

DEPAUL UNIVERSITY

**Service Oriented Architecture Metrics Suite for Assessing and  
Predicting Business Agility Results of SOA Solutions**

by

Mamoun A. Hirzalla

Dissertation Thesis submitted in partial fulfillment for the  
degree of Doctor of Philosophy

In the  
School of Computing  
College of Computing and Digital Media  
Chicago, Illinois

Advisor:

Jane Cleland-Huang, PhD

June 2012

**DePaul University**  
College of Computing and Digital Media

**Dissertation Defense Report**

I have read the dissertation written by:

Name \_\_\_\_\_ SSN \_\_\_\_\_

**(To the advisor):** The following dissertation title is identical to the one on the title page of the draft returned to the student. This title is approved by me and it is to be used when the final copies of the dissertation are prepared.

Title of dissertation:

---

---

Advisor's Initials \_\_\_\_\_

Acceptable. Candidate may proceed to make final copies

Pass, with revisions stated below:

---

---

Not Acceptable. Please explain:

---

---

_____ Advisor (Print Name) Date	_____ Signature
_____ 1 <sup>st</sup> Reader (Print Name) Date	_____ Signature
_____ 2 <sup>nd</sup> Reader (Print Name) Date	_____ Signature
_____ 3 <sup>rd</sup> Reader (Print Name) Date	_____ Signature
_____ 4 <sup>th</sup> Reader (Print Name) Date	_____ Signature



***Say: "My Lord! Increase me in knowledge."***

The Holy Quran, Surah 20, Verse 114

DEPAUL UNIVERSITY  
School of Computing  
College of Computing and Digital Media

***Abstract***

**Service Oriented Architecture Metrics Suite for Assessing and Predicting Business  
Agility Results of SOA Solutions**

by Mamoun A. Hirzalla

Service Oriented Architecture (SOA) is emerging as the predominant architectural style for building applications due to its promised business agility benefits. Business agility refers to an organization's ability to adapt rapidly and effectively in response to changing business conditions. Despite the significant claims of SOA vendors concerning the promised business agility benefits as a result of deploying SOA solutions, there are currently few SOA metrics designed to evaluate agility in general and none of which relate these metrics to SOA solutions. Moreover, such claims have never been empirically verified. Given the non-trivial investments associated with the development of SOA solutions, predicting business agility outcome grows increasingly important to project executives and stake holders.

This research develops an empirically validated Predicted Business Agility Index (PBAI) which is designed to measure the expected business agility of SOA deployments. The PBAI is constructed through statistically analyzing the relationship between 150 technical attributes and the attainment of business agility in 39 SOA deployments. The PBAI utilizes metrics as part of a SOA metrics framework. The framework includes both service-level and deployment-level SOA metrics to address the impact of architectural decisions on business agility qualities. Both service-level and deployment-level SOA

metrics are used to provide a perspective into SOA solutions' potential business agility outcome.

Data analysis showed that architecture (SOA), business process management (BPM), and impact analysis (IA), were the primary predictors for achieving business agility. Other hypothesized factor such as Loose Coupling (LC) was likewise an important contributing factor to business agility, but to a lesser extent. Surprisingly, our hypothesized Governance factor did not turn out to be significant as data analysis showed no significant divergence among projects with respect to governance. This suggests that, at least in the projects studied, governance is not a predictor of business agility. Further cross-validation of the PBAI model has shown high correlation with attained business agility. Finally, the investigation into the relationship between business agility and solution complexity revealed a weak positive correlation. This indicates that SOA solutions that produce higher levels of business agility are marginally more complex than those that are less business agile.

## *Acknowledgements*

First and foremost, all praise and thanks to God who granted me the will, determination, commitment and for his blessings through the support of loved ones, friends and colleagues to finish this milestone. I would like to thank my advisor, Jane Cleland-Huang, for her support and guidance throughout my years at DePaul. She has provided significant insight on how to enhance my research and writing style. In addition, Jane was a mentor to me on various topics and I do appreciate all the help you provided. I would also like to thank Craig Miller, Xiaoping Jia, Ugo Buy and Peter Bahrs for being on my defense committee and for the valuable feedback, thoughts and questions they provided.

Many thanks go to my IBM colleagues who participated in the data collection steps and endured many hours of questions and communications. Most importantly, a big thank you for Rob High, Peter Bahrs and Stefan Pappé for their valuable insight and for agreeing to sponsor IBM's Academy of Technology 3<sup>rd</sup> SOA Best Practices conference that generated the majority of collected data. Thanks to all the core team members of the conference and non-IBM survey participants who volunteered their time and effort in the data collection process. I also thank my colleague Ali Arsanjani for his early feedback into this research and his encouragement to pursue my initial thoughts and ideas. My gratitude goes to IBM, DePaul and National Science Foundation for funding this research. I would not have been able to complete my work without your help.

Finally, my family deserves a lot of the credit for where I am now; my mother and father for instilling the love of learning in me as long as I can remember. Most importantly, I would like to thank my wonderful wife Nuha for her love, support and encouragement over the years. I also appreciate our kids' (Jenine, Mohammad and Beesan) patience. Despite their young age, they never complained, ok may be a little bit, when I had to spend a lot of weekends and nights working on my research. That long promised trip to Disney can finally happen!

**To My Parents,**

For their struggle to raise 11 children and see most of them finish their education and realize their dreams!

# Contents

Approval Page.....	ii
<i>Abstract</i> .....	iv
<i>Acknowledgements</i> .....	vi
List of Tables.....	xii
List of Figures.....	xiv
Abbreviations.....	xvi
Definition of Key Terms.....	xvii
CHAPTER ONE: INTRODUCTION.....	20
1.1 Problem Statement.....	21
1.2 Research Questions.....	24
1.3 Contributions and Professional Significance of Study.....	31
1.4 Assumptions of Study.....	32
1.5 Published Work.....	33
1.6 Organization.....	34
1.7 Summary.....	35
CHAPTER TWO: SERVICE ORIENTED ARCHITECTURE.....	36
2.1 Introduction.....	36
2.2 Business Services vs. Web Services.....	37
2.3 SOA Lifecycle.....	38
2.4 SOA Governance.....	40
2.5 SOA Methodologies.....	42
2.6 Major SOA Architectural Components.....	46
2.6.1 Service Registry.....	46
2.6.2 Enterprise Service Bus (ESB).....	47
2.6.3 Business Process Management (BPM).....	48
2.7 Business Agility as an Outcome of SOA.....	49
2.8 Summary.....	50
CHAPTER THREE: BUSINESS AGILITY.....	52
3.1 Introduction.....	52
3.2 Business Agility.....	53
3.3 Types of Business Agility.....	54
3.4 SOA and Business Agility.....	55
3.5 Business Agility Contributors.....	56
3.5.1 SOA Architecture.....	58
3.5.2 Business Process Management.....	62
3.5.3 Impact Analysis.....	65

---

3.5.4 Loose Coupling .....	68
3.5.5 Governance.....	71
3.6 Challenges of Measuring Business Agility.....	77
3.7 Measuring Business Agility.....	80
3.8 Summary.....	83
<b>CHAPTER FOUR: SOFTWARE METRICS OVERVIEW .....</b>	<b>84</b>
4.1 Introduction.....	84
4.2 Software Metrics.....	84
4.3 Software Metrics Pitfalls .....	85
4.3.1 Metrics Pitfall: Improper Measurements.....	85
4.3.2 Metrics Pitfall: Improper Scales.....	87
4.3.3 Metrics Pitfall: Improper Metric Validation.....	89
4.3.4 Metrics Pitfall: Improper Use of Statistics .....	91
4.4 Addressing Metrics Pitfalls.....	92
4.5 Acceptance of Software Metrics.....	94
4.6 Summary.....	95
<b>CHAPTER FIVE: SOA AND BUSINESS AGILITY METRICS .....</b>	<b>97</b>
5.1 Introduction.....	97
5.2 SOA Metrics Literature Review .....	97
5.3 Business Agility Metrics Literature Review.....	102
5.4 Summary.....	105
<b>CHAPTER SIX: RESEARCH DESIGN AND METHODOLOGY .....</b>	<b>106</b>
6.1 Introduction.....	106
6.2 Application of GQM-MEDEA Process Steps .....	109
6.2.1 Setting Up of Empirical Study .....	109
6.2.2 Definition of Measures of the Independent Attribute.....	111
6.2.3 Definition of Measures of the Dependent Attribute.....	126
6.2.4 Hypothesis Refining and Verification .....	127
6.3 Data Analysis and Model Building.....	127
6.3.1 Selection of Data Analysis Method.....	127
6.4 Predictive Model Building.....	128
6.5 Data Validation and Construct Validity.....	129
6.6 Summary.....	129
<b>CHAPTER SEVEN: DATA COLLECTION PROCESS AND APPLICATION .....</b>	<b>131</b>
7.1 Introduction.....	131
7.2 Data Collection Process.....	131
7.3 Population .....	132
7.4 SOA Project Profile .....	132
7.4.1 Participants .....	133
7.4.2 Data Collection Activity and Instrumentation.....	134
7.5 Survey Overview .....	136

---

7.6 Summary .....	138
CHAPTER EIGHT: ANALYSIS OF DATA .....	139
8.1 Introduction.....	139
8.2 Descriptive Statistics.....	140
8.3 Calculating BAI Values for Collected Data .....	143
8.4 Factor Analysis .....	144
8.5 BAI Scores for Business Agility Factors .....	145
8.5.1 SOA Architecture .....	146
8.5.2 Business Process Management (BPM).....	148
8.5.3 Impact Analysis .....	151
8.5.4 Loose Coupling .....	153
8.5.5 Governance.....	155
8.6 Preliminary Results.....	157
8.7 Building a Predictive Model for Business Agility using Regression Analysis .....	158
8.8 Building the Predicted Business Agility Index (PBAI).....	160
8.8.1 Primary Hypothesis .....	160
8.8.2 Building the Predictor Model .....	161
8.9 Results Analysis.....	163
8.9.1 Validating Regression Analysis Assumptions.....	165
8.10 Predictive Model Cross Validation.....	168
8.11 Impact of Achieving Business Agility on Complexity.....	171
8.12 Applying PBAI for Not Included Projects.....	173
8.13 Threats to Validity .....	176
8.13.1 Construct Validity .....	177
8.13.2 Internal Validity.....	178
8.13.3 External Validity .....	178
8.13.4 Conclusion Validity.....	178
8.14 Limitations .....	179
8.15 Summary.....	180
CHAPTER NINE: CONCLUSIONS AND IMPLICATIONS .....	182
9.1 Introduction.....	182
9.2 Summary of Results.....	182
9.3 Using the PBAI in Real SOA Projects.....	183
9.4 Future Work .....	184
9.4.1 Additional Metrics.....	184
9.4.2 Extensive Empirical Validation.....	184
9.4.3 Embedding in Existing Methodology and Tools.....	185
APPENDIX A GQM-MEDEA .....	186
APPENDIX B SURVEY QUESTIONS.....	194
APPENDIX C SCORING METHOD PER SURVEY QUESTION TYPES.....	256

**Contents** **xi**

---

APPENDIX D SURVEY PARTICIPANTS ORIENTATION SESSION .....257

BIBLIOGRAPHY.....267

## List of Tables

Table 1 Business Agility Factors .....	58
Table 2 Business Agility Factors and Associated Attributes.....	76
Table 3 Business Agility Index Questions.....	82
Table 4 Measurement Goals per GQM-MEDEA .....	110
Table 5 Business Agility Factors and Rationales for Inclusion.....	111
Table 6 SOA Architecture Business Agility Factor and its Attributes .....	114
Table 7 Business Process Management Business Agility Factor and its Attributes.....	116
Table 8 Governance Business Agility Factor and its Attributes.....	119
Table 9 Loose Coupling Business Agility Factor and its Attributes .....	121
Table 10 Impact Analysis Business Agility Factor and its Attributes .....	124
Table 11 Business Agility Factors and Associated Attributes.....	125
Table 12 Survey Sections.....	137
Table 13 Projects by Industry and Duration .....	141
Table 14 BAI Assessment of 39 SOA Deployments.....	144
Table 15 Business Agility Factors .....	146
Table 16 Means for Normalized Business Agility Factors Scores .....	158
Table 17 Simple Correlations between BAI Scores and Business Agility Factors .....	162
Table 18 Multiple Linear Regression Results of Business Agility Factors .....	162
Table 19 Business Agility Contributors.....	165
Table 20 Predicted BAI vs. Reported BAI for All Projects.....	170
Table 21 SCI Components.....	171
Table 23 Business Agility Factors and BAI Scores for Unclassified Projects .....	173

**List of Figures****xiii**

---

Table 22 Simple Correlation between BAI and SCI.....	173
Table 24 Means for Business Agility Factors for both Agile and Not Business Agile ..	174
Table 25 BAI vs. PBAI for Unclassified Projects .....	176
Table 26 Measurement goals per GQM-MEDEA .....	190
Table 27 Survey Questions Scoring.....	256

## List of Figures

Figure 2 Relationships between Factor, Business Agility Factor and Business Agility Contributor .....	28
Figure 3 IBM SOA Lifecycle [34].....	39
Figure 4 SOMA Phases - An IBM Image [2] .....	43
Figure 5 The UDDI Publish and Lookup Lifecycle [44].....	47
Figure 6 Hub and Spoke Integration vs. Direct Connection [47] .....	48
Figure 7 SOA Architecture Business Agility Factor and its Attributes.....	59
Figure 8 BPM Business Agility Factor and its Attributes .....	62
Figure 9 Impact Analysis Business Agility Factor and its Attributes.....	66
Figure 10 Loose Coupling Business Agility Factor and its Attributes .....	69
Figure 11 Governance Business Agility Factor and its Attributes .....	72
Figure 12 Overall Research Methodology that Shows Application of GQM-MEDEA Process Steps to Our Research Questions – Author’s Image .....	108
Figure 13 GQM-MEDEA Major Steps and Main Activities - Author’s image.....	109
Figure 14 Projects Distribution by Industry and Duration.....	140
Figure 15 Business Agility Outcomes of Surveyed Projects .....	142
Figure 16 SOA Architecture Business Agility Factor Attributes .....	147
Figure 17 Project’s Business Agility Outcomes and SOA Architecture Score .....	147
Figure 18 BPM Business Agility Factor Attributes.....	149
Figure 19 Project’s Business Agility and BPM Score.....	150
Figure 20 Impact Analysis Business Agility Factor Attributes .....	151
Figure 21 Project’s Business Agility and Impact Analysis Score .....	152

**List of Figures****xv**

---

Figure 22 Loose Coupling Business Agility Factor Attributes.....	153
Figure 23 Project’s Business Agility and Loose Coupling Score.....	155
Figure 24 Governance Business Agility Factor Attributes.....	156
Figure 25 Project’s Business Agility and Governance Score .....	156
Figure 26 BAI vs. Predicted BAI for All Projects .....	170
Figure 27 BAI vs. SCI for All Projects.....	172
Figure 28 Projects 1-7 Business Agility Contributors Profile vs. Agile and Not Agile Profiles .....	175
Figure 29 Overall Research Methodology that Shows Application of GQM-MEDEA Process Steps to Our Research Questions – Author’s Image .....	186
Figure 30 GQM-MEDEA Major Steps and Main Activities - Author's image .....	189

## Abbreviations

<b>BAI</b>	<b>B</b> usiness <b>A</b> gility <b>I</b> ndex
<b>BPEL</b>	<b>B</b> usiness <b>P</b> rocess <b>E</b> xecution <b>L</b> anguage
<b>BPM</b>	<b>B</b> usiness <b>P</b> rocess <b>M</b> anagement
<b>CLT</b>	<b>C</b> entral <b>L</b> imit <b>T</b> heorem
<b>DE</b>	<b>D</b> irect <b>E</b> xposure
<b>ESB</b>	<b>E</b> nterprise <b>S</b> ervice <b>B</b> us
<b>GQM</b>	<b>G</b> oal <b>Q</b> uestion <b>M</b> etric
<b>GQM-MEDEA</b>	<b>G</b> oal <b>Q</b> uestion <b>M</b> etric – <b>M</b> etric <b>D</b> Efinition <b>A</b> pproach
<b>GSM</b>	<b>G</b> oal <b>S</b> ervice <b>M</b> odeling
<b>IE</b>	<b>I</b> ndirect <b>E</b> xposure
<b>KPI</b>	<b>K</b> ey <b>P</b> erformance <b>I</b> ndicator
<b>OO</b>	<b>O</b> bject <b>O</b> riented
<b>PBAI</b>	<b>P</b> redicted <b>B</b> usiness <b>A</b> gility <b>I</b> ndex
<b>ROI</b>	<b>R</b> eturn <b>O</b> n <b>I</b> vestment
<b>RUP</b>	<b>R</b> ational <b>U</b> nified <b>P</b> rocess
<b>SAM</b>	<b>S</b> ervice <b>A</b> ccess <b>M</b> ethod
<b>SCA</b>	<b>S</b> ervice <b>C</b> omponent <b>A</b> rchitecture
<b>SCI</b>	<b>S</b> olution <b>C</b> omplexity <b>I</b> ndex
<b>SLA</b>	<b>S</b> ervice <b>L</b> evel <b>A</b> greement
<b>SLT</b>	<b>S</b> ervice <b>L</b> itmus <b>T</b> est
<b>SOA</b>	<b>S</b> ervice <b>O</b> riented <b>A</b> rchitecture
<b>SOMA</b>	<b>S</b> ervice <b>O</b> riented <b>M</b> odeling and <b>A</b> rchitecture
<b>UDDI</b>	<b>U</b> niversal <b>D</b> escription <b>D</b> iscovery and <b>I</b> ntegration
<b>USDP</b>	<b>U</b> nified <b>S</b> oftware <b>D</b> evelopment <b>P</b> rocess
<b>WSDL</b>	<b>W</b> eb <b>S</b> ervice <b>D</b> efinition <b>L</b> anguage

## Definition of Key Terms

### **Definition 1: Service Oriented Architecture (SOA)**

Service Oriented Architecture (SOA) is defined as an architectural style for integrating business processes and other IT components through the use of standards-based reusable services to achieve alignment between business and IT to inject flexibility into an organization [1] [2] [3].

### **Definition 2: Business Agility**

Organization's ability to effectively sense, manage, adapt and respond efficiently and effectively to produce a desired outcome to opportunities, challenges and competition, through leveraging business strategy, people and information technology.

### **Definition 3: Factor**

A factor is defined as an object (such as a person or a room) or an event (such as a journey or testing phase of a software project) in the real world [4].

For the purposes of this research, a factor represents any element that may impact a SOA solution such as organizational structure, learning skills, SOA architectural components, etc.

### **Definition 4: Business Agility Factor**

A business agility factor is a factor that is hypothesized to contribute to attainment of business agility.

**Definition 5: Business Agility Contributor**

A business agility contributor is a business agility factor that was analyzed and empirically validated through statistical means to have contributed to attaining business agility.

**Definition 6: Attribute**

An attribute is a feature or property of a factor or a business agility factor or a business agility contributor. Typical attributes include the area or color (of a room), the cost (of a journey), or the elapsed time (of the testing phase) [4]. For the purposes of this research, the Impact Analysis factor includes attributes such as service level agreement (SLA) measurement, monitoring, etc.

**Definition 7: Business Agility Index (BAI)**

BAI is a tool that is developed to measure after-the-fact attainment of business agility in each of the studied SOA deployments. BAI is used to compare attainment of business agility benefits across SOA projects. The BAI is a scale that can range from 0, lowest, to 8. A project with high BAI values is considered to have achieved business agility benefits. Alternatively, SOA projects with low BAI values are not.

**Definition 8: Predicted Business Agility Index (PBAI)**

PBAI is a model that is used to explain the variability in business agility results that are indicated in the analyzed SOA projects. The PBAI is calibrated on a scale that can range from 0, lowest, to 8. A project with high PBAI values is considered a potential strong promoter of business agility. Alternatively, SOA projects with very low PBAI values are not and may require attention in order to achieve business agility benefits. Projects with medium PBAI values are considered likely promoters of business agility.

**Definition 9: Solution Complexity Index (SCI)**

SCI is a tool that is developed to differentiate SOA projects according to complexity in terms of number of requirements, project personnel, number of services, number of processes, and duration. The SCI is a scale that is normalized and can range from 0, lowest, to 10. A project with high SCI values is considered a complex project. Alternatively, SOA projects with low SCI values are less complex relative to the ones with higher SCI values.

## Chapter One: Introduction

Service Oriented Architecture (SOA) is an architectural style designed to inject flexibility into an organization and achieve alignment between business and Information Technology (IT). It accomplishes this through using standards-based reusable services to integrate business processes and other IT components [1][2][3]. The introduction of SOA into the enterprise has resulted in the proliferation of enterprise level solutions that significantly leverage services to solve very large scale problems [1][5]. As a result, it is now almost impossible to conduct business on the web without using some type of service. For example, if you purchase a book online, it is likely that you will be using different levels of services to retrieve and display the right inventory information, to verify your shipping address, compute correct taxes for your state, and to authorize your credit card information.

One of the primary reasons that businesses have so readily adopted SOA solutions is to increase their ability to respond to changing market conditions and to meet new business challenges manifested through changing requirements. Moreover, the maturity of standards that govern all aspects of services and service interoperability interactions has benefited services as it develops into a serious distribution model for the enterprise [5]. As SOA becomes popular, it mandates changes from both business and technical perspectives on how systems are built. From a business perspective, one of the primary objectives of SOA-based systems is the alignment between business and IT and its impact on business agility of organizations adopting SOA [1] [6]. Business agility is claimed to be realized through the decomposition of business processes into more manageable and pluggable business services, combined to form new business processes that meet the changing business needs of the organization [1] [6] [7] [8]. This research, however, has shown such decomposition alone is insufficient for achieving business agility. From a technical perspective, SOA provides a framework for creating concrete

web services that are implemented using either new components or reusing existing components that collaborate in fulfilling the business need. Such concrete web services can participate in various business processes as standalone services or they can be combined with other services to create composite services [2] [7].

