in the Y axis, but also other attributes like margin or product's characteristics and other marketing strategies employed. Hence, when GSD is successfully implemented, sales enablers like reduced sales lead time could directly and positively influence the product life cycle.

As Van Der Wiele et al (2002) indicate, there is a positive relationship between organizational performance indicators, like sales volume or sales margin customer satisfaction. Prasad (2012) adds that customer satisfaction ratings influence sustainability of product life cycle. Hallowell (1996) extends the discussion on how customer satisfaction is related to profitability which in turn is one of the drivers of product life cycle.
It could be therefore be summarized that literature brings out the relationship that GSD has sales enablement and related metrics and that customer satisfaction in turn could enable sales metrics, profitability and sustenance of product life cycle. This though has to be evaluated empirically which is the core objective of this study.

2.8 Chapter Summary

The literature reviewed in this chapter discuss in detail on the definitions, history and evolution behind both the concept and practice of Global Software Development followed by global enterprise software companies. Various literature cited, discuss in detail on the drivers, benefits, success factors, issues and challenges in the practice of GSD. Studies that have quoted some of the strategies at a high level that help addressing some of the key challenges in enabling GSD have been highlighted which have been considered as the variables and determinants for this study. Literature on the disrupting evolution of cloud based or ‘SaaS’ type of software products as against the traditional ‘On-Premises’ types and their cost models have been referred.

The literature gap with reference to evaluating if and how the two types of software when developed by GSD model have any moderating role on the impact on GSD roll out has been highlighted which in turn is part of the scope of this study. Other gaps in the literature such as not having specific studies with respect to GSD issues when it comes to software product development as much as software services and lack of empirical studies with respect to the impact of GSD strategies connected to products’ maturity stages have also been highlighted in this section. In addition, some of the key terms have also been defined and explained as cited by various literature.

Theories on product life cycle and its determinants have been quoted from various literature to be able to connect GSD and organizational drivers of product life cycle which
is the main theme of this study. As Day (1981) explains, it is not easy to ascertain the present life cycle position of a product unambiguously established. The boundaries between the maturity stages are elusive and planners could not accurately determine the maturity stage of a product or in particular when and how the external factors could influence the product life cycle. In this background, it is beneficial for the product companies to be able to understand if and how GSD and its strategies could effectively be applied to be able to influence some of the drivers of product life cycle.

Table 2.2 presents a table that lists the key studies and literature discussed in the above sections.

| GSD and Offshoring - Evolution and Definition | Edwards and Sridhar, 2005; Aspray et al, 2006; Carmel et al, 2006; Davis et al, 2006; Lewin et al, 2006; Sangwan et al, 2006; Steven and Anil, 2006; Avram, 2007; Chakrabarty, 2007; Rönkkö et al, 2007; Contractor et al, 2010 |
| Strategy of co-located key roles at global locations with Product Engineering or Software Development | Battin et al, 2001; Mockus et al, 2001; Terwiesch et al, 2001; Boland and Fitzgerald, 2004; Yan, 2004; Carmel et al, 2006; Lane et al, 2008; Botzenhardt et al, 2011 |
| Product Maturity stage as a key factor of considerations during product transfer decisions of GSD | Battin et al, 2001; Mockus et al, 2001; Herbsleb and Mockus, 2003; Cormican et al, 2009; Alex et al, 2011; Hahn et al, 2011; Smite et al, 2013 |
| Software type based on deployment – ‘On-Premises’ Vs ‘SaaS’ | Cusumano, 2010; Stuckenborg et al, 2011; Tarif Rahman, 2018 |

Table 2.2. Key studies and topics

Chapter 3 discusses the research questions and the conceptual framework for the development of the various hypotheses included used in this study.