Business agility is a term that has gained considerable popularity and as a result there are many different definitions. Many authors provided their own definitions that agree on an organization's ability to adapt proactively to unexpected and unpredicted changes [9][10][11][12][13]. The achievement of business agility is usually attributed to many factors that range from organizational structure, finance policies, alignment of business and IT, infrastructure, leveraging the impact of people and information, education and skills [9][11][14]. It is worth noting that many authors use the term enterprise agility to mean the same thing as business agility [11][15][16]. In this research, the term business agility will be used since it is referenced the most in the currently available SOA literature.

### **1.1 Problem Statement**

Although business agility is considered one of the primary benefits achieved from building SOA solutions, this claim has never been examined or validated. A review of the available SOA metrics and business agility literature reveals that each topic is considered in isolation. Subsequently, none of the reviewed SOA metrics literature provides a treatment of the relationship between building SOA solutions and the impact of architectural decisions on achieving business agility. To the best of our knowledge, the questions that are investigated as part of this research are unique.

The development of a SOA solution represents a non-trivial investment in human resources, capital and time. It is often undertaken with the expectations that it will position the organization to respond more adeptly to changing market conditions. Unfortunately, SOA projects that do not consider the impact of various architectural,

process, and governance-related decisions upon future business agility may fail to achieve these desired benefits [7]. Therefore, one of the primary objectives of this research is the development of a quantitative method to measure the attainment of business agility as a result of SOA deployments. Furthermore, we use our previously developed quantitative method for measuring attainment of business agility in the development of a business agility predictor model, which is designed to evaluate a SOA deployment currently under development in order to (i) predict the extent to which it is likely to achieve business agility in the future, (ii) identify deficiencies, and (iii) suggest corrective measures.

Establishing a link between SOA solutions and business agility is fraught with many challenges. This is due to the lack of methods or tools to quantify business agility benefits for SOA solutions. Therefore, this study addresses this gap through the introduction of a method to measure business agility benefits as a result of deploying SOA solutions. To this end, we first introduce the Business Agility Index (BAI) that is developed to measure after-the-fact attainment of business agility in each of the studied SOA deployments. Using the BAI requires expertise in both business and technology, and it can only be applied after the SOA project has been deployed for a period of time and its business agility has been actually put to the test.

Although the BAI is able to largely differentiate between business and non-business agile projects, its usefulness is limited because it requires business expertise and can only be assessed after the fact, once the SOA system has been built, deployed, and its business agility tried and tested. The BAI therefore does not provide predictive value. In contrast the intent of our work is to develop a model with a concrete set of factors and attributes that are easily collectable by technical project personnel during early phases of a project, and which have the capability of accurately differentiating between projects which are likely to attain business agility and those which are not. To accomplish this we constructed the Predicted Business Agility Index (PBAI) by identifying technical factors which could be collected and measured during early phases of the project, and which

were shown to be highly correlated with the attainment of business agility, as measured by the previously created BAI. Figure 1 shows the typical software development lifecycle activities across the various phases of project’s timeline according to the Rational Unified Process [17]. The diagram illustrates the activities over the various phases of the project and the disciplines of the project. The diagram was amended to illustrate where BAI and PBAI calculations fit within the software development lifecycle and when each index is calculated.

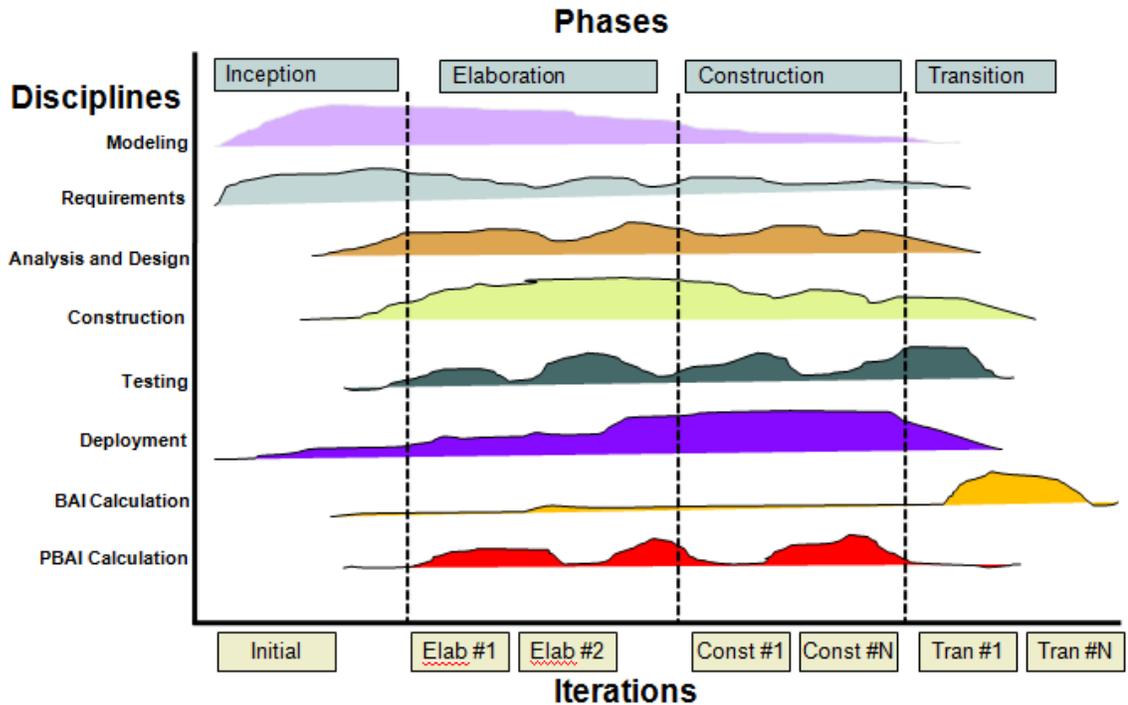


Figure 1 BAI and PBAI during the Project Lifecycle

Projects’ business agility potential outcomes are discovered through the calculation and interpretation of PBAI values. The PBAI values provide organizations with an early warning mechanism to identify projects that may not achieve their desired business agility results. Projects are then further analyzed with respect to PBAI-identified factors

to determine the reasons for potential agility failures. Corrective measures are then enacted to rectify counterproductive project decisions. The application of the PBAI could save organizations a huge amount of money given the costs associated with large SOA projects. This is especially true if those projects are expected to be strategic and transformational in nature and constitute a major investment in resources and time.

To fully understand the impact of attaining business agility on cost, we further extend this research to study the relationship between achieving business agility and complexity. This is done primarily to appreciate the impact that the quest for business agility might have on the complexity of SOA solutions. To this end, we propose a solution complexity index (SCI) that quantifies the overall complexity of a given SOA solution and is used to differentiate among analyzed SOA projects based on their complexity. The correlation between attained business agility (BAI) and solution complexity (SCI) is then analyzed to understand the dynamics of such a relationship.

## **1.2 Research Questions**

As stated previously, one of the key challenges for establishing the link between SOA solutions and their impact on business agility is the lack of measurement tools to assess the business agility achieved as a result of SOA solutions. Moreover, SOA solutions are a mix of many architectural components with varying impacts on business agility. As a result, the need to identify and validate the most common factors that contribute to business agility in SOA solutions is well established. We therefore present the following research questions that this dissertation addresses:

### **RQ1 How can attained business agility be measured in completed SOA projects?**

This question focuses on developing a method for measuring and assessing achieved business agility in completed and deployed SOA solutions. In order to evaluate the impact of SOA solutions on business agility, there needs to be an objective method for measuring their achieved business agility. To accomplish

this objective, we present the **Business Agility Index (BAI)** as a tool that can be used to compare attainment of business agility benefits across completed and deployed SOA projects.

**Definition: Business Agility Index (BAI):**

BAI is a tool that is developed to measure after-the-fact attainment of business agility in each of the study SOA deployments. BAI is used to compare attainment of business agility benefits across SOA projects. The BAI is calibrated on a scale that can range from 0, lowest, to 8. A project with high BAI values is considered to have achieved business agility benefits. Alternatively, SOA projects with low BAI values are not.

In practice, many projects fall somewhere in between, suggesting that they have only partially realized their potential in terms of business agility.

**RQ2 What are the significant measurable technical and process factors of a SOA solution that are correlated with the attainment of business agility?**

Having developed a technique to estimate the business agility of a completed SOA solution (i.e. research question #1), the second question explores in depth the architectural decisions that impact the results of SOA solutions. To build a SOA solution, architects and designers of the system make numerous decisions concerning the selection and implementation of components. The inclusion of some components that may be associated with enhancing business agility in a SOA solution is not necessarily a promoter of overall agility of an organization. Instead, the knowledge of additional architectural details on how a component is implemented could be of more importance in determining business agility contributions.

For example, BPM is generally believed to be an important factor in achieving business agility [1][7][18]. BPM usually includes many elements that together contribute to overall business agility in varying degree. Elements of BPM as indicated in many texts [1][7] may include business rules, use of workflow engine, ease of maintenance of rules to name a few. The inclusion of BPM and some of its elements in a SOA solution does not necessarily indicate any attainment of business agility. The knowledge of implementation details of BPM and its elements is necessary to inspect before making such claims. Implementation details may reveal inflexible business processes or hard to maintain business rules that negatively impact business agility. To illustrate, business rules are generally believed to be good promoters of dynamic behavior in BPM components within a SOA solution. Business rules are critical for fine-tuning and enabling different dynamic behaviors under different business conditions. However, if a business process hard codes business rules in one of its sub-processes, the benefits of business rules are severely impacted and would limit business agility benefits. Therefore, the manner in which business rules are implemented plays an important role in determining whether they contribute positively to achieving business agility.

In addition to BPM, SOA consists of many components that are believed to be core to building SOA solutions such as use of an enterprise service bus, loose coupling, governance and use of registries. Components, referred to as **factors** from now on, collaborate to achieve the stated business agility objectives.

**Definition: Factor**

A factor is defined as an object (such as a person or a room) or an event (such as a journey or testing phase of a software project) in the real world [4].

For the purposes of this research, a factor represents any element that may impact a SOA solution such as organizational structure, learning skills, SOA architectural components, etc.

Given the many components and architectural structures that comprise a SOA solution, there is currently no agreed upon set of architectural entities that are considered factors for achieving business agility. Most available SOA methodologies today provide methods for architecting SOA solutions in the hope of attaining business agility. Moreover, architectural factors are generally broad and may contain many contributing sub-factors (or elements), referred to as **attributes** from now on, which contribute in varying degrees to achieving business agility.

**Definition: Attribute**

An attribute is a feature or property of a factor or a business agility factor or a business agility contributor. Typical attributes include the area or color (of a room), the cost (of a journey), or the elapsed time (of the testing phase) [4]. For the purposes of this research, the Impact Analysis factor includes attributes such as service level agreement (SLA) measurement, monitoring, etc.

To this end, we examine a set of candidate architectural factors that are believed to contribute to business agility, referred to as **business agility factors**, and empirically validate which of their associated attributes are more correlated to achieving business agility than others.

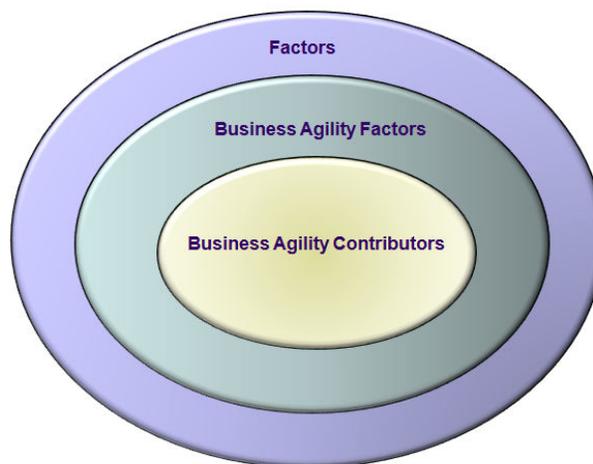
**Definition: Business Agility Factor**

A business agility factor is a factor that is hypothesized to contribute to attainment of business agility.

The investigation into the primary solution factors and their associated attributes is used to validate a set of established hypotheses that we formulated as part of this research. The research hypotheses consider the important factors and attributes and stipulate their impact on business agility. An empirically validated set of attributes are then grouped into the final set of **business agility contributors** that make up the set of metrics for assessing business agility in deployed SOA solutions. Figure 2 Shows the relationship between the terms factor, business agility factor and business agility contributor.

**Definition: Business Agility Contributor**

A business agility contributor is a business agility factor that was analyzed and empirically validated through statistical means to have contributed to attaining business agility.



**Figure 2 Relationships between Factor, Business Agility Factor and Business Agility Contributor**

**RQ3 How can the proposed business agility metrics be used to predict the business agility outcome of in-progress SOA solutions?**

Although the BAI (research question #1) is able to largely differentiate between business and non-business agile projects, its usefulness is limited because it requires business expertise and can only be assessed after the fact, once the SOA system has been built, deployed, and its business agility tried and tested. The BAI therefore does not provide predictive value. In contrast the intent of our work is to develop a model with a concrete set of factors and attributes (research question #2) that are easily collectable by technical project personnel during early phases of a project, and which have the capability to accurately differentiate between projects which are likely to attain business agility and those which are not. To accomplish this we constructed the PBAI by identifying technical factors which could be collected and measured during early phases of the project, and which were shown to be highly correlated with the attainment of business agility.

**Definition: Predicted Business Agility Index (PBAI)**

PBAI is a model that is used to explain the variability in business agility results that are indicated in the analyzed SOA projects. The PBAI is calibrated on a scale that can range from 0, lowest, to 8. A project with high PBAI values is considered a potential strong promoter of business agility. Alternatively, SOA projects with very low PBAI values are not and may require attention in order to achieve business agility benefits. Projects with medium PBAI values are considered likely promoters of business agility.

Projects with low and medium PBAI values require further actions, to varying extents, in order to enhance the likelihood of attaining business agility. In other words, the PBAI suggests additional architectural decisions that have been shown

to promote business agility in previous projects, and which could be expected to increase business agility.

In addition to the primary research questions outlined earlier, we also investigated the following **secondary research question** to provide preliminary results based on the collected data:

**RQ4 Are SOA solutions that achieve business agility generally more complex than those that do not?**

From our observations in the field, striving for business agility as a result of SOA solutions does not come for free. The use of different SOA solution entities, integration and testing introduces many potential challenges to overall SOA solutions. Like every IT project, SOA projects have their own associated complexity. Adding additional entities that are believed to help attain business agility increases the level of overall complexity of resulting SOA solutions. While a thorough treatment of the question of cost and complexity is beyond the scope of this research, we do perform an initial analysis. To be precise, we set out to understand whether those SOA solutions which effectively achieve business agility are generally more complex than those that do not. For the purposes of this research, complexity indicators are used at a very high level and represented in the number of requirements, processes, services, solution components, personnel and timelines that are available through our data collection process.

In order to capture solution complexity, we therefore introduce a **solution complexity index (SCI)** to provide a scale to the complexity associated with each analyzed SOA project. The SCI is a relative index and is constructed as a means to compare projects against the same criteria.

**Definition: Solution Complexity Index (SCI):**

SCI is a tool that is developed to differentiate SOA projects according to complexity in terms of number of requirements, project personnel, number of services, number of processes, and duration. The SCI is a scale that is normalized and can range from 0, lowest, to 10. A project with high SCI values is considered a complex project. Alternatively, SOA projects with low SCI values are less complex relative to the ones with higher SCI values.

**1.3 Contributions and Professional Significance of Study**

The majority of SOA metrics that are available today focus on services as the building blocks of SOA solutions. Very few metrics touch on the overall solution and none of the SOA metrics we were able to find relate the impact of a SOA solution to its most touted benefit of business agility. The metrics proposed in this research differ from existing SOA metrics research in the following ways:

- 1- The empirically validated metrics suite addresses both services and SOA solution attributes and the dynamics between them for a given SOA solution.
- 2- The empirically validated metrics are focused on business agility and its primary factors.
- 3- Other SOA solution considerations that are essential to successful SOA solutions and may have a direct impact on business agility are considered such as SOA Governance.

One of the key benefits of this research is the incorporation of the new proposed SOA metrics suite into SOA design methodologies. Templates can be created to collect the required data as a deliverable to any SOA methodology. The results of this research can easily provide indications to potential business agility outcome based on collected data. This will help architects, project managers and designers to keep an eye towards potential business agility benefits. Moreover, the collection of some of the metrics can be easily

automated through the use of tools to calculate the values of the metrics and provide the interpretation of the values as it relates to business agility. The use of metrics can help project managers and architects to identify minor issues impacting the project's ability to achieve business agility early in the project when course corrections are still possible.

Another key contribution is the introduction of empirically validated tools to address business agility measurements that are objective and generically applicable to any SOA project. The measurements introduced are used for two primary objectives. First, the measures can be used to evaluate the attainment of business agility after the completion of one project phase and before continuing into additional phases of the project. This should allow project managers, architects and executives to evaluate attained business agility benefits. Second, projects that are undergoing design activities can calculate the predicted values for business agility based on the current or planned architectural decisions of a SOA project. The empirically validated thresholds for attaining business agility serve as a useful guideline for project managers and architect through the identification of early red flags. The set of empirically validated business agility contributors should help project managers, architects and executives to focus on demonstrated solutions and architectural decisions that have been shown to significantly impact business agility; thereby allowing project executives to adjust course early in the development lifecycle without incurring painful setbacks to time and resources.

#### **1.4 Assumptions of Study**

This study was developed using a primary set of data that pertained to major SOA implementations across various different industries with varying project complexities. A major assumption for all participating projects was the use of the SOA architectural style as the primary architectural method for building such projects. A profile for the selection criteria and assumptions for the participating SOA projects were documented in the project profile section 7.4.

Although one of the research objectives is to identify the primary factors that impact business agility in relationship to building SOA solutions, there are no guarantees that identified factors are inclusive of all potential factors that impact business agility. In fact, with the rapid pace of innovation in technology and the creation of new software tools and practices, it is very likely that additional factors will play a role in impacting business agility as a result of building IT solutions. Nevertheless, the analysis of 39 SOA projects and the opinions of more than 60 experts in building SOA solutions provide a degree of credibility that the majority of factors that impact business agility as a result of building SOA solutions as we know it today are identified as part of this research. Nevertheless, we do not claim to have investigated all the factors that impact business agility.

### 1.5 Published Work

This dissertation includes work published in the following international peer-reviewed workshops and conferences:

- **Mamoun Hirzalla**, Peter Bahrs, Jane Cleland-Huang, Craig S. Miller, Rob High: *A Predictive Business Agility Model for Service Oriented Architectures*. ICSOC 2011: 653-660 [19]
- **Mamoun Hirzalla**, Peter Bahrs, Jane Cleland-Huang, Craig S. Miller, Rob High: *Analysis of Business Agility Indicators in SOA Deployments*. SERP 2011:400-407 [20]
- **Mamoun Hirzalla**, Andrea Zisman, Jane Cleland-Huang: *Using Traceability to Support SOA Impact Analysis*. SERVICES 2011: 145-152 [21]
- **Mamoun Hirzalla**, Jane Cleland-Huang, Ali Arsanjani: *A Metrics Suite for Evaluating Flexibility and Complexity in Service Oriented Architectures*. ICSOC Workshops 2008: 41-52 [22]

## **1.6 Organization**

This dissertation is organized in the following manner:

Chapter 1 provides general background information and sets the expectations and motivations for this research while describing the primary research questions. Chapter 2 provides an overview of fundamental SOA concepts and a brief literature review for the major SOA themes. Chapter 3 introduces the concept of business agility and how it relates to SOA. Furthermore, the chapter describes the factors that are believed to be major contributors to business agility and introduces our method for measuring business agility. Chapter 4 covers software metrics literature, metrics pitfalls and how this research is avoiding common metrics pitfalls. Chapter 5 presents SOA and business agility metrics and a review of the limited literature that is available for both topics. Chapter 6 provides a detailed treatment of the research methodology and research hypotheses. Appropriate methodology tools and statistical analysis method are presented with the rationales for selecting them and their appropriateness to this research. Chapter 7 documents the data collection process and how data was gathered for this research.

The results of this research are presented in Chapter 8 along with the supporting material. Chapters 9 presents research conclusions and a discussion into the significance of the results and how can they be applied in real life. Moreover, this chapter includes suggestions for future research to answer additional interesting questions that were uncovered during this study.

Finally, appendix A provides a brief description of the GQM-MEDEA methodology that was used in the data collection and analysis of this study. Appendix B provides documentation to one of the items used in the data collection process. An extensive survey was administered to conference participants and was given 45 days in advance to complete. The survey is the result of discussions with 20 SOA experts on how to harvest the right information needed to get a good representation for the state of analyzed SOA

projects. The survey was one of the instruments used in addition to interviews and presentations. Appendix C documents the numeric translation of survey question types where numeric translation is required. Appendix D provides a sample of the training and orientation that was provided to both IBM experts and survey participants to normalize the understanding of key important topics for the data collection process.

### **1.7 Summary**

Service oriented architecture and business agility are two timely topics that are generating a lot of interest. Business agility is generally accepted, without any challenge, as a primary benefit from building SOA solutions. This research aims at investigating the relationship between SOA deployments and attainment of business agility. Furthermore, we focus our attention on identifying the primary factors that contribute the most to attaining business agility. Moreover, we establish the need and usefulness for building a model that can predict potential attainment of business agility given the technical and process attributes of a given SOA solution.

The introduction of the right set of metrics to assess the impact of SOA solutions on achieving business agility will benefit SOA practitioners and organizations as they adopt SOA as their preferred architectural style. Moreover, the creation of a model that can predict business agility benefits based on analyzing SOA projects is useful for project managers, architects and executives to assess business agility contributions before it is too late in the development lifecycle.

## Chapter Two: Service Oriented Architecture

### 2.1 Introduction

This chapter discusses the principles and concepts of Service Oriented Architecture in order to lay the groundwork for identifying contributors to business agility. Moreover, we review some of the fundamental components of SOA and their perceived impact on business agility.

As noted earlier, SOA is defined as a “framework for integrating business processes and supporting IT infrastructure as secure, standardized components -services- that can be reused and combined to address changing business priorities” [1] . SOA solutions build on top of optimized business processes that leverage reusable services that in turn collaborate in an effort to address business issues and contribute to business agility [1] [5] [6] [7] [23] [24] [25] [26]. Well designed SOA solutions that incorporate varying business rules and use well-thought out business-aligned services are therefore able to respond to changing business events dynamically with minimal human intervention [7]. These individual services are loosely coupled, often highly focused in nature, and must often meet tight performance and reliability constraints. Service providers publish or advertise their services into a registry by providing the service interface contract, referred to as Web Service Definition Language (WSDL) document that contains all the required information about a given business service ) [9] [27] [28] [29]. The registry provides support for publishing and locating web services through an open standard discovery interface or through a proprietary interface. Service requesters or consumers find the required business service and invoke the service by using the published WSDL to bind to the published business service through the service’s supported binding protocols provided in the WSDL.

While many service consumers connect to services using the previously discussed pattern, there is an emerging desire to have services invoked through a third party in order to realize the loose coupling benefits and business agility that can be achieved by connecting to the service through a mediator. The direct connectivity to the service, while simple, does not bode well for realizing the business agility of a SOA solution due to the rigidity of interfaces that cannot change easily during runtime.

## **2.2 Business Services vs. Web Services**

So far, the terms web service and business service have been used interchangeably. However, we will distinguish between them from now on since each will have a different meaning going forward. Fiammante [7] defines **Business Service** as a: “grouping of repeatable business tasks that address the same functional domain (e.g. Process Customer Order). Business policies will be defined at this business service level, such as policies to handle variations of business services for corporate customers or individual.” Fiammante defines a **Web Service** as: “A technical representation of elements of a business service, grouping discrete business tasks together for technical management such as versioning in a technical descriptor; using as technical standards, either Web Services Description Language (WSDL) or in a more abstract fashion, Service Component Architecture (SCA).” SCA refers to a technology agnostic method of packaging SOA components to enable composite applications within the SOA programming model [2]. According to the definition, business services are the true building blocks that enable business agility since they encapsulate the knowledge of a given business process. The reuse factor of business services is also implied as part of the definition and this is how business services show their true value. The ability to play multiple roles within different business processes that collaborate to achieve different business objectives make business services documentation and discovery more of a business-driven task rather than an IT-driven one. Therefore, business services are considered the domain of business analysts while web services are considered the domain

of SOA developers [1] [2]. Both analysts and developers use the guidelines of a SOA methodology to discover business services to build a SOA solution [30].

### **2.3 SOA Lifecycle**

Our own experience and the many references in the literature [1][2][3][7][18][30][31][32][33][34] suggest that SOA is best achieved through a lifecycle process. As Figure 3 shows, the SOA lifecycle has four primary phases: Model, Assemble, Deploy and Manage.

In the Model phase, solution business requirements are gathered and later modeled into business processes. The modeling of business processes captures additional business requirements and key performance indicators (KPI) information that is useful during the simulation step of the business process. Once a business process is modeled, it is usually simulated to experiment with what-if type of scenarios to test different business situations. For example, a business process for order fulfillment may be modeled to handle one type of customer based on a set rate of transactions. An organization can simulate the introduction of additional customer types and the impact on the business process and the potential existence of bottlenecks. Simulation results are later incorporated into the process design to enhance the overall design of the process.

The Assemble phase starts out through a *discovery* step. In this step, solution designers search the existing assets for services or service components that may meet their needs before starting a new design. If assets are discovered as part of this investigation, designers will assess the suitability for reuse in the existing form or the need to modify. If no assets are found, then solution designers are free to create new assets and use those assets in the implementation of the modeled business process through the *compose* step. The primary objective of the Assemble phase is to encourage reuse among solution assets. This step is usually supported through some of the governance best practices that are required to ensure the success of SOA implementations [6].

Once assets are assembled and ready for deployment, additional activities commence to ensure the appropriateness of the deployment environment to meet the non-functional requirements of a SOA solution. This usually means making sure that infrastructure platforms are ready to run the developed SOA components from the previous step. Integration into additional people portal applications, business processes, or data systems may be fine tuned in this step.

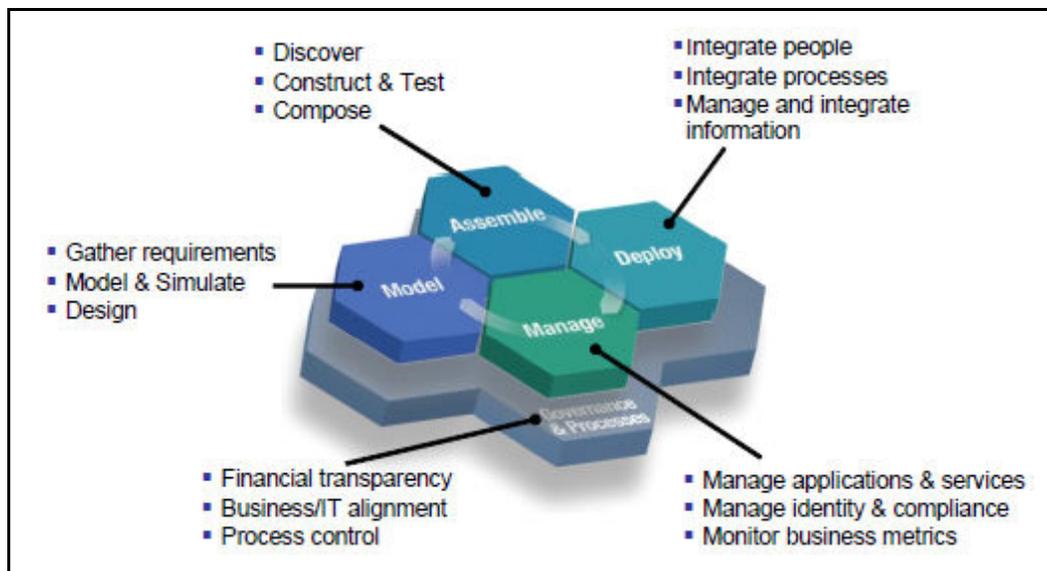


Figure 3 IBM SOA Lifecycle [34]

In the Manage phase, the deployed SOA solution is monitored for a variety of indicators. Given the additional business and service layers introduced by SOA, there are many indicators that can be monitored and measured against stipulated requirements. For example, a business process may have associated KPIs that can be monitored at the business level to ensure those KPIs are respected. Similarly, services may have associated SLAs that must be respected during runtime. Both KPIs and SLAs are usually stored in a repository and investigated to ensure that the overall solution is meeting its performance parameters. The same is true for security and the management of identities across the layers. Given the additional layers imposed by SOA, a variety of security techniques may

be implemented to secure the system resources. This implies the need to exchange security tokens as systems communicate across many layers, and the need to audit all involved parties when resources are accessed.

The SOA lifecycle encourages the communication of the feedback from the Manage phase into the Model phase so that performance issues that may arise from process or service design can be fixed. Moreover, potential bottlenecks or processes that may be executing at or near peak capacity may indicate that the deployed infrastructure is undersized and tweaks may be required to ensure the performance of such processes. The emphasis on infrastructure and its potential to derail rolling out new business challenges has been documented and is a recurring challenge [35] [36]. Therefore, extra attention must be given to ensure IT infrastructure alignment, monitoring, and governance is addressed properly. It is worth noting that SOA design methodologies are iterative in nature, therefore the SOA lifecycle may be repeated many times across different iterations until all requirements are implemented.

To ensure the proper execution of SOA implementation, the SOA lifecycle is predicated on a set of governance processes that govern many aspect of the SOA implementation. Governance practices are required to ensure the alignment between business and IT through the introduction of processes that govern all development aspects of services from inception through sunset. Governance practices also provide guidelines for initiating a SOA center of excellence and how to staff it with the right roles, processes and skills to ensure the successful implementation of SOA [6] [37].

## **2.4 SOA Governance**

SOA Governance is considered one of the major contributors to business agility [6] [7] [38]. Figure 3 shows the SOA lifecycle which is grounded on a set of best practices and a SOA governance lifecycle. Governance in general is defined by Brown [6] through two major activities:

- Establish chains of responsibility, authority and communication to empower people (decision rights).
- Establish measurement, policy and control mechanisms to enable people to carry out their roles and responsibilities.

Extending this definition to IT, Brown [6] defines IT Governance by applying the previous two steps to decisions associated with IT and how IT decisions are made and monitored. Therefore, SOA governance becomes an extension of IT governance that is focused on the *lifecycle of services* to ensure the business value of SOA [6].

Similar to SOA, SOA governance has its own lifecycle that is distinct from the assets that are being governed. The SOA governance lifecycle is characterized as a four-stage process [6] [38]:

1. **Plan** phase, during which the need for governance is established and the existing mechanisms are assessed.

The Plan phase is aimed at understanding the existing as-is nature of governance at the organization. This includes activities that address maturity of the current governance practices to identify gaps. Once the current understanding is formulated, a vision is developed to address future needs as it pertains to SOA governance.

2. **Define** phase, during which the desired governance framework, including new and modified principles, processes, organizational structures and roles are established.

The define phase includes a set of activities that when completed will enable building the new SOA governance model for the organization. This phase will take input from information discovered and understood about the organization's existing environment, principles, policies, procedures, goals and vision from the previous Plan phase. The understanding and incorporation of the previous phase output is used in this phase to build the processes, capabilities, key performance indicators (KPIs) and mechanisms recommended. Modifications are made to the identified reusable

governance assets and where necessary additions are put in place to support SOA rollout.

3. **Enable** phase, where the new governance framework is introduced into the enterprise.

During the Enable phase the defined framework is rolled out to the organization. Roles are assigned, staff is trained, the decision rights may be automated in workflow tools, and the metrics collection and reporting mechanisms are put in place.

4. **Measure** phase, during which the metrics are gathered and analyzed to refine the SOA governance process.

The Measure phase is continually being performed. It is where the metrics identified in the Define phase get captured and reported. This phase will allow stakeholders to determine if the services and the governance model are being used effectively. Given the iterative nature of this step, services and processes within the SOA governance model can be added, modified or removed based on their performance.

## **2.5 SOA Methodologies**

SOA methodologies are designed to increase agility. SOA development methods such as IBM's Service Oriented Modeling and Architecture (SOMA) [2], define key techniques and provide prescriptive tasks and detailed normative guidance for analysis, design, implementation, testing, and deployment of services, components, flows, information, and policies needed to successfully design and build a robust and reusable SOA solution in an enterprise.

SOMA activities are incremental in nature and can be embedded in any incremental development methodology such as Rational Unified Process (RUP) [17] or the Unified Software Development Process (USDP) [39]. Figure 4 shows SOMA phases and the main activities per phase. SOMA emphasizes the role of SOA Governance as a key

enabler for ensuring the success of SOA solutions and the realization of the potential ROI from such solutions [1] [2]. Typically in SOA approaches, the business modeling step precedes the rest of activities in building a SOA solution, especially if a top-down approach is followed. During this step, business processes are modeled using any of the available modeling tools or using business process reengineering methods. The outcome of the modeling step is an optimized business process that may contain sub-processes that collaborate to realize the overall business process. In SOA solutions, the sub-processes may map into one or more services that collaborate to achieve the objectives of a given sub-process. SOMA and other SOA methodologies provide the right methods and techniques to realize business processes into SOA solutions.

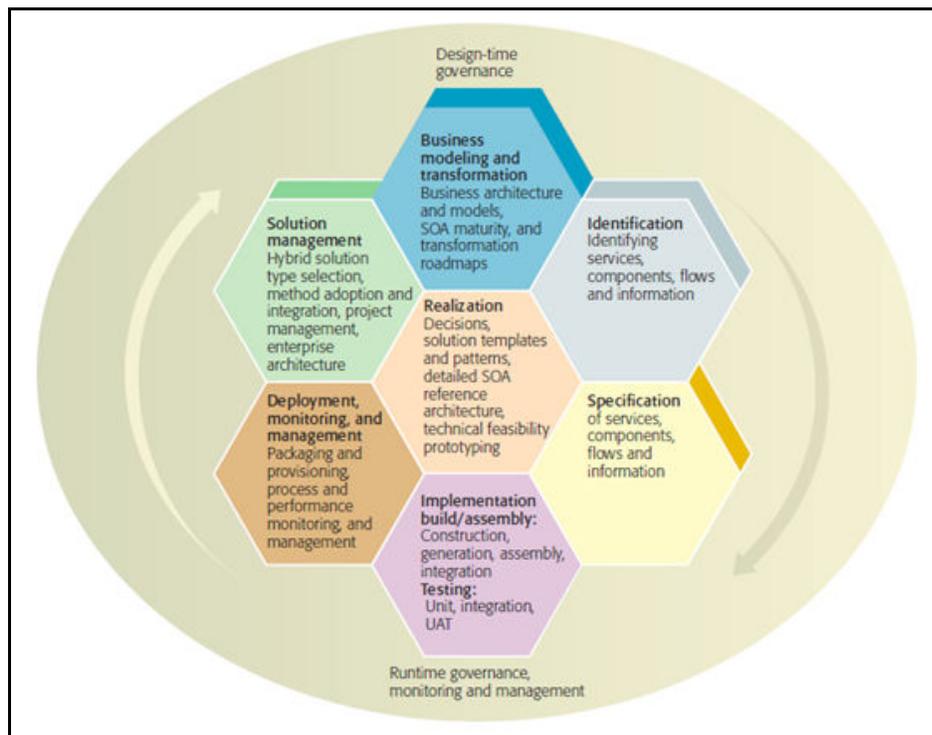


Figure 4 SOMA Phases - An IBM Image [2]

One of the key steps in any given SOA solution implementation is the identification of proper business services and their granularity. To that effect, SOMA provides different approaches to address this question through the Service Identification step. This step is carried out through the following activities: domain decomposition, existing asset analysis and Goal Service Modeling (GSM) [2].

Domain decomposition, sometimes referred to as a top-down approach, advocates the use of business process decomposition to be applied at the highest levels of a streamlined and optimized business process in order to reach the next level of sub-processes. The second level of sub-processes is further decomposed to reach the third level of decomposition that may identify candidate services. Further decomposition to a fourth level may be necessary in very complex processes but generally is not required.

Existing asset analysis, sometimes referred to as a bottom-up approach, calls for identifying candidate services by reusing the IT assets that are already available and have been proven in the past to be useful. Example of such assets could be a stored procedure or an enterprise Java bean (EJB) or a regular Java or C++ class. GSM sometimes referred to as meet-in-the-middle, advocates the use of business goals, objectives, and pain points to ensure the alignment of identified candidate services with business goals and objectives of an organization. Traceability between top-down, bottom-up and linkage to business goals, objectives and pain points is emphasized during GSM to ensure that identified candidate business services have the potential to be understood by business analysts and reused across different business processes. Moreover, it is an insurance step that provides some trust level that identified candidate business services are properly aligned with the business. The identification of candidate business services is the first step into identifying true business services. SOMA provides a service litmus test (SLT) that must be applied to candidate business services to determine the right business services and which ones will be exposed and implemented as a web service for later discovery and reuse across different business processes.

The drilling into service properties and attributes is referred to as Service Specification step in SOMA. This step provides guidelines on how to specify the inputs, outputs and service parameters that are required to define the contract for the service. The contract is documented using WSDL and is communicated to service consumers via different mechanisms to advertise the presence of services. This step also addresses the guidelines analyzing subsystems and refactoring and rationalizing services. During this step, an inventory of services is developed, referred to in SOMA as a service context diagram or service model. The model plays an important role later in the governance of services since it provides a complete list of services that are part of the overall solution.

Once all service properties are specified to meet the requirements, the Service Realization step provides guidelines on how to realize the actual implementation of services. This usually involves the distribution decisions that must be in place to determine how services will be accessed, and to define security policies and guidelines, governance measures, scalability, service level agreements and other runtime attributes for service execution. In summary, it addresses all the architectural decisions that need to be present to provide solution architecture for the business requirements.

Following the realization of service architecture, SOA introduces additional requirements for managing the SOA deployment environment. This is primarily the case due to the additional layers (e.g. modeled business processes and services), that require additional considerations to ensure that business processes are also adhering to their advertised SLAs as in the case with services. This introduces the concept of key performance indicators (KPIs) which business processes must adhere to during runtime. The KPIs are usually achieved through the accumulation of the underlying layers that implement the functionality of a business process; typically services and service components

Other SOA design methodologies emphasize the role of services, processes and design elements that need to be addressed as part of building SOA solutions [31] [24] [27] [40] [41] [42]. While the majority of SOA methodology providers claim a great deal of

business agility and flexibility as a result of building SOA solutions, none of the methodologies contain any mention of metrics to validate the claims of business agility benefits. Even though SOMA advocates the establishment of service metrics or key performance indicators (KPIs) per identified service during the service identification step, such metrics or KPIs do not, however, relate to the primary factors that would enable business agility as a result of building SOA solutions. Moreover, previous claims of SOA benefits have not been validated empirically to precisely identify the primary factors that contributed to the business agility benefits. Furthermore, the notion of business agility is loosely defined as the ability to change without specifying a baseline that can be used as a measure to determine whether an organization is business-agile or not.

## **2.6 Major SOA Architectural Components**

### ***2.6.1 Service Registry***

In order to achieve the promise of service ubiquity, there is a need for a standard method to advertise the presence of services, goals, objectives, contracts and performance information. Historically, service registries were originally specified using Universal Description Discovery and Integration (UDDI) [43][44]. UDDI is an XML-based registry that allowed organization to advertise their services and for consumers to locate those services in order to consume them. UDDI allowed consumers to use SOAP [45] messages to interrogate available services and return an advertised service interface (WSDL documents) for available services. The WSDL contains all the specifications required by consumers on how to invoke the services and the bindings used to allow for the invocation. Figure 5 shows a typical interaction between Service Consumers and Service Providers while leveraging the capabilities provided by a Service Registry. UDDI did not see significant success in the industry adoption and major SOA vendors offered their own solutions for registries and repositories while maintaining support for UDDI as an optional method to discover services.

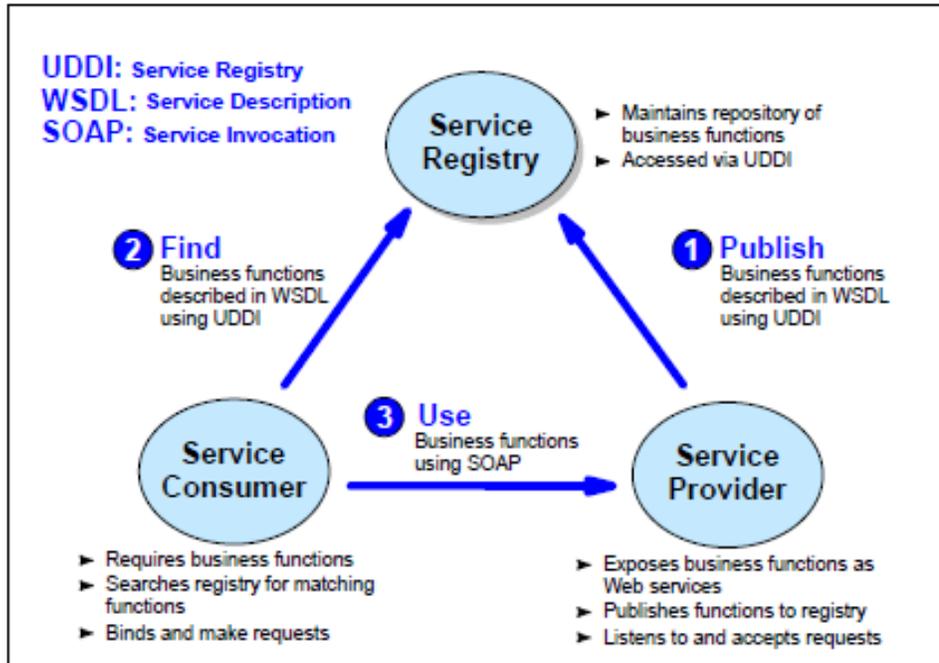


Figure 5 The UDDI Publish and Lookup Lifecycle [44]

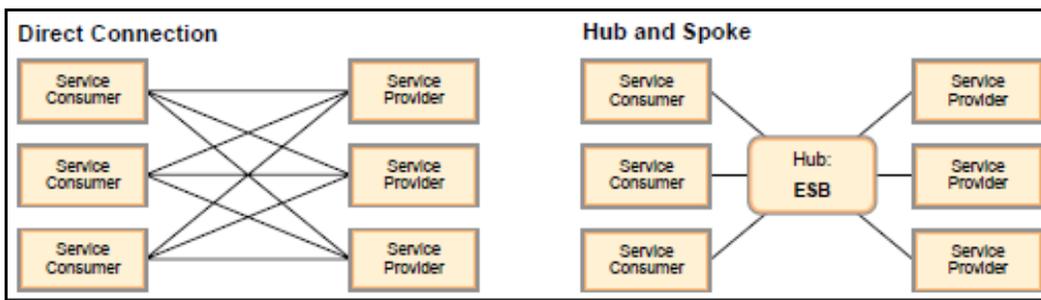
### 2.6.2 Enterprise Service Bus (ESB)

The **Enterprise Service Bus (ESB)** is defined as an architectural pattern that provides virtualization across service interactions and management features that extend SOA [46]. The ESB is generally implemented as a virtualization layer for services where service consumers put a message on the bus and are not concerned with the actual location of services. This pattern of service access is believed to be one of the factors that introduce loose coupling to SOA and help in achieving business agility.

The bus usually plays a significant role in acting on behalf of service consumers and service providers. In essence, it uses the integration pattern *Hub and Spoke* [47] [44] for the traffic from both consumers and producers. From the consumer's perspective, it eliminates the need for consumers to know significant amount of details regarding the consumed services. For example, the consumer may interface with a service that may require a different protocol or message structure that a consumer is not ready to support.

In this instance, the bus may play the role of an intermediary to fill in the missing piece and invoke the service on behalf of the consumer. Likewise, the service provider may produce an output that cannot be understood by some consumers and may depend on the ESB to translate output to acceptable formats for consumers. Therefore, the benefits of the bus are mutual for both consumers and providers [1] [7].

The ESB is often a fundamental component in any SOA design. Without a bus, service consumers and producers may start interacting using *Direct Connection* pattern that is hard to maintain as indicated in Figure 6. Moreover, the direct knowledge of service producer end points would create maintenance issues when services are decommissioned, versioned or updated. The ESB also plays a significant role in the management of services. Given the fact that all service traffic passes through the ESB, it becomes apparent that the ESB is the ideal place to have monitoring agents to monitor service traffic and determine whether services are adhering to their service level agreements (SLAs).



**Figure 6 Hub and Spoke Integration vs. Direct Connection [47]**

### **2.6.3 Business Process Management (BPM)**

In SOA, it is generally believed that business agility is achieved through the proper alignment of business and IT and derived from the flexible representation of the core business processes and business models that can be mapped to business services in order to formulate a SOA solution [1][2]. Before an organization can achieve business agility

benefits of SOA, there are some important considerations to be discussed across the business processes and horizontal optimization of such business process in what is referred to as Business Process Management (BPM). According to Association of Business Process Management Professionals (ABPMP), **BPM** is defined as: “a disciplined approach to identify, design, execute, document, monitor, control, and measure both automated and non-automated business processes to achieve consistent, targeted results consistent with an organization's strategic goals” [48]. ABPMP emphasizes the role of collaboration, technology-driven, improvement and streamlining and the end-to-end management of business process to meet business goals and objective with “more agility.”

It is generally believed that using BPM components and solutions in combination with SOA would bring the most business agility benefits as part of building SOA solutions to enable what is referred to as an agile enterprise. Fiammante [7] points to some issues that plagued some of the early adopters of SOA and BPM due to their lack of experience in using the right tools and making the right architectural decisions. For example, some companies that adopted BPM failed to realize the power of loose coupling among business processes components and the underlying building blocks, services. This in essence has led to the creation of rigid business process that did not achieve much of the perceived business agility benefits. Other issues were related to the use of rigid information models that were used in exposing services which led to the rigid nature of integrating processes and services. Most of the failures were primarily missing the variability that a truly dynamic business process can introduce into the organization by having the ability to adapt and meet varying different business needs.

### **2.7 Business Agility as an Outcome of SOA**

Many authors and vendors have claimed wide-ranging benefits of utilizing SOA, including the achievement of business agility. A quick search on Google for the words “SOA Business Agility” returned 96,200 entries as of Jan 7, 2012. A quick browse

through the top 20 entries revealed that the majority (85%) of the references agree with the assumption that business agility is an outcome of SOA and provide some advice on how to achieve that. The remaining entries either hinted to business agility through business flexibility benefits or did not mention it at all. It is generally believed that business agility is achieved through the proper alignment of business and IT and derived from the flexible representation of the core business processes and business models that can be mapped to business services in order to formulate a SOA solution [1][2].

The evidence for the relationship between SOA and business agility has received little to no attention. None of the available reviewed literature provides support on how SOA achieves business agility. While the majority of authors advocate the use of business process management and reuse of services in SOA as primary factors for achieving agility, there is no single reference that provides any evidence to show how business agility was achieved. Some authors approached the reporting of benefits from the cost, performance and better services perspective by implying reduced cost and better performance as a result of SOA. This is not necessarily correct since our own experience shows that adopting SOA is costlier than other approaches initially. Over the long run, it is true that maintenance costs can decrease if the right architectural decisions are made. The benefit of performance requires a more detailed overview before reaching conclusions and may or may not be attributed to the overall architecture of a given solution since too many parameters collaborate to achieve better performance.

## **2.8 Summary**

This chapter presented an overview of the SOA and its lifecycle that is manifested in the Model, Assemble, Deploy and Manage phases. The phases of the SOA lifecycle promote the reuse of existing SOA components through encouraging solution designers to reuse first and build afterwards. This chapter also introduced one of IBM's SOA methodologies, SOMA, and described some of the primary activities used to build a SOA solution. Finally, two of the most important architectural components of SOA were

introduced: service registry and the enterprise service bus. The service registry is a key component since it allows the advertisement of services to consumers. The enterprise service bus is equally important since it virtualizes access to services and therefore alleviates maintenance issues. The ESB also renders the integrated components loosely coupled and helps in introducing agility into the overall solution.

## **Chapter Three: Business Agility**

### **3.1 Introduction**

As noted in the previous chapter, SOA's primary focus is the delivery of flexible and resilient solutions that contribute to achieving business agility in an organization. The agile term is often associated with agile methods such as Scrum and XP. Agile methods focus on a specific set of development practices that are designed to deliver project level agility regardless of the outcome of a project. The context for the word agile or agility in this research is not related to agile development practices. This research investigates business agility as an outcome of SOA projects regardless of the development method used to deliver such projects.

The history of the word agility has its roots in the manufacturing field. The term 'agility' was introduced in this context in the early 1990's in response to continuously changing requirements that were introduced in order to become or continue to be competitive [49] [50]. The discussion of agility has led to the creation of two important terms in the manufacturing field: lean manufacturing and agile manufacturing. According to Sanchez and Nagi [49], "Lean manufacturing is a response to competitive pressures with limited resources. Agile manufacturing, on the other hand, is a response to complexity brought about by constant change. Lean is a collection of operational techniques focused on productive use of resources. Agility is an overall strategy focused on thriving in an unpredictable environment".

Lean manufacturing requires operational techniques that effectively use the resources to the maximum allowed levels while focusing on the customer. This had led to the departure from mass production to more customized versions of products and additional focus on customer and producer relationships. With additional focus on customers, newer requests and competitive threats emerged. Therefore, the need for an organization to be

agile and respond to such competitive threats became an essential requirement. According to Richards [50], “agility enables enterprises to thrive in an environment of continuous and unanticipated change”. In some texts authors use the term ‘enterprise agility’ to mean the same thing as ‘agility’. In other texts, [9][10][12] the term is referred to as ‘business agility’. The term business agility will be used in this research.

### 3.2 Business Agility

Business agility is a term that has gained significant popularity and has been defined in numerous ways. Dove [9] defines business agility as the ability to “manage and apply knowledge effectively, so that an organization has the potential to thrive in a continuously changing and unpredictable business environment”. Gartner [10] defines business agility as "the ability of an organization to sense environmental change and respond efficiently and effectively to that change". The proactive initiation of change is also considered to be a key requirement of achieving business agility [9][10][11][12]. Kidd [13] defines agility as ability of enterprise elements to adapt proactively to unexpected and unpredicted changes. All definitions do have one common attribute in common and that is the ability of an organization to adapt in response to events.

#### **Definition: Business Agility:**

Organization’s ability to effectively sense, manage, adapt and respond efficiently and effectively to produce a desired outcome to opportunities, challenges and competition, through leveraging business strategy, people and information technology.

A closely related term that is commonly used with business agility is referred to as *business flexibility*. Business flexibility is defined as the ability to respond effectively to changing circumstances [51]. The definition of business flexibility overlaps with that of business agility. Similar to business agility, there is no standard definition for business flexibility and usually definitions are dependent on the domain where it is being used.

The majority of business flexibility definitions documented in literature [52] [53] [54] focus on manufacturing, operations management and supply chains. According to [55], both business agility and flexibility are similar in what they attempt to achieve. However, business agility provides an emphasis on the speed at which new features can be deployed or existing functionality modified. For the purposes of this research we use the term business agility as it is more commonly associated with the outcomes of deploying SOA solutions.

### **3.3 Types of Business Agility**

Business agility can be described from two different perspectives. From a business perspective, business agility means the business can sense changing conditions and adapt dynamically for changing business conditions that arise either expectedly or unexpectedly. From IT perspective, business agility means how ready and adaptable are the underlying IT applications and infrastructure to meet the demand of changing business needs. In other words, how flexible the runtime architecture is in order to support the overall business agility.

Martensson [56] documented three levels of agility where a system can exhibit agility through versatility, reconfiguration and reconstruction. Versatility implies that a given system is able to cope with changes and unforeseen conditions given its current configuration state. Reconfiguration refers to the previous trait with the additional reconfiguration parameters needed to help a system cope with new changes. If the situation is severe enough that reconfiguration is not able to help a system cope with changes, then reconstruction is recommended. Martensson [56] provides a depiction of how agility can be produced and consumed through an organization. He also warns about an organization's overconsumption of agility without replenishing agility capabilities may lead to reduced overall agility.

The low level classification and distinction by Martensson is not necessary since agility will be primarily driven by business requirements. Our research has shown that users do not necessarily understand business agility or are able to quantify the levels of required business agility. Therefore, in this research we do not distinguish between the varying classifications of business agility. We take a holistic approach to business agility and report our results based on a common set of features that indicate whether business agility was achieved or not.

### **3.4 SOA and Business Agility**

The notion of business agility leads to what is known as an agile enterprise or business agile enterprise. Cummins [18] defines an agile enterprise as “an enterprise that incorporates an SOA along with BPM techniques for process design and optimization and an effective governance structure.” He further explains that agile enterprises use “model-based management tools to manage complexity, support optimization of enterprise operations, and enable rapid reconfiguration of service units to respond to changing business threats and opportunities.”

The evidence for the relationship between SOA and business agility has received little to no attention. Many authors have claimed to achieve business agility as a result of SOA implementations. For example, Ling et al. [57] argue for achieving business agility through the integration of SOA and BPM. The authors claim that business agility is achieved through the use of flexible layers with BPM as the key driver for achieving business agility. The authors provide no evidence on how is that possible other than the typical arguments of added flexibility that BPM provides to the overall solution architecture. Other studies [58] examined SOA implementation strategies and their impact on organization’s cost and process discrepancy. Furthermore, the authors focus on SOA implementation decisions and its impact on agility especially for organizations considering changing IT processes into services. Other literature[59] reported their experience of consolidating existing IT systems into SOA-based solutions and achieving

business agility in the process. Business agility was realized through reducing cost and better services that add value to their customers. None of the previously mentioned references provided any support on how SOA achieved business agility. Nor did they use any form of empirical validation to substantiate their claims.

### 3.5 Business Agility Contributors

There are many universal factors that impact business agility such as organizational structure, finance policies, resource utilization, alignment of business and IT, infrastructure, leveraging the impact of people and information, education and skills [9][11][14] [50][60][61]. Considering all the factors that contribute to business agility is outside the scope of this research. Moreover, it is very likely that new business agility contributors may emerge in the future as a result of advances in the IT field. Therefore, the discussion of business agility contributors in this section is focused on SOA solutions and their influence over attaining business agility.

To group the factors that impact business agility, we use three primary guidelines:

- **Business Agility Definition:** The definition of business agility is used to identify a set of factors that are likely to positively impact business agility. As stated earlier, business agility is defined as an organization's ability to effectively sense, manage, adapt and respond efficiently and effectively to produce a desired outcome to opportunities, challenges and competition, through leveraging business strategy, people and information technology. The primary components of the definition are the keywords: sense, manage, adapt and produce right outcomes. Given our focus on SOA solutions, we limit the scope of interpretation of such terms to the context of SOA solutions.

- **SOA Best Practices and Expert Opinion:** This guideline is used to explore SOA best practices that are consistent with the previously identified keywords from business agility definition. It also takes into consideration expert opinion given the number of SOA experts that were consulted during the phases of this research.
- **Factor Analysis Grouping:** This step is used to reduce the overall number of attributes that belong to each factor. Factor analysis allows for determining which of these attributes accounted for the differences in attained business agility.

Using the first two guidelines yields Table 1 which describes the high level business agility factors that are consistent with our definition. The business agility factors themselves will have their own attributes and are discussed later in this chapter. The third guideline is applied later in section 8.4 and produced a smaller set of attributes within the high level factors.

<b>Business Agility Factor</b>	<b>Description</b>	<b>Rationale</b>
<b>SOA Architecture</b>	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc.	Poorly architected systems fail to fulfill their potential in meeting stakeholder's objectives [62] [1] [18] [3] [6] [7].
<b>BPM</b>	BPM tracks the attributes of the business process management (BPM) component of a given SOA solution. This includes use of modeling and simulation.	The use of modeling & simulation helps drive the discovery of additional requirements that may impact business agility, identifies bottlenecks, & expose areas that are ripe for streamlining.
<b>Governance</b>	This factor tracks the level of governance in a given SOA solution.	Governed services go through a well-defined process to ensure they are aligned to an organization's business objectives, catalogued or registered, and monitored[6].
<b>Loose Coupling</b>	This factor tracks properties that inject flexibility into a given SOA solution.	Loosely coupled architectures facilitate faster change with little impact to other components.
<b>Impact Analysis</b>	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.	Underlying systems are equipped with the proper structures that can sense changes immediately and provide corrective measures to potential problems.

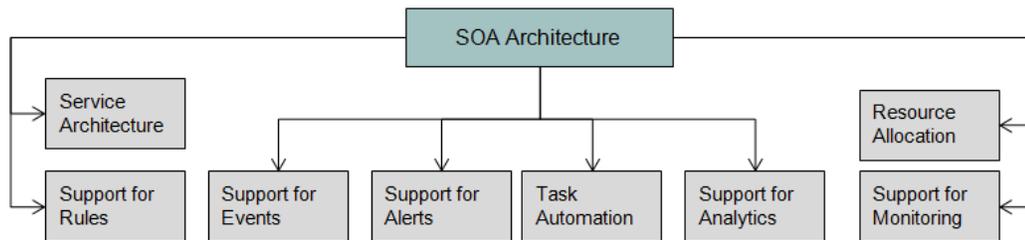
**Table 1 Business Agility Factors**

In the following sections, we explore each business agility factors and discuss their related attributes.

### **3.5.1 SOA Architecture**

The business agility factor SOA Architecture is used to capture elements of solution architecture and design that are pertinent to SOA best practices and in compliance

with the overall definition of business agility. While there are too many SOA best practices to adhere to, there are some practices that are more influential than others that provide for better overall SOA solutions' contribution to business agility. In the following paragraphs, we will describe some of the most common attributes of SOA architecture business agility factor as depicted in Figure 7:



**Figure 7 SOA Architecture Business Agility Factor and its Attributes**

- **Service Architecture**

SOA is predicated on the use of reusable business components that are called services. Any system that claims to be implementing SOA must have this property implemented and supported across the layers of the solution architecture. The concept of business services enables organizations to build new functional capabilities through the reuse of business services in different contexts to achieve new capabilities faster. The speed of creation new functional capabilities to meet market demands is one of the premises of business agility and, therefore, this property was included [18] [1] [7].

- **Support for Rules**

A business rule is defined as “a statement that defines or constrains some aspect of the business”[63]. The objective of business rules is to influence the behavior of the business and provide for better control and structure. Business rules and

their automated execution in IT systems provide a powerful means for business to add variability on how to cater to different requests and demands within the same business process. During design time of IT systems, most of potential scenarios can be worked out and the proper business rules are identified for subsequent design and realization in IT systems[7][34]. The need to adapt and produce new outcomes is important to achieving business agility and, therefore, this property was included.

- **Support for Events**

An event is “anything that happens, or is contemplated as happening” [64]. The use of events in the solution architecture and supporting of events provides resiliency to the overall solution. In addition, it enables the creation of dynamic scenarios that can be handled in the overall design without having to rebuild all the solution components. This is important to achieving business agility due to the need to anticipate new scenarios and act quickly upon them with minimal disruption to existing solutions [3] [65].

- **Task Automation**

The automation of tasks to include less human intervention provides for easier mechanisms for changing outcomes of processes easier. Coupled with the use of rules, the combination of automated tasks that leverage rules provides for a necessary combination to positively impact business agility [7]. With business agility’s focus on responding to changing market conditions, having a solution that leverages task automation would be a great benefit to have.

- **Support for Alerts**

The need to identify abnormal conditions in any solution architecture and issue alerts to correct situations is a key requirement for systems to be resilient to

potential faulty conditions. In organizations thriving to achieve business agility, it is important to have solutions that may trigger issues as soon as they happen or even before happening. This type of capability will ensure that solutions will continue to perform and adapt to potential situations that may arise due to unforeseen business conditions [66] [67].

- **Resource Allocation**

Resources allocation refers to the ability of the underlying infrastructure where solutions are deployed to achieve goals by allocating resources among the various applications to ensure optimal performance [68]. This capability is important from business agility perspective since it enables organizations to meet unanticipated demand and guarantee levels of services to consumers.

- **Support for Monitoring**

The importance of this property stems from the need for agile organization to be responsive and react to changing needs. For example, in the event of a system disruption due to an earthquake, monitored applications that are architected to handle such situations are easier to shift traffic to operating systems that are not impacted to ensure continuation of service [66].

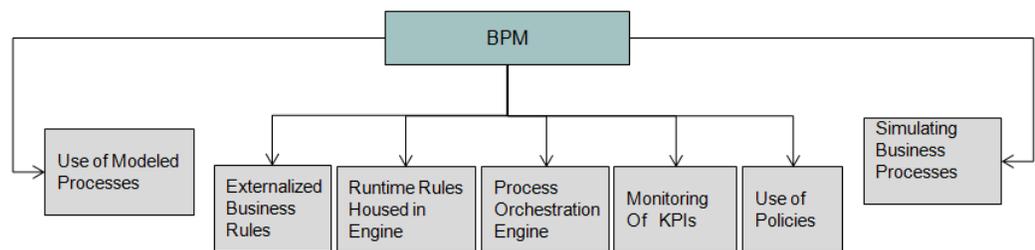
- **Support for Analytics**

Analytics is the use of sophisticated data-collection practices, methods, technology and the application of analysis on collected data to extract value from existing processes and data elements [69]. Analytics usually employs the use of statistical analysis and data mining to reach conclusions that can be used in marketing, fraud analysis, science and predictive measures. The importance of supporting analytics manifests itself through the use of the essential building blocks that support analytics such as data warehouse, information analysis,

information services and decision support systems. One of the primary benefits of analytics is the potential discovery of patterns that may not be obvious to humans based on cursory inspection of the data. For example, organizations that use analytics to predict major market trends may be better positioned to offer new services and products that cater to this potential new trend.

### 3.5.2 Business Process Management

Business Process Management (BPM) activities are considered among the important factors that contribute to business agility[7]. In this research, BPM as a business agility contributor captures the attributes of the business process management component of a given SOA solution. This includes use of modeling and simulation since modeling & simulation helps drive the discovery of additional requirements that may impact business agility, identifies bottlenecks and exposes areas that are ripe for streamlining. Figure 8 shows the attributes of the BPM business agility factor.



**Figure 8 BPM Business Agility Factor and its Attributes**

- **Use of Modeled Processes**

Modeling business processes provides a documentation of the business process. More importantly, it promotes the holistic review of a given business process for optimization purposes and streamlining through the use of simulation tools. The combination of modeled and simulated processes often result in the creation of better business processes that potentially contribute to overall business agility [1][7][18].

- **Externalized Business Rules**

The use of externalized business rules allows for adding variability into a given business process. Rules can be either hardcoded in the business process that may render the process rigid and hard to maintain. Alternatively, business rules can be externalized and manipulated outside of the business process. This allows the business process to cope with unforeseen variables for a given business scenario[1] [7] [18].

- **Runtime Rules Housed in Engine**

This attribute is an implementation procedure for the previous attribute of externalized business rules. Business rules can be externalized through different forms. One form is the use of custom built databases or files that are managed by programmers. A better alternative is the use of dedicated engines that provide a complete runtime environment for managing, maintaining, versioning and executing externalized business rules. This also facilitates the reuse of externalized business rules across many business applications [1] [7] [18].

- **Process Orchestration Engine**

Process orchestration engines provide for a mechanism for orchestrating the steps of a given business process, usually through a central workflow manager that orchestrates the flow of processes and their associated services [70]. This requirement forces the underlying services to be independent and make no assumptions about the operating conditions. It also enables the orchestration engine to change the outcome given the presence of rules and policies that will help in adding variability to business outcomes [1] [7] [18].

- **Monitoring of KPIs**

Key performance indicators represent a set of measurements that are most critical to an organizations' success [71]. KPIs can be associated with activities, processes, tasks or any type of activity that is deemed important. KPIs provide specific performance

objectives, often stated in terms of response times, throughput, latency, security, reliability, usability, accuracy, or cost. As such, KPIs can be used to evaluate whether a deployed system is currently achieving its stated business goals. Key processes typically have their associated KPIs that must be respected in order for a process to be considered successful. The monitoring of KPIs is crucial in understanding how modeled processes are executed and adhered to their anticipated KPIs. The breach of KPIs usually provides process designers with enough insight to re-model processes to eliminate potential issues that may impact KPIs negatively. Or it may lead to alternative design of business processes to achieve better overall KPIs [1] [7].

- **Use of Policies**

The term **policy** refers to “a definite course or method of action selected from among alternatives and in light of given conditions to guide and determine present and future decisions” [72]. In the context of BPM and SOA in general, the term policy may refer to different requirements that pertain to objectives or goals that need to be accomplished. For example, a security policy may require all communication to be encrypted across business layers. A service design policy may indicate that each service must have its own security policy. The use of policies is an indicator to the maturity of organizations implementing SOA given the level of sophistication associated with it. The use of policies enables systems to be more loosely coupled through the injection of dynamic behaviors that can be set in policies that may supersede business rules.

- **Simulating Business Process**

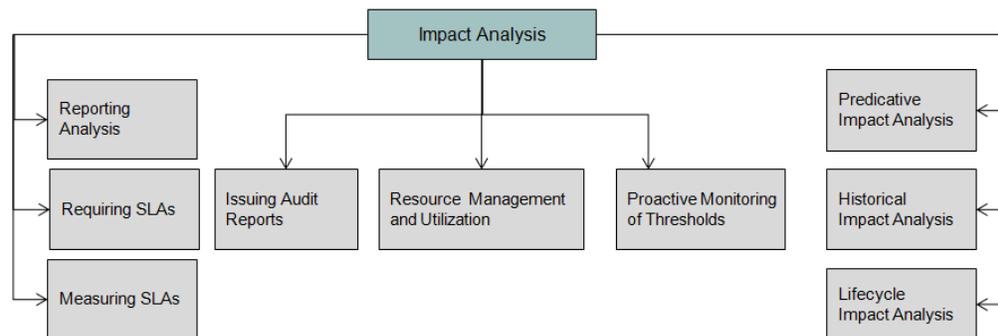
Simulation of business processes refers to executing “what if” scenarios on modeled business processes [1]. Business processes that are modeled and simulated tend to be more optimized business processes and therefore will increase the level of business agility achieved from a SOA solution. The use of modeling tools to model and simulate a business process provides organizations with an effective and easy approach to test changes to business processes and see how they will behave in real

deployments. This step will uncover many potential issues before a business process goes into development stage. The simulation capabilities will encourage business analysts to optimize their business processes. The optimization step is more likely to result in business processes that are flexible with less potential issues during execution [1] [7].

### ***3.5.3 Impact Analysis***

Impact Analysis is typically viewed from two different perspectives. The first view focuses on design of software where changes may impact design. This view is advocated by Bohner and Arnold [73] where they define Impact Analysis as "identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change". The other perspective which is provided by Pfleeger and Atlee [74] focus on the risks associated with changes and they define Impact Analysis as "the evaluation of the many risks associated with the change, including estimates of the effects on resources, effort, and schedule". In our view, we consider both aspects to be important to the success of IT projects and SOA projects are no exception to this rule. However, we introduce additional attributes that we consider important to achieving business agility; attributes that may relate to identifying and analyzing which systems that may be impacted due to a proposed change.

Such attributes are required in order to establish the capabilities required for sensing changes and enacting on the corresponding measures to address potential changes. As a result, the solution's capabilities to address dynamic changes are enhanced and overall business agility can be achieved. Figure 9 shows the Impact Analysis business agility factor and its attributes and the following paragraphs provide a brief description for their inclusion as part of this factor



**Figure 9 Impact Analysis Business Agility Factor and its Attributes**

- **Reporting analysis**

This attributes serves as an indicator to the presence of practices that lead to impact analysis. Typically, proactively analyzing reports and searching for potential issues is a sign of an organization's maturity and seriousness for identifying issues before or as soon as they happen [73].

- **Requiring SLAs**

Service level agreements serve as a success criteria for a given solution component. Components that have SLAs associated with them are monitored for signs of achieving or failing their required SLAs. Moreover, components that advertise their SLAs go through more rigorous inspections to ensure that such SLAs are respected. Requiring SLAs is part of the necessary steps of gathering the evidence to whether solution components are behaving according to their contracts. From business agility perspective, failed components are proactively identified and failure conditions are corrected immediately [73].

- **Measuring SLAs**

The use of SLAs implies the measurement of achieved SLAs during execution times. This step is necessary to make educated decisions regarding the health status of overall application [73].

- **Issuing audit reports**

Issuing and storing audit reports provides a temporal view of the performance of the solution components over various conditions.

- **Resource management and utilization**

This attribute refers to the capabilities of existing infrastructure where solutions are deployed to handle extra loads without the need of human intervention. This typically includes the use of monitoring tools to continuously monitor system loads and the existence of available policies to handle overload situations should that become necessary. Business agile organizations are able to shift resources and manage extended loads during unforeseen conditions.

- **Proactive monitoring of thresholds**

This attribute is necessary to determine proactively whether a solution component is about to reach its maximum load. Based on this measurement, certain policies can be triggered to allow the system to cope with a new situation that may help prevent a system failure or interruption of service.

- **Predictive impact analysis**

The storage of historical audit reports, SLAs adherence data and workload conditions that is coupled with analytics will enable organizations to predict patterns of failure conditions before they happen; therefore enhancing business agility of an organization.

- **Historical impact analysis**

The use of historical audit reports, SLA performance data and other attributes will enable organizations to understand better how their existing systems are meeting their stated objectives and requirements. Moreover, it allows organizations to better understand the operating environment and plan accordingly. While this view focuses primarily on past execution conditions, it provides an accurate view of the impact of

earlier architectural decisions on the execution of systems and enables users to design better systems in the future [75].

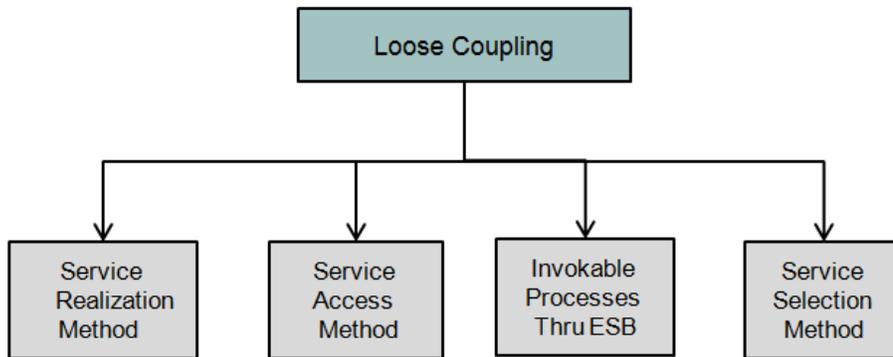
- **Lifecycle impact analysis**

This attributes refers to the solution assets, mainly services, which may need to be versioned, sunsetted, renamed or redesigned. Each service will have many consumers that may get impacted whenever a change is applied to services in production. Business agility requires that such changes are managed and their impact is well-known in advance prior to enacting changes [76].

### ***3.5.4 Loose Coupling***

The term **Loose Coupling** refers to the degree of interdependence among system components and the little or no knowledge assumed of the inner details of other components and how they achieve their stated objectives [32][77]. Loosely coupled systems are believed to offer better structures to enhance the maintainability of the overall system. From business agility perspective, we argue that loosely coupled systems offer better business agility benefits than those that are not loosely coupled. This is attributed to the speed of reconstructing new solutions to various business conditions. Speed of reconstruction stems from the independence that is a property of designing good services and processes that make no assumptions about other services. Furthermore, the integration mechanism in SOA solutions through the enterprise service bus offers significant loose coupling benefits.

For the purposes of this research, we focus on overall loose coupling across the layers of the SOA solution architecture as manifested in the presence of the virtualization layer, i.e. the enterprise service bus. Figure 10 shows the attributes of the Loose Coupling business agility factor and the following paragraphs provide a brief description for their inclusion as part of this factor.



**Figure 10 Loose Coupling Business Agility Factor and its Attributes**

- **Service Realization Method**

There are many service realization patterns that can be used for exposing and using services including the two primary patterns of **Direct Exposure (DE)** and **Indirect Exposure (IE)**. DE refers to exposing current IT systems or modules as a service without having to go through an intermediary component. For example, a stored SQL procedure could be turned into an information service directly by wrapping it through a web service and exposing the web service to consuming clients. Indirect Exposure, on the other hand, refers to exposing current IT systems or a module as a service by going through an intermediary component such as an EJB. Direct Exposure services provide a much faster method for creating and invoking services. They also require less time to decide on appropriate interfaces since they tend to match the interfaces that can be exposed from the legacy asset. Direct Exposure services also require less time to develop and test due to the direct connectivity with the backend system. In comparison, Indirect Exposure realization of services entails additional IT components to mediate between a service and an IT asset. While this provides additional flexibility to the overall SOA solution, it also increases the time to build and test such services, and requires additional management and monitoring steps to ensure services and their associated components are functioning properly. From

business agility perspective, indirect exposure services provide for a better overall agility due to the loose coupling nature of the implementation method.

- **Service Access Method (SAM)**

Services can be accessed directly by an invoking client or through a broker component, referred to as an Enterprise Service Bus (ESB) which looks up the address of required services through a registry component, retrieves the Web Service Definition Language (WSDL) file, and then binds to that service during the invocation process. The ESB in essence provides a virtualization layer so that invoking clients do not need to know individual physical addresses of services. The ESB is responsible for routing and translating requests and responses among service requestors and service providers. The invocation of services also plays a role in the level of complexity associated with this metric. Services that are invoked directly are considered point to point connections and are harder to maintain. On the other hand, services invoked through an ESB are easier to maintain but more complex to setup, because adding an ESB component to the overall SOA solution is not a simple task. It requires proper planning and design of the ESB and interacting services.

From business agility perspective, invoking services through the ESB enhances the loose coupling of services and processes. It also shields invokers from knowing too many details about service location and deployment parameters.

- **Invokable Processes thru ESB**

Business processes can be executed through a process engine or can be deployed through service interfaces to act as services. As indicated earlier, the use of an ESB to invoke services provides a virtualization layer for services and enhances the potential for overall agility. Likewise, invoking processes as services would make process available to be combined more readily in new business scenarios and allow organizations to adapt faster.

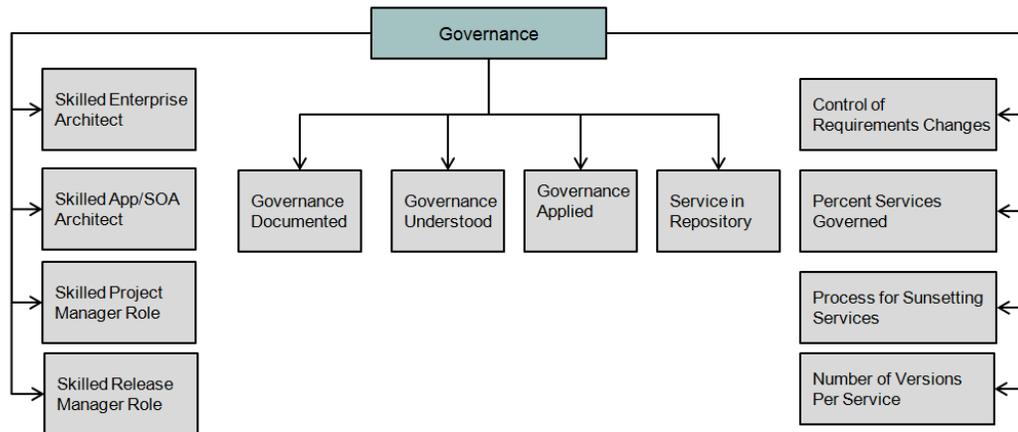
- **Service Selection Method**

Service consumers can invoke services through the typical stub and tie classes that are generated by the available tools in the market place. However, there are instances where the business logic requires a truly dynamic method for invoking the proper service based on a business rule that mandates a better service for loyal customers. Service selection method refers to the percentage of services that are selected dynamically in any given SOA solution.

The dynamic selection of services based on business rules requires additional architectural components to be present in the overall solution. The advertising of available services and their attributes becomes mandatory as well as the need to having a services broker or ESB. For this kind of scenario, the ESB and registry along with service monitoring software will play a significant role in enhancing the solution's overall flexibility and contribution to business agility.

### ***3.5.5 Governance***

According to Brown [6], "SOA Governance is an extension of IT Governance that is focused on the business and IT lifecycle of services to ensure business value". Properly governed SOA services are those services that are funded properly for an obvious business reason and tie directly to business goals and objectives. Moreover, governed services are properly advertised, managed, secured and deployed to an infrastructure that will meet execution demands. Therefore, ensuring that services are well-governed is a key attribute of well-built SOA solutions. For example, SOA Governance provides guidelines about service governance best practices. In this research, we will consider key attributes and determine their impact on business agility through the creation of a metric that tracks the measurement of the level of service governance used in a SOA project. Moreover, we extend the definition to include additional governance aspects that are important for IT projects. Figure 11 shows the attributes of the Governance business agility factor and the following paragraphs provide a brief description for each attribute.



**Figure 11 Governance Business Agility Factor and its Attributes**

- **Skilled Enterprise Architect**

Based on our own experience and observations, the availability of a skilled enterprise architect is important to governing the aspects of the solution architecture to ensure that overall requirements are met. Business agile systems tend to have their own agility requirements and therefore having an enterprise architect becomes an important governance role. Moreover, applying the concepts of enterprise architecture forces organizations to think strategically and facilitates the creation of greater flexibility into applications [18][78].

- **Skilled App/SOA Architect**

The use of SOA for building solutions requires training to think in terms of services and the application of appropriate SOA methodologies. Our own research shows that building SOA projects without having the right skilled SOA architect on the project introduces significant risks to SOA projects [1][6].

- **Skilled Project Manager role**

The project manager role is crucial in every IT project and SOA projects are no exception. The importance of the role of the project manager has been well documented in literature [6] [79].

- **Skilled Release Manager role**

The release manager role is similar in importance to the project manager role and is required in SOA projects especially given the nature of the incremental approach for implementing SOA solutions [79].

- **Governance documented**

Documenting governance practices is among the first step in establishing governance programs for organizations. Having documented governance practices is a sign of maturity for an organization and a good indicator for the success of IT projects if coupled with understanding and application of such practices [6].

- **Governance understood**

Understanding governance practices symbolizes the communication and effort organizations undertake to ensure that governance practices are well understood across the organization. This is also another sign of organization's maturity and its seriousness about applying governance practices [6].

- **Governance applied**

Once governance practices are documented and understood, it becomes easier to apply such practices. The application of governance practices is an indicator of the organizational capabilities in place to apply such practices with the right governance roles and responsibilities [6].

- **Services in repository (discovery)**

The application of SOA governance practices requires the need to govern services as part of a SOA solution. Service governance is an important aspect of overall governance practices. The use of the service registry and repository to house service

definitions provides a central mechanism for discovering services and invoking them using their advertised interfaces. The use of the repository also helps in alleviating the proliferation of duplicate services. Moreover, the service statistics provides a perspective to the use of service in production environments [6] [24] [1].

- **Control for requirements changes**

Requirements change control procedures and scope definition practices are well established best practices of successful IT projects. SOA projects are no exception and would benefit from such practices [79].

- **Process for sunseting services**

Every service has a lifecycle starting from early planning and design and ending with the sunset of a given service. Organizations who practice SOA governance must provide a process for sunseting services and when to decommission a given service from production [6].

- **Percent services governed**

This metrics measures the percentage of governed services to the overall number of services. The higher the number of governed services, the better the outcome for achieving business agility [6].

- **Number of Versions Per Service**

The number of versions available in production per service is dependent on the level of change that services undergo while they are in production. It may signal an unstable service interface in the first place and a situation where services were rushed into production. The greater the number of available versions per service within a SOA solution the more maintenance and testing it generates. Ideally, the number of versions per service in production should be minimized to avoid the maintenance issues that may result from additional testing. Moreover, the reuse

aspect of service may be negatively impacted due to the instability of service interfaces [6].

Table 2 provides a summary of all business agility factors and their associated attributes.

<b>Business Agility Factor</b>	<b>Description</b>	<b>Associated Attributes</b>
<b>SOA Architecture</b>	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc.	<ul style="list-style-type: none"> <li>1.1 Support for rules</li> <li>1.2 Support for events</li> <li>1.3 Task automation</li> <li>1.4 Support for alerts</li> <li>1.5 Resource allocation</li> <li>1.6 Support for monitoring</li> <li>1.7 Support for analytics</li> <li>1.8 Service architecture</li> </ul>
<b>BPM</b>	BPM factor the attributes of the business process management (BPM) component of a given SOA solution. This includes use of modeling and simulation.	<ul style="list-style-type: none"> <li>2.1 Use of modeled processes</li> <li>2.2 Externalized business rules</li> <li>2.3 Runtime rules housed in engine</li> <li>2.4 Process orchestration engine</li> <li>2.5 Monitoring of KPIs</li> <li>2.6 Use of policies</li> <li>2.7 Simulating business processes</li> </ul>
<b>Governance</b>	This factor tracks the level of governance in a given SOA solution.	<ul style="list-style-type: none"> <li>3.1 Skilled Enterprise architect</li> <li>3.2 Skilled App/SOA architect</li> <li>3.3 Skilled Project Manager role</li> <li>3.4 Skilled Release Manager role</li> <li>3.5 Governance documented</li> <li>3.6 Governance understood</li> <li>3.7 Governance applied</li> <li>3.8 Services in repository (discovery)</li> <li>3.9 Control for requirements changes</li> <li>3.10 Process for sunseting services</li> <li>3.11 Percent services governed</li> <li>3.12 Number of versions per service</li> </ul>
<b>Loose Coupling</b>	This factor tracks properties that inject flexibility into a given SOA solution.	<ul style="list-style-type: none"> <li>4.1 Service realization method</li> <li>4.2 Service access method</li> <li>4.3 Invokable processes thru ESB</li> <li>4.4 Service selection method</li> </ul>
<b>Impact Analysis</b>	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.	<ul style="list-style-type: none"> <li>5.1 Reporting analysis</li> <li>5.2 Requiring SLAs</li> <li>5.3 Measuring SLAs</li> <li>5.4 Issuing audit reports</li> <li>5.5 Resource management and util.</li> <li>5.6 Proactive monitoring thresholds</li> <li>5.7 Predictive impact analysis</li> <li>5.8 Historical impact analysis</li> <li>5.9 Lifecycle impact analysis</li> </ul>

**Table 2 Business Agility Factors and Associated Attributes**

### 3.6 Challenges of Measuring Business Agility

As noted earlier, the term business agility means different things and the notion of sensing and proactively adapting is a common ground for all documented definitions. Furthermore, the scope of business agility is too wide to be considered generically. Therefore, many of the approaches for measuring business agility that are documented in the literature are applicable to certain sectors and are not generic in nature. For example, the methods documented in [11][12][16][80][81] and [82] are all biased heavily towards the manufacturing sector. However, some of the concepts that are documented can be easily borrowed and adapted to apply to different sectors with the same objective of measuring agility.

There is no one universal method for measuring business agility. Some of the authors argued that given the vagueness of the concept, it is extremely hard to apply regular quantitative measurement [16][81][82]. For example, Tsourveloudis et al. [16] use fuzzy logic to build an agility measurement approach for the manufacturing sector that is based on direct measurements of operational characteristics that impact agility directly such as change in quality, versatility, product variety, rather than measuring the indirect measurement results of agility such as better profits, time to market, customer satisfaction. Moreover, the measurement method incorporates the knowledge of experts from operational systems to handle both numerical and linguistic data such as agility = 0.8 or agility is high. Furthermore, the authors include different dimensions of agility that are observable in many organizations that are labelled infrastructures such as: production, market, people and information. The authors further define the attributes that are applicable to each dimension that can be measured. The production infrastructure is measured through the ability of the enterprise to deliver on products and services. The market infrastructure is measured through the ability to continuously identify opportunities and develop products at varying levels of time to market and cost. The people infrastructure is measured through people's skills, knowledge, flatter organizational structure that enable an agile work force that can respond better to

demands. The information infrastructure is measured through the ease and speed of which data is disseminated, shared and used in the organization to reach better decisions. The infrastructure agility parameters and their variations are later considered as part of a calculation to create an overall agility value.

Other studies were also conducted using fuzzy logic to measure business agility. Lin et al. [81] [82], introduced fuzzy agility evaluation (FAE) framework that enabled the calculation of a fuzzy agility index (FAI). The FAE uses a survey to collect and analyze agility drivers such as IT integration, competence, team building, technology, quality, change, partnership, market, education and welfare. The framework includes steps to analyze and synthesize the answers to the agility drivers and provide weights that will contribute to the calculation of FAI thresholds that map to different agility levels. According to Lin et al. [81], the use of fuzzy logic depends on a membership function that is required to determine the values associated with agility attributes. The creation of such a function is not objective and depends on the knowledge and experience level of the decision maker, which impacts the accuracy of the measure. Tsourveloudis et al. [16] point to a similar observation about the selection of an optimal membership function and recommend the use of fuzzy modeling based on empirical results.

Yusuf et al. [11] proposed an agility index that measures the intensity levels of agility capabilities the authors defined as part of their research. Ren et al. [83] introduced an agility measurement that was based on analytical hierarchical process methodology where the authors combined two different frameworks for measuring agility that were proposed by Yusuf et al. [11] and Goldman et al. [12] into one model for measuring business agility. Ren et al. [50] built a four-level hierarchy with the objective agility as the main node of the model. The main node branched to four different branches (enriching the customer, cooperation, mastering change and uncertainty, leveraging the impact of people) that were documented by Goldman et al. [12]. The next two levels of the hierarchy included ten decision domains at one level and 32 agility attributes as the final level of the hierarchy that were documented by Yusuf et al. [11]. Ren et al. [83]

evaluated each hierarchy level using paired comparison rankings and converted the resulting data into relative weights that contributed to final calculation of an agility score for an organization.

Other authors tried a different approach altogether for measuring agility. Arteta et al. [84] focused on the ability of an enterprise to respond to change and considered a surrogate measure for agility that was first considered by Buzacott [85]. Buzacott[85] suggested that agility is inversely proportional to complexity and therefore agile enterprises tend to have less complex structures that would enable them to react better to unanticipated changes. Arteta et al. [84] considered the complexity factor as a surrogate measure for agility and applied it to the business processes and systems complexity as predictors for agility. However, the study did not have enough data to support their claims and using complexity as a surrogate measure for agility continues to be an observation that requires further empirical validation.

The majority of previous approaches for measuring business agility were tailored to either manufacturing or supply chain domains. In most cases, authors stipulated some of the factors that were thought to impact business agility and proceeded to measure the impact of such factors on achieving business agility. Very few of the provided studies had empirical validation to them to substantiate their claims for measuring business agility. Some of them had some radical ideas to associate the level of complexity with the lack of attainment of business agility without any proof. Others had good ideas that were reused in this research which were in compliance of developing metrics using any of the known methodologies such as GQM or its variant GQM-MEDEA. Therefore, as part of this research we borrowed high level concepts from reviewed literature to develop our own understanding of business agility as it relates to SOA outcome. The use of an index to capture the results of many factors is not new. The concept is used heavily in the field of economics where there are many indices [86] that are used to track and measure every detail for economic activity such as consumer price index. The construction of BAI with

SOA and business agility in mind seemed like a natural fit to address the issue of measuring business agility as an outcome of SOA.

### 3.7 Measuring Business Agility

As noted earlier, there is no general single method for measuring business agility. The question becomes even harder when considering measuring business agility as an outcome of deploying SOA solutions. In order to accomplish this task, we wanted to establish a measurement tool to measure and compare business agility benefits across different SOA projects. Each participating project will have an associated BAI value that determines the level of achieved business agility. A similar analogy that comes to mind is the measurement of how talented a soccer player is? There is no single actual quantified value that can tell if player A is more talented than player B. However, players possess certain attributes that enable them to have different skills in the field. One attribute that may have an impact on player's skills is how good a player is in using their left foot to kick the ball, assuming the player uses their right foot most of the time. A metric about how many times a player kicked the ball in their left foot and how many kicks of those resulted or abetted in scoring field goals can be calculated. An index of a player's talent can be created based on the calculated metrics that track player's attributes, and players can be ranked accordingly relative to each other.

In the same manner, the BAI was used to compare business agility outcome of various SOA projects that were evaluated for business agility using the same index. In order to create the BAI, we executed the following steps:

- **Identify criteria for measurement index**

In this step, twenty SOA experts met over a period of eight weeks to construct the basis for a Business Agility Index (BAI) designed to measure attained business agility. These experts included a mix of IBM fellows, distinguished engineers, senior IT architects, project managers and research staff. All participants were

experienced in SOA and its best practices, and had an average IT and SOA experience of the experts was approximately 20 and 5 years respectively. The experts were divided into two independent groups, each one tasked with defining what it means for a SOA project to contribute to business agility. Each group came to conclusions to what it means to be business agile and how SOA projects contribute to business agility. All SOA experts later discussed findings from both groups until consensus was reached. Results were then documented in the form of the eight equally weighted true/false questions, referred to from now on as the BAI. The BAI questions were included in the data collection process.

- **Define measurement scales and their meaning**

Once the criteria for the measurement were defined, the scale of the measurement was defined. To define a scale, one point was assigned to each question answered with a 'yes', and the points were then summed to determine the BAI score. As a result, BAI scores were scaled from 0 to 8. A project achieving high levels of business agility was expected to achieve BAI scores in the higher range, while an organization achieving lower levels of business agility was expected to achieve lower scores. Although it is not necessary or expected that a single SOA solution would achieve all the business benefits outlined in the BAI, the experts concluded that business agile solutions would be expected to exhibit several of these benefits.

- **Validate measurement criteria**

As part of the data collection process, technical and business project leads for each of the projects were asked to assess the achieved business agility for their individual projects using the BAI questions presented in Table 3. Project leads were not aware of the BAI and the purpose behind the questions; however they were provided with an initial orientation session which included presenting our definition of business agility. In addition to the eight BAI questions, the project

leads were asked to directly assess whether their SOA deployment contributed positively to their organization's business agility. Their response was recorded as a simple Yes/No answer.

1.	As a result of deploying this SOA solution, is your organization able to achieve better business outcomes and respond faster to customer changing <i>requirements</i> and demands? (Yes, No)
2.	As a result of deploying this SOA solution, is your organization able to address potential ad-hoc business situations? (Yes, No)
3.	As a result of deploying this SOA solution, is your organization able to adapt better to dynamic business situations once they were identified? (Yes, No)
4.	As a result of deploying this SOA solution, is your organization able to create solutions that address market requests and competitors faster and more efficiently? (Yes, No)
5.	As a result of deploying this SOA solution, is your organization able to get a better view of their business conditions and react faster to events that have the potential to cause disruption of their business? (Yes, No)
6.	As a result of deploying this SOA solution, is your organization able to adapt better to changing situations and enabled efficient routing of business needs with minimal interruptions? (Yes, No)
7.	As a result of deploying this SOA solution, is your organization able to view trusted and useful data? (Yes, No)
8.	As a result of deploying this SOA solution, is your organization able is able to create services faster to the marketplace? (Yes, No)

**Table 3 Business Agility Index Questions**

Knowing BAI values would help engineers to proactively plan and enhance their projects' business agility outcome. Furthermore, an investigation of the architectural decisions and components that stand behind the business agility contributors would force architects to consider impact of architectural decisions on business agility. Finally, business agility contributors are grounded on SOA best practices which are likely to help projects to achieve better overall results. It should also be noted that the BAI index can only be computed after-the-fact, once the system has been deployed. Section 8.6 provides a deeper insight into research results and how the BAI was used in the analysis of project data.

### **3.8 Summary**

This chapter presented an overview of the origins of the word agility and what it means to be agile. The variety of definitions for business agility agree on the need for organizations to continuously sense and adapt for various conditions in order to stay competitive. In SOA, business and IT alignment is considered among the key factors for promoting the development of business agility. The alignment of business and IT and derived from the flexible representation of the core business processes and business models that can be mapped to business services in order to formulate a SOA solution. We also presented some of the challenges that are associated with measuring business agility in the context of manufacturing and supply chain. We also discussed our own method for measuring business agility as it relates to SOA and the results of SOA deployments.

## **Chapter Four: Software Metrics Overview**

### **4.1 Introduction**

Software metrics have been and continue to be an active area of research in the software engineering field. One of the primary objectives of this dissertation is to propose and validate a new set of metrics that are specific to attaining business agility as a result of building solutions using service oriented architectures. It is important first to discuss the pitfalls and challenges that affect metric design. The following sections therefore discuss the previously stated issues and how this research is addressing these pitfalls in the construction of our set of business agility metrics.

### **4.2 Software Metrics**

A metric is defined as a measurement function or a standard for measurement [87]. Indicators are the results of applying evaluation and assessment knowledge to the metrics in order to identify trends and insights into software projects and development processes [79]. Therefore, Software metrics are quantitative measures that enable software project managers, lead developers, architects and stakeholders to gain insights into the software process. Goodman [88] provides a more formal definition of the term software metrics as “the continuous application of measurement-based techniques to the software development process and its products to supply meaningful and timely management information, together with the use of those techniques to improve that process and its products.”

To illustrate the concept, a defect is considered an error in a program that can be used as the basis of measuring a piece of code’s conformance to its expected functional or non-functional requirements. The “defects per module” metric measures the number of found

defects for a given module of code. The “defects per module” metric data can be collected, evaluated and analyzed to reach further conclusions; for example, a module with too many defects may serve as an indicator indication of a programmer’s skill set or may highlight issues in a module’s stability.

### 4.3 Software Metrics Pitfalls

Many key critiques of software metrics research is centered around the use of imprecise definitions, wrong scales, lack of empirical validation of measurements for the traits that are being measured, and the improper use of statistical methods to analyze data in order to draw correct conclusions [4] [89] [90]. In the following sections, we review the most common software metrics pitfalls and then discuss how they are addressed in this research.

#### 4.3.1 Metrics Pitfall: Improper Measurements

The role of Measurement Theory is fundamental to the creation of metrics. Measurements, metrics and indicators are very tightly coupled. Measurement is defined as the process by which numbers or symbols are assigned to attributes of entities in such a way as to describe them according to clearly defined rules. Attributes are features or properties that belong to entities such as the number of developers in a project or number of services in an IT Solution.

There are two types of measurements: **direct** and **indirect** [4] [91] [92]. **Direct measurement** refers to measurements of attributes that are measured on their own and do not depend on the measurement of any other attribute. **Indirect measurement** refers to measurements that involve the measurement of one or more attributes and comparing different measurements. Fenton [4] also refers to indirect measures as calculated measures. For example, the software measurement “lines of code” is a direct measurement of programmer productivity, while the “quality of code” (e.g. number of defects per thousands of lines of code ) is an indirect measurement. The “lines of code” is a straightforward simple measure that can be directly gauged. However, there are other

factors that need to be measured in order to determine “quality of code” such as number of lines and number of defects. Calculations are also required to determine the number of defects per thousands of lines of code. Since direct measures carry a status of inherent validity [87], it becomes very tempting to incorrectly classify many measures as direct. For example, Kaner and Bond [87] show that the “Mean Time to Failure” (MTTF) that is usually classified as a direct measure is in fact not a direct measure, because it is impacted by many different factors. For example, the heavy reliance upon human interactions would increase the chances that reapplying the measurement may not yield the same results. Software engineering metrics are not shielded from this condition and therefore extra care will be taken to classify the proposed business agility metrics. Moreover, the accurate measurement of key attributes will enhance the predictive abilities of any metrics that will be based on such measurements.

According to Kitchenham et al, [93] and Fenton [4] [91], measurements are made for various reasons. Fenton [4] recognizes two primary objectives: assessment and prediction. *Assessment* measurements call for the accurate measurement of certain entities and characterize numerically or symbolically some specific attributes of entities. Such measurements can be used on their own or as inputs into prediction models. *Predictive* measurements are those measurements that will be used in a model to predict certain values. In general, such models will require procedures for determining model parameters and how to interpret the results. Fenton notes that the use of such prediction models will be stochastic at best and will generally yield different results whenever a different prediction procedure is used. While using measurement theory and obtaining the right measurements provide the first step in adding the required rigor to metrics, it is the validation step; however, that lends credibility to such measurements. In fact, Fenton [91] emphasizes the need to clearly specify the prediction components of any predictive model and to propose the proper hypotheses before validation experimental design begins. Fenton goes further by equating the validation step of assessment type measures with empirical validation of the measure to establish the representation condition is met for the attributes being measured.

In their critique of defect prediction and program complexity metrics, Fenton et al. [94] provide a detailed review of some of the widely used defect prediction models that are based on size and complexity and their relationship to failure. The authors critique the use of flawed and imprecise definitions by Halstead [95] and McCabe[96]. McCabe's cyclomatic complexity metric and Halstead metrics offer an imprecise and flawed definition of the defects and the difficulty in determining the seriousness of a defect and its potential to lead to a failure. In other words, not all defects are created equal and finding and removing such defects may not lead to improved overall reliability [94]. Fenton et al. also point to issues with another study [97] that attempts to predict the complexity of program based on a single number that is, according to Fenton, "deceptively appealing" but based on improper measurements and measurement scales. Other problems include the misrepresentation of relationships between identified metrics. For example, many of the proposed size and complexity models assume a direct relationship between size, complexity and defects and point to the high correlation between defects and program complexity. Fenton points out that designers of such models ignore the role of humans in introducing defects that may be a result of poor initial design introduced by novice designers and programmers.

#### ***4.3.2 Metrics Pitfall: Improper Scales***

Scales were first introduced by Stevens [98] where he proposed that measurements can be classified into four different scales: nominal, ordinal, interval, ratio and absolute. ***Nominal*** scale refers to the use of numbers or labels to identify a group or a category. For example, we can use 1 to indicate a boy or 2 to indicate a girl. There is no meaning for any arithmetic operations on nominal data even though data has numeric values. Nominal scales are sometimes referred to as categorical. ***Ordinal*** scale refers to data that has all the properties of nominal scale with the added feature of ranking values from highest to lowest. An example of ordinal data includes the classification system used for cars based on their size: compact, intermediate or full size. The criterion for ranking based on size was implied since compact cars are smaller than intermediate cars, and intermediate cars are smaller than full-size cars. It is important to note that there is no

indication of how much smaller or bigger a car is from the next level up or down. *Interval* scale refers to a level of measurement when there is a fixed numerical unit of measurement and each measure assigned is expressed as a quantity of those units. An example of interval scale is the measurement of temperature using a thermometer. There is a fixed unit of measurement which is the degree and each measure assigned represents a quantity of degrees. Additions and subtractions are meaningful for this type of scale. However, comparisons such as claiming that 90 degrees Fahrenheit is twice as hot as 45 degrees Fahrenheit, is not valid since Zero degrees does not represent a condition of no heat. Moreover, the Zero degree is represented differently on the Celsius scale. In order to use such comparisons, we will need to use the ratio scale. *Ratio* scale refers to measurements where there is a fixed unit of measure and the zero point is defined on the scale of measurement. All ratio scales must start at zero and intervals on the scale must increase in equal intervals or units [4]. Ratio comparisons are meaningful using this type of scale. For example, when we measure the number of services in a project A to be 20 and another project B to be 10 services, we can conclude that project A has twice the number of services than project B. This conclusion is valid since the implied knowledge of a zero point means the absence of the element we are measuring, i.e. number of services, and therefore we were able to make such a comparison. *Absolute* scale refers to measurements that simply count the elements of a set. This measure has only one possible measurement as in the number of services in a SOA solution. Absolute scale is sometimes considered a special case of a ratio scale and may not be mentioned in some statistics literature. Scale transformation is allowed as long as it preserves the meaning, order or ratio of the scale prior to transformation and the truth value of the statement remain invariant [99] [100].

The debate over the correct use of scales, their meanings and applicable statistical methods has continued since Stevens published his seminal paper [98]. Some authors adopt a very strict view over which applicable statistical methods are appropriate for which scales. Briand et al. [101] argue that Fenton [4] and Zuse [102] adopt a strict view of the proper statistics that can be used for the different types of scales. For example,

Stevens [98] and Zuse [102] restrict the use of parametric statistics only to scales of type interval and ratio while allowing the use of non-parametric statistics for all scale types. They recommend that if measures are not of interval or ratio scales, then the use of non-parametric statistics is advised. Briand et al. [101] argue that it is very difficult to determine scale type in software engineering.

The majority of our scales fall under the categories of ratios and absolutes and therefore warrant the use of parametric statistics. In the instance that a deviation from using proper scales is assumed in our research, it will be noted and taken into consideration during the interpretation phase to identify any impact on results. It is worth mentioning that some of our identified metrics are qualitative in nature and the scale discussion is not applicable to those metrics.

#### ***4.3.3 Metrics Pitfall: Improper Metric Validation***

The validity of software metrics will depend largely on the accuracy of measurements that represent the data [44]. The validation of measurements is a key step to ensure that metrics are in fact measuring what they claim to measure [4] [103]. Many researchers have documented detailed treatments on how to address measurements validation [4][87][90][91][93][99][104]. Fenton[99] distinguishes between two types of validations: internal and external. *Internal* validation refers to a theoretical exercise to ensure the measure is a proper numerical characterization of the property claimed to be measured. In other words, is the measure measuring the attribute it intended to measure? *External* validation refers to the emphasis on the importance of the internal attribute measured and its relationship to an observable attribute of the process or the product. For example, when measuring reliability of a program, we can define internal attributes that relate to the defects of a module and that would suffice from an internal validation perspective. However, from an external validation perspective, we must show that a defect property is related to an externally observed attribute that may be referred to as reliability. Kitchenham et al. [93] refer to internal and external validation as theoretical and

empirical validation. We will use the term empirical validation throughout the remainder of the text.

According to many researchers [4][89][90][94], some software metrics lack internal and external validity. The flaws in the experimental design are generally pointed out along with the use of improper methods to collect and analyze the data. Moreover, the interpretation of the results was not based on sound statistical grounds. Briand et al. [1] cite two studies where assumptions were not clear, measurements definitions were fuzzy and as a result, two independent studies, [52] and [53], duplicating the same experiment reached contradictory results. Schneidewind [104] proposed a metrics validation framework that had six validity criteria “each of which is keyed to a quality function, so the user of metrics can understand how a characteristic of a metric, as revealed by validation tests, can be applied to measure six validity criteria, which support the quality functions of assessment, control, and prediction”. Schneidewind [104] emphasized the need for metrics to be used together to create a predictive model as a basis for validating the metrics in question. This view was critiqued by Kitchenham et al. [93] since individual metrics’ usage can differ from the usage of an overall model that may be built on the set of metrics.

Weyuker [105] created a set of properties that were initially intended to evaluate properties of complexity metrics. The Weyuker properties are generally used to ensure that measures possess a set of desirable properties and the resulting metrics are validated on their ability to exhibit the same set of properties that were defined on the measurements. The work of Weyuker has been critiqued by Kitchenham et al. [93] since some of Weyuker’s properties (properties 5 and 6) imply a scale type. According to Kitchenham et al. [93], any attribute can be measured using many various methods and “attributes are independent of the unit used to measure them”. Therefore, implying a measurement scale as part of the definition of an attribute is invalid. Kitchenham et al. [93] also consider Weyuker’s property 9 to be “inadmissible” since it involves a relation “<” that excludes nominal scale type. It is worth noting that one of the most influential

and heavily cited OO metrics suites [106] used Weyuker's properties for their analytical analysis of OO metrics.

Other critiques of current metrics point to their lack of rigor in terms of validation and formal hypotheses that need to be verified. Fenton [91] cites a study [107] in which the authors fail to stipulate their hypotheses, the dependent variable they are attempting to predict, and the overall prediction system altogether. Fenton [91] cites similar issues with interpreting Halstead's theory of software science measures of internal program attributes that help with determining effort and size of programs [95]. However, many have interpreted Halstead metrics as complexity metrics that can predict the degree of complexity in programs. For a prediction system to be complete, it needs to provide the means for determining model variables and for interpreting results. Halstead did not provide such measures in his work [91].

#### ***4.3.4 Metrics Pitfall: Improper Use of Statistics***

Fenton et al. [94] provide a critique of the statistical methods used and the lack of rigor in adhering to well-known statistical procedures to ensure proper relationships among input variables. Furthermore, some studies are based on wrong assumptions for model fitting and model prediction, such as the case of Akiyama [108], Compton [109] and Hatton [110] where they failed to apply proper techniques to their models. As a result their models were unable to predict the number of defects in new modules unless those modules were used in the derivation of the model [94]. Fenton et al. attribute this to the lack of original data and the fact that data was totally consumed in building their original models. Other researchers removed data points unjustifiably or used averaged data instead of using raw data. For example, Fenton et al. [94] point to the issue of multicollinearity in [109] [111]. Multicollinearity happens when there is high correlation among the predictor variables. One of the key assumptions when using regression models to establish relationship between a dependent variable and independent variables is the fact that independent variables are not highly correlated. Introducing multicollinearity into models leads to incorrect correlation coefficients and unexpected coefficient signs [112].

Fenton et al. [94] and Courtney and Gustafson [113] cite additional issues related to the removal of data points and lack of justification on why such points were removed and whether they were removed before, during or after analysis. Fenton acknowledges, however, the need to remove minor data sets due to errors in recording. The issue of averaging data, instead of raw data, was pointed out as another reason for biasing the results of the study as noted in [114]. This does not necessarily mean that data transformations are not allowed. On the contrary, there are valid cases where data transformation is allowed to eliminate exponential relationships between independent variables and within the realm of statistical guidelines and best practices [112].

Other researchers refer to low quality of research data and misinterpretation of results. Briand points to the low quality of some studies in their design and analysis. For example, Briand et al. [115] cite Benalarbis and Melo's [116] incorrect interpretation of statistically insignificant results to indicate strong correlations regarding the impact of polymorphism. Other researchers used small samples to justify their results and reached the wrong conclusions. For example Aebru et al. [117] applied "shotgun" correlations on a very small sample of eight projects to assess the impact of OO design on software quality attributes.

#### **4.4 Addressing Metrics Pitfalls**

The significant amount of research, both good and controversial, on the validation of measurements has led to the creation of disciplined approaches for the definition of metrics measurements. Solid theoretical foundations, clear motives and rationales were used for any defined measurements to address the shortcomings that were outlined earlier. To this effect, researchers have introduced many methods for building software measurements and predictive models for software engineering metrics and validating the correctness of the provided metrics [87] [90] [93] [118]. Notable among them is Basili's Goal/Question/Metric (G/Q/M) method [118] for its systematic approach in reaching the proper set of metrics based on proper goals, definitions and validations of the right software engineering attributes.

The G/Q/M is not free of criticism. Hefner[119] argues that using G/Q/M top-down approach does not provide an optimal approach for reaching the most useful set of metrics since the G/Q/M approach does not have a bottom-up process that can use the raw data that may provide another view to the potential set of metrics. The G/Q/M method has been subsequently enhanced to include lessons learned from applying the method in various software projects from different domains to produce the GQM-MEDEA (GQM-Metric Definition Approach) [89]. The GQM-MEDEA templates, theoretical and empirical guidelines were used as part of this research to ensure for a rigorous approach for defining and validating the proposed metrics. The use of GQM-MEDEA is appealing since it provides a framework for asking the right questions to identify software metrics and has enough flexibility for plugging additional steps as required. For example, the method relies on a property-based software engineering measurement framework to theoretically validate metrics [120]. However, there are no restrictions, in our opinion, on what method to use to theoretically validate software measurements.

Although metrics research in software engineering had its share of flaws and critique, many independent researchers have reported positive results that empirically validated the research results that were reported by the original metrics' developers. In addition, the methods used in such studies do conform to good software measurement practices and proper validation and statistical techniques. For example, the metrics suite for object-oriented systems proposed by Chidamber and Kemerer [106] is among the most referenced suite of OO metrics among many OO metrics [121]. Many researchers in the field have studied the metrics proposed by Chidamber and Kemerer for validation purposes with mixed results. Li and Henry [122] showed that Chidamber and Kemerer's metrics can predict the frequency of changes across classes in the maintenance phase.

Moreover, Basili [123] have empirically validated Chidamber and Kemerer's Depth of Inheritance Tree (DIT) metric and showed that the larger DIT value, the easier it becomes to detect faults. It was also shown that the larger value of Number of Children (NOC), the

harder it gets to detect faults since errors will be buried down in the inheritance hierarchy. Basili reported that all Chidamber and Kemerer's measures except LCOM (Lack of Cohesion in Methods) seem to be useful for predicting class fault-proneness during high- and low-level design phases. Hitz and Montazeri [124] reached similar conclusions about Chidamber and Kemerer's LCOM metric and showed the invalidity of the LCOM metric from theoretical perspective.

#### **4.5 Acceptance of Software Metrics**

There is mixed acceptance of software metrics in the software engineering community. On the one hand, the majority of the original software metrics were based on intuition and lacked the theoretical rigor that was required to entice software project managers to use them. Moreover, the metrics lacked empirical validation, and researchers who created them inadvertently misused collected data, or did not apply proper statistical methods to reach proper conclusions. On the other hand, companies did not have properly defined software processes that were mature enough to measure the right attributes. Nor did companies have access to the right measurement tools [79]. Another factor that aggravated the situation was the use of metrics for none of their original intended purposes [125]. For example, the defects-per-lines-of-code metric was used to measure tester's productivity in many organizations in addition to its proper use.

To highlight the usefulness of metrics, some researchers borrowed concepts from other disciplines. Umarji et al. [126] identified three constructs that are borrowed from the social sciences that are applicable to software metrics acceptance: usefulness, ease of use and attitude. For a metrics program to be successful, it needs to be useful at both the organizational and practitioner's level. In addition, the metrics program should be easy to use and adapt to a working environment and practices, and should not impede normal working practices. Attitude is primarily concerned with the perception of usefulness and overall evaluation of performing metrics collection by a practitioner [126] [127]. Davis [128] has shown that attitude is a significant determinant of intention to perform a behavior. It is no secret that many of the developers and managers did not have favorable view of

the usefulness of the earlier metrics and developers complained of the extra overhead required to collect metrics data.

Pfleeger [125] reported from a study about lessons learned in building metrics program that one of the major concerns of metrics programs is convincing practitioners of the value of metrics and the potential benefits of collecting the right data to enable measurements. Many of the earlier concerns were due to the lack of automation tools in collecting metrics data [87]. Similar results for the need for automation in the collection process were also reported by literature [129] [130] since automation reduces the perceived overhead from collecting metrics data. However, it is not always possible to automate the collection process in its entirety and some practitioner's interaction may still be required.

Despite the previous issues of metrics' acceptance, the considerable progress of metrics studies, empirical validation, automation of tools and proven usefulness of metrics have resulted in wider adoption of metrics programs in industry. Much of the credit goes to researchers that demanded the adoption of proper theoretical rigor into measurement practices and proper application of statistical methods to the field of software metrics. In addition, companies started appreciating the need for mature software processes that demanded measurements and execution of processes capabilities to achieve better controls over the software development practices.

#### **4.6 Summary**

This chapter presented an overview of the current literature that spanned many primary areas covered throughout this research. Given the status of the reviewed literature to the treatment of SOA and business agility in isolation, there is an opportunity to break new ground and investigate the link between business agility and SOA. Moreover, the creation of a set of metrics and a predictive model to help identify potential business agility benefits is appealing. The review of the pitfalls of previous metrics research highlights many potential pitfalls to avoid. Furthermore, the use of best practices for defining

metrics and their measurements provide an additional rigor to the approach we use to reach further conclusions. Finally, the use of empirical validation to test our hypotheses and validate our proposed metrics ensures that our results are grounded on practical real life experiences. This will help in evaluating the value from our proposed results and help with their practical implementation in SOA projects.

## **Chapter Five: SOA and Business Agility Metrics**

### **5.1 Introduction**

Service-Oriented Architecture (SOA) is now considered a mainstream option for delivering solutions which promise business agility benefits. The development of a SOA solution represents a non-trivial investment in human resources, capital and time. It is often undertaken with the expectations that it will position the organization to respond more adeptly to changing market conditions. Unfortunately, there is currently no quantitative approach for measuring the attainment of business agility as a result of deploying SOA solutions. In this chapter, we review current SOA and business agility metrics to highlight the need for a new set of a new set of metrics that are pertinent to attaining business agility as a result of deploying SOA solutions.

### **5.2 SOA Metrics Literature Review**

Some researchers advocated the use of OO metrics since most services do use OO components and services are seen as a natural extension of OO development methods [131]. However, other researchers have shown that such claims may have some validity but do not necessarily apply generically to services [132][133][134]. Applying a selection of OO metrics may have its merits when applied to service attributes such as complexity, it does not, however, cover other attributes of services and SOA solutions. For example, services formulate a structure of interacting and collaborating entities that may be orchestrated through a business process flow. Moreover, services are truly distributed components that need to be secured, managed and governed on their own. Such attributes would be completely lost if we were to extend component metrics to services without taking additional service attributes into consideration. For example, coupling and cohesion metrics used as is from OO without any refinement are not considered a good fit for SOA [132] [133]. Perepletchikov [132] asserts that the majority of OO metrics

measure implementation-level cohesion and therefore cannot be applied to SOA directly since SOA implies a higher level stage of the solution building process.

A closer look at the most famous OO metrics suite in [106] reveals that some metrics may be applied to services with some refinement. The same conclusion was reached by Perepletchikov in [132] [133]. However, such metrics are not sufficient for measuring more global SOA qualities. For example, the inheritance concept in OO systems does not have an equivalent in services, which are the SOA equivalents for classes in OO systems. It is true that some services are based on OO classes and may be inheriting behavior and attributes from base class hierarchies. However, once a class is packaged as a service, there are certain characteristics and attributes that render that unit independent of its OO origin and therefore a new set of metrics and rules are required. Another fundamental difference among OO and SOA systems is the fact that a service is considered the distribution unit of the architecture and can participate as an independent unit of a business process. On the other hand, a class cannot participate as an independent unit and must be packaged differently in order to participate. In this sense, services are more comparable to OO modules rather than classes. Moreover, services can be advertised, discovered and invoked independent of having access to the actual service code. Classes, on the other hand, cannot be advertised or discovered. Such differences warrant the need for having SOA-specific metrics to address the new nature of services.

A few researchers have proposed various SOA metrics. Rud et al. [135] focused on the infrastructure and performance aspects of SOA solutions and identified many SOA metrics that are granular in nature. These metrics were classified into the three major areas of complexity, criticality and reliability and performance. The metrics identified a relationship between complexity of a service and amount of time required to build such a service. Moreover, Rud et al. [81] introduced three resource quality metrics that focused on performance, service versioning and reliability. Qian et al. [136] developed decoupling metrics for SOA software composition such as Average Service State Decomposition (ASSD), Average Service Persistent Dependency (ASPD) and Average

Required Service Dependency (ARSD), and used it to evaluate decoupling between service-oriented components in the service composition such as Business Process Execution Language (BPEL) [70]; a useful set of metrics that should be considered for loose coupling considerations as part of the health status of SOA solutions. Other researchers looked at the relationship between security and complexity of services. For example, Liu et al. [137] developed complexity and attack-ability metrics and showed that complexity has a negative impact on security. Their Average Service Depth metric computes the average number of dependency relationships per atomic service node, as representatives of various software capabilities within a system.

Perepletchikov et al. [132] [133], created sixteen metrics for SOA that address coupling and five metrics that address cohesion. Perepletchikov et al. focused on identifying cohesion metrics that help in the identification of system interfaces with emphasis on service granularity. They also argued for taking service granularity and cohesion as an important step of service analysis and design. The coupling metrics identified the early discovery of high levels of “intra and extra” service coupling where this may indicate the need to re-structure the system in design. Moreover, the authors investigated the impact of coupling and cohesion on predicting maintainability in service oriented design. The metrics were empirically validated using only two systems that were developed specifically to validate both coupling and cohesion metrics. Weak correlations were present and insignificant statistical results were reported due to the small sample size used in empirically validating the metrics. More substantial validation work would be required in order for the provided metrics to gain some trust.

Salasin and Madni [138] identified a set of metrics that were positioned for the various stages of the SOA development lifecycle. Early lifecycle metrics are considered “predictive metrics” that relate to SOA and are based on models of the system and will influence success in later stages. According to Salasin and Madni [138], the metrics can be used to help in identifying likely problems or opportunities at future stages of

development and to provide data to make better decisions and take corrective actions.

The authors go on to identify three major types of metrics:

- Metrics that exist at early stages of the development lifecycle. Such metrics will be leveraged primarily for estimation purposes. The metrics identified for the early stages are focused on processes and activities that need to take place in order to map a concept to business processes.
- Metrics that relate to architecture and construction stage. In this stage an organization is refining requirements, conceptual designs, and models from Early Stage activities. It is developing detailed specifications of Services/components, and is building or buying those components needed to flesh out the design, providing the ability to further refine and validate models used in the Early Stage.
- Metrics that relate to technology deployment and the issues that can be uncovered during deployment of applications.

The metrics that were provided by Salasin and Madni were based on intuition and observations from projects and were never empirically validated by their authors.

Given the importance of reuse in supporting agility, many researchers focused on metrics for evaluating loose coupling in service compositions. In [139], the authors propose a formula for evaluating coupling of services in SOA compositions. The formula proposed by the authors takes into consideration different coupling techniques such as semantic, syntactic and physical that map to the different phases of the SOA lifecycle: modeling, composition and executions phases. The approach followed by the authors in [139] provides an edge over other loose coupling metrics since it incorporates a broader view of the SOA lifecycle and makes use of information available about services in the early stages of the SOA lifecycle as evidenced in the modeling phase.

Service granularity had its share of metrics frameworks that addressed the internal structure of services and levels of granularity in services. Service granularity measures the amount of exposed functionality in services and plays a role in determining the overall SOA quality attributes [10] [140]. Many of the SOA best practices advocated the use of coarse-grained services due to their improved potential for reuse [7] [25]. Service granularity metrics that are documented in [141] focus on service design and parameter granularity. The metrics proposed are based on the number of operations per services and how similar these operations are within the context of the same service. Parameter granularity is also considered and used to evaluate the impact on the overall service granularity. The authors do not distinguish between varying parameter granularity levels. Other service granularity metrics provide means for measuring service granularity through counting the number of services and associated messages and operations [142]. Senigvongse et al. [143] considers a different approach that attempts to find a service execution path to help discovering a pattern that may lead to candidates for combining and formulating coarser-grained service capabilities.

Additional research into service granularity was taken up by Ma [134]. Ma [134] reused service coupling and cohesion metrics and added two new metrics to create a quantitative approach for evaluating service identification and showed how it can be used during the early stages of service oriented design. Hofmeister and Wirtz [144] reused size and coupling metrics and added additional metrics to calculate complexity of service oriented design and to determine if design may need to be revisited. It is worth noting that both previous papers had only simple case studies documented and the metrics were never applied at a large scale. The empirical validation was also missing in both studies.

With so many SOA metrics focusing on the fundamental building block of SOA, there are fewer metrics that focus on SOA solutions in general. In [145], the authors propose a set of metrics for selecting a project from candidate SOA projects that would be suitable for SOA pilot. The proposed framework balances benefits and risks when selecting a candidate project. The most suitable candidate project is likely to be the one that exhibits

the highest benefits and poses the least amount of risk to the organization. The authors provide a set of dimensions where a project can be evaluated against the two primary factors of benefits and risk and a project profile is created for all candidate projects. The work proposed by the authors in [145] benefited greatly from a similar research regarding SOA maturity models that were documented in [146] [147] [148].

Whether it is service granularity, coupling, complexity, infrastructure or design metrics, none of the reviewed metrics would be suitable for the purposes of this research. As noted earlier, one of the primary objectives of this research is to investigate whether business agility can be achieved as a result of building and deploying SOA solutions. The previously reviewed metrics were designed to address specific topics that are important to services and service oriented computing. While these metrics are important, the majority of reviewed metrics do not provide the needed help in answering our top research question. To investigate this, we review in the next section some of the available literature about business agility and some of the available methods for measuring business agility.

### **5.3 Business Agility Metrics Literature Review**

The majority of available literature about business agility metrics is applicable to the manufacturing domain. In fact, the term agility was first coined in 1991 to describe key practices that were considered important to manufacturing to help with expediting delivery and responding to change [49] [50]. Therefore, the metrics that are found in literature [14] [50] [149] to foretell achievement of business agility do not have a direct relationship to what we are considering as part of this research. For example, the following metrics are considered common business agility indicators and contributors in manufacturing and supply chain: lead time to market of new products, Just-In-Time (JIT), or Vendor Managed Inventory (VMI), Efficient Customer Response (ECR) and Modular Product Design (MPD). Other agility attributes such as responsiveness, cost and robustness were also documented in the manufacturing domain [11]. Even though

previously mentioned metrics are considered standard in the manufacturing and supply chain domains, they do not have direct relationship to SOA solutions.

In section 3.6, we outlined some of the challenges associated with measuring business agility. Many of the challenges stemmed from the lack of precise definition for business agility which resulted in multiple conceptions on how best to achieve business agility. Furthermore, the majority of the reviewed methods for metrics and measures for business agility were context dependent for a particular industry or domain. These challenges prompted researchers to focus on processes as a fundamental agility enabler. They observed that agile organizations tend to have flexible processes as part of their operations. Moreover, agile processes had the ability to change from an optimum state in a reversible manner while retaining the essential format of the parts that were not impacted by the change [149] [150]. This has led to developing procedures that addressed measuring business agility from process perspective. Gong et al. [55] used both qualitative and quantitative measures to measure process flexibility in the public sector domain and applied their metrics to an e-Government project. They built upon previous observations from other studies [151] [152] [153] that focused on cost, quality, time and performance as primary measures for process flexibility and agility. They expanded on this fundamental observation to create additional metrics that were applied to measure agility in an eGovernment case study. Identified metrics included throughput, response time, case-handling time, law implementation time, operational cost, law implementation cost and quality. Quality is further defined by tracking number of complaints or number of appeals per case.

Realizing that agility is not the result of a single component, Izza et al. [61] examined the role of interoperability in achieving business agility and proposed a more encompassing framework for measuring agility. Interoperability refers to the ability of two heterogeneous components to communicate and cooperate with each other regardless of differences in platforms, messaging structures, interfaces and languages [154]. The authors included Process, Organization, Information, Resource and Environment

(POIRE) as potential candidates for impacting agility and focused on exploring agility in the context of enterprise interoperability. The Process dimension focused on the business process aspect and assumed it can be measured in terms of time and cost. The Organization dimension focused on organizational elements and organizational metrics were specified. The Information dimension focused on stored information within the enterprise and metrics to address accuracy, integrity, security, actuality and accessibility of data. Resource dimension focused on used resources within an organization such as people, IT resources and organizational infrastructures. Qualitative metrics such motivation and inspiration were used in addition to training levels of employees. Finally, the Environment dimension dealt with external factors of the enterprise such as customer service and marketing feedback. Metrics as reactivity, productivity and accuracy were used to capture how good an organization is in responding to external market forces. The POIRE model, while addresses information systems domain, failed to recognize SOA as one potential promoter of business agility and therefore ignored it completely.

It is very clear that previous metrics are domain dependent and applying them across different domains requires the creation of a new set of metrics that match target domains, which reduces their usefulness and broader applicability. Therefore, cost and performance can be thought of as 'generic' agility metrics at a high level which need to be broken down to a set of more fine-grained set of metrics for any given industry. While this approach may provide useful results as indicated in [55], it does not, however, provide significant value in addressing how business agility can be measured as a result of SOA.

While many of the reviewed business agility metrics are domain specific, it does not necessarily mean that such metrics cannot be reused. In fact, we incorporate elements of cost and performance as part of our own devised method to create a business agility measurement in section 3.7. The link between this research and business agility in supply chain or manufacturing can be seen through the results that can be achieved as a result of deploying SOA solutions in such domains. For example, enhancing the lead time to

manufacture a product and market it faster than competitors would require a better alignment between business and IT within the organization. It also requires efficient IT systems that can provide insight to the decision makers of the organization to make better informed decisions through accessing the right services that integrate information across various layers of the system. Moreover, the use of flexible business processes and having resilient architectures that can respond better to change would ensure that organizations can become more business agile.

#### **5.4 Summary**

This chapter presented an overview of the limited available literature about SOA metrics and business agility as it pertains to SOA. As indicated earlier business agility is all about responding effectively and successfully to changing conditions.

Given the status of the reviewed literature to the treatment of SOA and business agility in isolation, there is an opportunity to break new ground and investigate the link between business agility and SOA. Moreover, the creation of a set of metrics and a predictive model to help with identifying potential business agility benefits is appealing. The review of the pitfalls of previous metrics research provides us with the right amount of awareness to avoid such pitfalls. Furthermore, the use of metrics definition best practices for defining the proper set of metrics and their measurements provide an additional rigor to the approach we use to reach further conclusions. Finally, the use of empirical validation to test our hypotheses and validate our proposed metrics ensures that our results are grounded on practical real life experiences. This should help in recognizing the value from our proposed results and help with their practical implementation in SOA projects for keeping an eye on potential business agility benefits.

## **Chapter Six: Research Design and Methodology**

### **6.1 Introduction**

The preceding chapters introduced a high level overview of the problem statement and the issues related to the evaluation of the main research questions. As stated earlier, business agility is considered one of the primary benefits of building SOA solutions; however, such a claim has never been rigorously examined or validated. Furthermore, claims about business agility contributors such as SOA architecture, BPM, Governance, Loose Coupling and Reuse often go unchallenged as to whether they provide the articulated business agility benefits. In this research, we examine such claims through the empirical analysis of more than 39 SOA projects that claimed to have achieved some degree of business agility.

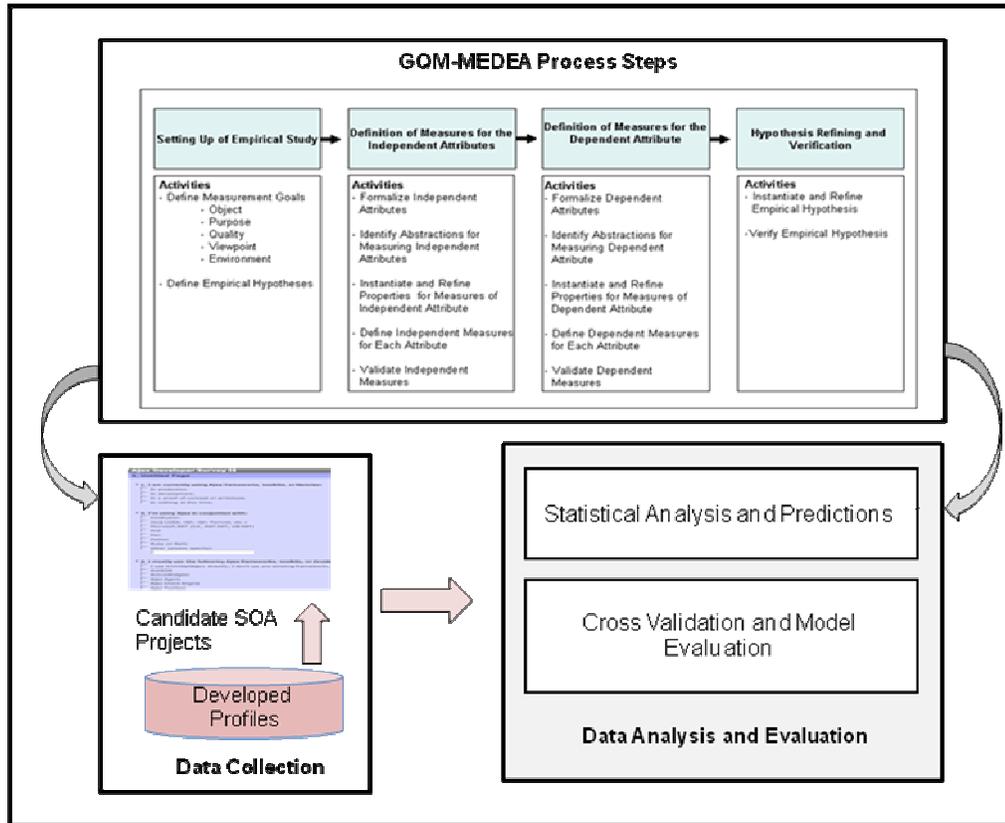
In particular, we devise a novel method for measuring business agility benefits in completed SOA projects and provide an approach to compare the attainment of business agility across various SOA projects. This is primarily accomplished through the creation of the business agility index (BAI). Using the BAI provides an objective and quantifiable method to search for factors that may have impacted the level of BAI in completed SOA projects and further identify such business agility contributors. While the BAI is able to largely differentiate between business and non-business agile projects, its usefulness is limited to SOA systems that have been built and deployed, and for which business agility has been tried and tested. The BAI therefore does not provide predictive value.

Given the significant investments in SOA projects, it is extremely useful to develop a model capable of predicting the future attainment of business agility. The model should provide users with a concrete set of factors and attributes that are easily collectable by technical project personnel during early phases of a project, and which have the capability of accurately differentiating between projects which are likely to attain

business agility and those which are not. To accomplish this we constructed the Predicted Business Agility Index (PBAI) by identifying technical factors which could be collected and measured during early phases of the project, and which were shown to be highly correlated with the attainment of business agility.

In this chapter, we describe the application of the GQM-MEDEA to accomplish the objectives of this research, Goal Question Metric: METric DEfinition Approach (GQM-MEDEA) [89] was used as the overall governing process. The approach is detailed in Briand et al. [89] and appendix A documents the primary steps of the approach . In this chapter, we illustrate how the method was applied for the purposes of this research as described in Figure 12 to help with the definition of hypotheses and metrics.

According to the GQM-MEDEA methodology, there are three primary steps that interlock to achieve the basis for collecting and analyzing the results of research. The ***GQM-MEDEA Process Steps*** that are outlined in Figure 12 offer a set of recommendations and guidelines on how to conduct activities to help with the definition of properties under investigation. It also provides a structure for the identification of entities and attributes impacting overall goals.



**Figure 12 Overall Research Methodology that Shows Application of GQM-MEDEA Process Steps to Our Research Questions – Author’s Image**

The outcome of GQM-MEDEA is a set of hypotheses and metrics that will be investigated further and empirically validated. The *Data Collection* step in Figure 12 generates the required data by the analysis step to empirically validate the hypotheses. The outcome of the Data Collection step is the definition of a project profile for participating projects in this research. In addition, an extensive data collection survey is created to meet the data demands of the empirical validation for every hypotheses stipulated in the process step. Finally, the *Data Analysis and Evaluation* in Figure 12 addresses the empirical validation of the hypotheses stipulated in the process step based on the data that was collected during the Data Collection step.

**6.2 Application of GQM-MEDEA Process Steps**

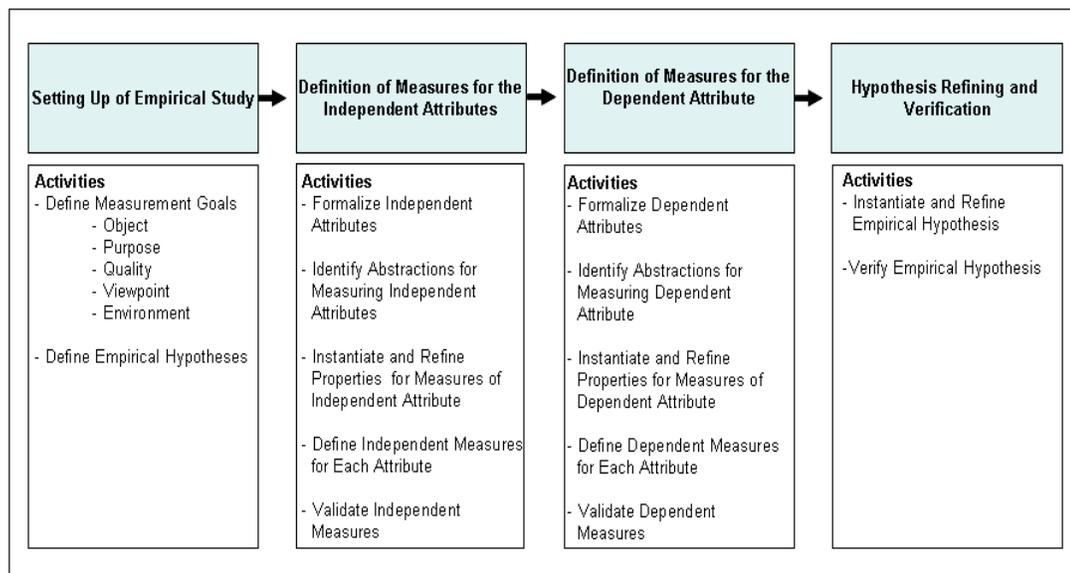
As mentioned earlier, the outcome of GQM-MEDEA is a set of hypotheses and metrics that will be investigated further and empirically validated. Figure 13 describes the major steps that are defined by GQM-MEDEA that are later explained in the following sections.

**6.2.1 Setting Up of Empirical Study**

This step has two major activities: define measurement goals and define empirical hypotheses.

- **Define Measurement Goals:**

The application of this step yields the following results documented in Table 4 which shows that business agility as the dependent variable under investigation in this research. It also shows the primary objective of investigating business agility claims and reaching a set of business agility contributors that can be used to help in predicting business agility in SOA projects.



**Figure 13 GQM-MEDEA Major Steps and Main Activities - Author's image**

<b>Goal</b>	1
<b>Object of Study</b>	Investigate existing claims about business agility as result of completed SOA projects and create of a set of business agility contributors/ metrics that can contribute to the business agility benefits that can be expected from SOA solutions.
<b>Purpose</b>	Prediction
<b>Quality Focus</b>	Business agility, dependent variable, as a result of building SOA solutions
<b>Viewpoint</b>	Project leaders and architects
<b>Environment</b>	Completed SOA projects over the past 3 years for IBM customers

**Table 4 Measurement Goals per GQM-MEDEA**

- **Define Empirical Hypotheses**

The hypotheses identified in this step are a collection of the claims that were found in many references [1] [3] [6] [7] to the effect of SOA on business agility. We group and examine such claims later in our research to identify whether there is basis for such claims and more importantly, to determine whether such factors can be used to predict the outcome of business agility. Table 5 summarizes business agility contributors from an earlier chapter. Each business agility contributor becomes the basis of a generic hypothesis that states that such a contributor positively enhances business agility in completed SOA projects. Therefore, fundamental hypothesis is similar for all the investigated contributors as part of this study. The null hypothesis  $H_0$  states that the coefficients relating our business agility contributors to the BAI are equal to zero. Or simply stated that business agility is not impacted by SOA deployment factors.  $H_0: \beta_1 = 0$ . The alternative hypothesis  $H_1$  states that the coefficients relating our business agility contributors to the BAI (dependent) variable are not equal to zero.  $H_1: \beta_1 \neq 0$ , or stated differently, that there is a relationship

between business agility and our hypothesized business agility contributors (SOA Architecture , BPM, Impact Analysis, Loose Coupling and Governance Score).

<b>Business Agility Factor</b>	<b>Description</b>	<b>Rationale</b>
<b>SOA Architecture</b>	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc.	Poorly architected systems fail to fulfill their potential in meeting stakeholder's objectives. [1] [3] [6] [7] [62]
<b>BPM</b>	BPM factor tracks the attributes of the business process management (BPM) component of a given SOA solution. This includes use of modeling and simulation.	The use of modeling & simulation helps drive the discovery of additional requirements that may impact business agility, identifies bottlenecks, & expose areas that are ripe for streamlining.
<b>Governance</b>	This factor tracks the level of governance in a given SOA solution.	Governed services go through a well-defined process to ensure they are aligned to an organization's business objectives, catalogued or registered, and monitored[6]
<b>Loose Coupling</b>	This factor tracks properties that inject flexibility into a given SOA solution.	Loosely coupled architectures facilitate faster change with little impact to other components.
<b>Impact Analysis</b>	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.	Underlying systems are equipped with the proper structures that can sense changes immediately and provide corrective measures to potential problems.

**Table 5 Business Agility Factors and Rationales for Inclusion**

### ***6.2.2 Definition of Measures of the Independent Attribute***

In the previous step, we identified our dependent variable, business agility, and identified a set of hypotheses that linked the dependent variable to a set of business agility contributors. The business agility contributors are considered the independent variables or factors for the purposes of this research. The identified hypotheses are set up to link both the dependent variable and independent variables.

At the early stages of this research, we documented about 150 SOA and architectural properties that may play a role in contributing to achieving business agility based on the set of available SOA best practices. In addition, we used our own observations and experiences learned through consulting in 15 SOA projects. However, the use of a large number of properties and factors as independent variables is not a recommended method to reach convincing conclusions due to the inter-correlation that may exist among independent variables, or collinearity, that may impact the validity of the research results [90][112].

Factor analysis can be used to address the issue of too many independent variables; and we therefore utilized it in section 8.4 to determine which of these attributes accounted for the differences in attained business agility. Dealing with a smaller set of composite factors is a better approach for statistically analyzing the data and reaching meaningful and credible results. It also removes one of the risks of construct validity, collinearity, which may impact research results. In the following text, we explain the identified set of independent variables, their potential impact on business agility and how their measures are calculated.

- **SOA Architecture:** The generic grouping factor SOA Architecture is used to capture elements of solution architecture and design that are pertinent to SOA best practices. Factor analysis was used to determine which best practices had the most significant impact on explaining the variability in attained business agility. The list of attributes is detailed in Table 6 below.
- **SOA Architecture Measurement:** In order to come with a single score for the SOA Architecture measurement, we assigned a weight for every attribute that is part of the SOA Measurement factor. In the case of this factor, all attributes were treated equally and each attribute was assigned a single point. Therefore, the maximum a project can achieve on this scale is the summation for all attributes.

**Definition:**

$$\text{SOA Score} = \sum \text{SOA}_a$$

where  $\text{SOA}_a$  is an attribute of the SOA Score

**Potential Values:**  $0 \leq \text{SOA Score} \leq 8$  (Not Normalized)

$0 \leq \text{SOA Score} \leq 8$  (Normalized)

#	SOA Architecture Factor	Reason
1-	Service architecture	SOA is predicated on the use of reusable business components that are called services. Any system that claims to be implementing SOA must have this property implemented and supported across the layers of the solution architecture. The concept of business services enables organizations to build new functional capabilities through the reuse of business services in different contexts to achieve new capabilities faster. The speed of creation new functional capabilities to meet market demands is one of the premises of business agility and, therefore, this property was included.
2-	Support for rules	A business rule is defined as “a statement that defines or constrains some aspect of the business”[63]. The objective of business rules is to influence the behavior of the business and provide for better control and structure. Business rules and their automated execution in IT systems provide a powerful means for business to add variability on how to cater to different requests and demands within the same business process. During design time of IT systems, most of potential scenarios can be worked out and the proper business rules are identified for subsequent design and realization in IT systems. The need to adapt and produce new outcomes is important to achieving business agility and, therefore, this property was included.
3-	Support for events	An event is “anything that happens, or is contemplated as happening” [64]. The use of events in the solution architecture and supporting of events provides resiliency to the overall solution. In addition, it enables the creation of dynamic scenarios that can be handled in the overall design without having to rebuild all the solution components. This is important to achieving business agility due to the need to anticipate new scenarios and act quickly upon them with minimal disruption to existing solutions [65].
4-	Task automation	The automation of tasks to include less human intervention provides for easier mechanisms for changing outcomes of processes easier. Coupled with the use of rules, the combination of automated tasks that leverage rules provides for a necessary combination to positively impact business agility. With business agility’s focus on responding to changing market conditions, having a solution that leverages task automation would be a great benefit to have.
5-	Support for alerts	The need to identify abnormal conditions in any solution architecture and issue alerts to correct situations is a key requirement for systems to be resilient to potential faulty conditions. In organizations thriving to achieve business agility, it is important to have solutions that may trigger issues as soon as they happen or even before happening. This type of capability will ensure that solutions will continue to perform and adapt to potential situations that may arise due to unforeseen business conditions.
6-	Resource allocation	Resources allocation refers to the ability of the underlying infrastructure where solutions are deployed to achieve goals by allocation resources among the various applications to ensure optimal performance[68]. This capability is important from business agility perspective since it enables organizations to meet unanticipated demand and guarantee levels of services to consumers.
7-	Support for monitoring	The importance of this property stems from the need for agile organization to be agile and react to changing needs. For example, in the event of a system disruption due to an earthquake, monitored applications that are architected to handle such situations are easier to shift traffic to operating systems to ensure continuation of service.
8-	Support for analytics	Analytics is the use of sophisticated data-collection practices, methods, technology and the application of analysis on collected data to extract value from existing processes and data elements [69]. Analytics usually employs the use of statistical analysis and data mining to reach conclusions that can be used in marketing, fraud analysis, science and predictive measures. The importance of supporting analytics manifests itself through the use of the essential building blocks that support analytics such as data warehouse, information analysis, information services and decision support systems. One of the primary benefits of analytics is the potential discovery of patterns that may not be obvious to humans based on cursory inspection of the data. For example, organizations that

**Table 6 SOA Architecture Business Agility Factor and its Attributes**

- **Business Process Management:** Business Process Management (BPM) activities are considered among the important factors that contribute to business agility [7]. In this research, BPM as a business agility contributor captures the attributes of the business process management component of a given SOA solution. This includes use of modeling and simulation since modeling & simulation helps drive the discovery of additional requirements that may impact business agility, identifies bottlenecks and exposes areas that are ripe for streamlining. Table 7 shows the attributes of the BPM factor and a brief description for their inclusion as part of this factor.
- **BPM Measurement:** In order to come with a single score for the BPM measurement, all attributes were treated equally and each attribute was assigned a single point. Therefore, the maximum a project can achieve on this scale is the summation for all attributes.

**Definition:**

$$\text{BPM Score} = \sum \text{BPM}_a$$

where  $\text{BPM}_a$  is an attribute of the BPM Score

**Potential Values:**  $0 \leq \text{BPM Score} \leq 20$  (Not Normalized)

$0 \leq \text{BPM Score} \leq 8$  (Normalized)

#	BPM Factor	Reason
1-	Use of modeled processes	Modeling business processes provides a documentation of the business process. More importantly, it promotes the holistic review of a given business process for optimization purposes and streamlining through the use of simulation tools. The combination of modeled and simulated processes often result in the creation of better business processes that potentially contribute to overall business agility.
2-	Externalized business rules	The use of externalized business rules allows for adding variability into a given business process. Rules can be either hardcoded in the business process that may render the process rigid and hard to maintain. Alternatively, business rules can be externalized and manipulated outside of the business process. This allows the business process to cope with unforeseen variables for a given business scenario.
3-	Runtime rules housed in engine	This attribute is an implementation procedure for the previous attribute of externalized business rules. Business rules can be externalized through different forms. One form is the use of custom built databases or files that are managed by programmers. A better alternative is the use of dedicated engines that provide a complete runtime environment for managing, maintaining, versioning and executing externalized business rules. This also facilitates the reuse of externalized business rules across many business applications.
4-	Process orchestration engine	Process orchestration engines provide for a mechanism for orchestrating the steps of a given business process, usually through a central workflow manager that orchestrates the flow of processes and their associated services [70]. This requirement forces the underlying services to be independent and make no assumptions about the operating conditions. It also enables the orchestration engine to change the outcome given the presence of rules and policies that will help in adding variability to business outcomes.
5-	Monitoring of KPIs	Key performance indicators represent a set of measurements that are most critical to an organizations' success [71]. KPIs provide specific performance objectives, often stated in terms of response times, throughput, latency, security, reliability, usability, accuracy, or cost. As such, KPIs can be used to evaluate whether a deployed system is currently achieving its stated business goals. Key processes typically have their associated KPIs that must be respected in order for a process to be considered successful. The monitoring of KPIs is crucial in understanding how modeled processes are executed and adhered to their anticipated KPIs. The breach of KPIs usually provides process designers with enough insight to re-mode processes to eliminate potential issues that may impact KPIs negatively. Or it may lead to alternative design of business processes to achieve better overall KPIs.
6-	Use of policies	The term policy refers to "a definite course or method of action selected from among alternatives and in light of given conditions to guide and determine present and future decisions" [72]. In the context of BPM and SOA in general, the term policy may refer to different requirements that pertain to objectives or goals that need to be accomplished. For example, a security policy may require all communication to be encrypted across business layers. A service design policy may indicate that each service must have its own security policy. The use of policies is an indicator to the maturity of organizations implementing SOA given the level of sophistication associated with it. The use of policies enables systems to be more loosely coupled through the injection of dynamic behaviors that can be set in policies that may supersede business rules.

**Table 7 Business Process Management Business Agility Factor and its Attributes**

- **Governance:** According to Brown [6], “SOA Governance is an extension of IT Governance that is focused on the business and IT lifecycle of services to ensure business value”. Properly governed SOA services are those services that are funded properly for an obvious business reason and tie directly to business goals and objectives. Moreover, governed services are properly advertised, managed, secured and deployed to an infrastructure that will meet execution demands. Therefore, ensuring that services are well-governed is a key attribute of well-built SOA solutions. For example, SOA Governance provides guidelines about service governance best practices. In this research, we will consider key attributes and determine their impact on business agility through the creation of a metric that tracks the measurement of the level of service governance used in a SOA project. Moreover, we extend the definition to include additional governance aspects that are important for IT projects. Table 8 shows the attributes of the Governance factor and a brief description for their inclusion as part of this factor.
- **Governance Measurement:** In order to come with a single score for the Governance measurement, we assigned a weight for every attribute that is part of the Governance Measurement factor. In the case of this factor, all attributes were not treated equally since some factors are more important than others. For example, understanding governance practices was assigned a single point, while applying governance practices was assigned 3 points. Similarly, the percentage of governed services was assigned one point for percentage below 30 and greater than 0, two points for percentages greater than 30 and less than 70, and three points for percentages greater than 70. No points are given for 0 percent of assigned services. Therefore, the maximum a project can achieve on this scale is the summation for all attributes.

**Definition:**

$$\text{Governance Score} = \sum Gov_a * W_a$$

where :

$Gov_a$  is an attribute of the Governance Score

$W_a$  is the weight of the attribute

**Potential Values:**  $0 \leq \text{Governance Score} \leq 18$  (Not Normalized)

$0 \leq \text{Governance Score} \leq 8$  (Normalized)

#	Governance Factor	Reason
1-	Skilled Enterprise architect	Based on our own experience and observations, the availability of a skilled enterprise architect is important to governing the aspects of the solution architecture to ensure that overall requirements are met. Business agile systems tend to have their own agility requirements and therefore having an enterprise architect becomes an important governance role.
2-	Skilled App/SOA architect	The use of SOA for building solutions requires training to think in terms of services and the application of appropriate SOA methodologies. Our own research shows that building SOA projects without having the right skilled SOA architect on the project introduces significant risks to SOA projects.
3-	Skilled Project Manager role	The project manager role is crucial in every IT project and SOA projects are no exception.
4-	Skilled Release Manager role	The release manager role is similar in importance to the project manager role and is required in SOA projects especially given the nature of the incremental approach for implementing SOA solutions.
5-	Governance documented	Documenting governance practices is among the first step in establishing governance programs for organizations. Having documented governance practices is a sign of maturity for an organization and a good indicator for the success of IT projects if coupled with understanding and application of such practices.
6-	Governance understood	Understanding governance practices symbolizes the communication and effort organizations undertake to ensure that governance practices are well understood across the organization. This is also another sign of organization's maturity and its seriousness about applying governance practices.
7-	Governance applied	Once governance practices are documented and understood, it becomes easier to apply such practices. The application of governance practices is an indicator of the organizational capabilities in place to apply such practices with the right governance roles and responsibilities.
8-	Services in repository (discovery)	The application of SOA governance practices requires the need to govern services as part of a SOA solution. Service governance is an important aspect of overall governance practices. The use of the service registry and repository to house service definitions provides a central mechanism for discovering services and invoking them using their advertised interfaces. The use of the repository also helps in alleviating the proliferation of duplicate services. Moreover, the service statistics provides a perspective to the use of service in production environments.
9-	Control for requirements changes	Requirements change control procedures and scope definition practices are well established best practices of successful IT projects. SOA projects are no exception and would benefit from such practices.
10-	Process for sunseting services	Every service has a lifecycle starting from early planning and design and ending with the sunset of a given service. Organizations who practice SOA governance must provide a process for sunseting services and when to decommission a given service from production.
11-	Percent services governed	The percentage of governed services to the overall number of services

**Table 8 Governance Business Agility Factor and its Attributes**

- **Loose Coupling:** The term Loose Coupling refers to the degree of interdependence among system components and the little or no knowledge assumed of the inner details of other components and how they achieve their stated objectives [77] [32] . Loosely coupled systems are believed to offer better structures to enhance the maintainability of the overall system. From business agility perspective, we argue that loosely coupled systems offer better business agility benefits than those that are not loosely coupled. This is attributed to the speed of reconstructing new solutions to various business conditions. Speed of reconstruction stems from the independence that is a property of designing good services and processes that make no assumptions about other services. Furthermore, the integration mechanism in SOA solutions through the enterprise service bus offers significant loose coupling benefits. For the purposes of this research, we focus on overall loose coupling across the layers of the SOA solution architecture as manifested in the presence of the virtualization layer, i.e. the enterprise service bus.

Table 9 shows the attributes of the Loose Coupling factor and a brief description for their inclusion as part of this factor.

#	Loose Coupling Factor	Reason
1-	Service realization method	<p>There are many service realization patterns that can be used for exposing and using services including the two primary patterns of Direct Exposure (DE) and Indirect Exposure (IE). DE refers to exposing current IT systems or modules as a service without having to go through an intermediary component. For example, a stored SQL procedure could be turned into an information service directly by wrapping it through a web service and exposing the web service to consuming clients. Indirect Exposure, on the other hand, refers to exposing current IT systems or a module as a service by going through an intermediary component such as an EJB. Direct Exposure services provide a much faster method for creating and invoking services. They also require less time to decide on appropriate interfaces since they tend to match the interfaces that can be exposed from the legacy asset. Direct Exposure services also require less time to develop and test due to the direct connectivity with the backend system. In comparison, Indirect Exposure realization of services entails additional IT components to mediate between a service and an IT asset. While this provides additional flexibility to the overall SOA solution, it also increases the time to build and test such services, and requires additional management and monitoring steps to ensure services and their associated components are functioning properly.</p> <p>From business agility perspective, indirect exposure services provide for a better overall agility due to the loose coupling nature of the implementation method.</p>
2-	Service access method	<p>Services can be accessed directly by an invoking client or through a broker component, referred to as an Enterprise Service Bus (ESB) which looks up the address of required services through a registry component, retrieves the Web Service Definition Language (WSDL) file, and then binds to that service during the invocation process. The ESB in essence provides a virtualization layer so that invoking clients do not need to know individual physical addresses of services. The ESB is responsible for routing and translating requests and responses among service requestors and service providers. The invocation of services also plays a role in the level of complexity associated with this metric. Services that are invoked directly are considered point to point connections and are harder to maintain. On the other hand, services invoked through an ESB are easier to maintain but more complex to setup, because adding an ESB component to the overall SOA solution is not a simple task. It requires proper planning and design of the ESB and interacting services.</p> <p>From business agility perspective, invoking services through the ESB enhances the loose coupling of services and processes. It also shields invokers from knowing too many details about service location and deployment parameters.</p>
3-	Invokable processes thru ESB	<p>Business processes can be executed through a process engine or can be deployed through service interfaces to act as services. As indicated earlier, the use of an ESB to invoke services provides a virtualization layer for services and enhances the potential for overall agility. Likewise, invoking processes as services would make process available to be combined more readily in new business scenarios and allow organizations to adapt faster.</p>

**Table 9 Loose Coupling Business Agility Factor and its Attributes**

- **Loose Coupling Measurement:** As stated earlier, given the focus on the overall solution architecture, typical interface level metrics for loose coupling and cohesion [132] [133] are not taken into consideration in this research. Therefore, for this research objective, we constructed a Loose Coupling Score that takes into consideration the solution level loose coupling properties as manifested in the enterprise service bus, how services are constructed and how processes are invoked. In order to produce a single score for the Loose Coupling measurement, we assigned equal weight for every attribute that is part of the Impact Analysis Measurement factor.

**Definition:**

$$\text{Loose Coupling Score} = \sum LC_a$$

where  $LC_a$  is an attribute of the Loose Coupling Score

**Potential Values:**  $0 \leq \text{Loose Coupling Score} \leq 14$  (Not Normalized)

$0 \leq \text{Loose Coupling Score} \leq 8$  (Normalized)

- **Impact Analysis:** Impact Analysis can be typically viewed from two different perspectives. The first view focuses on design of software where changes may impact design. This view is advocated by Bohner and Arnold [73] where they define Impact Analysis as "identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change". The other perspective which is provided by Pfleeger and Atlee [74] focus on the risks associated with changes and they define Impact Analysis as "the evaluation of the many risks associated with the change, including estimates of the effects on resources, effort, and schedule". In our view, we consider both aspects to be important to the success of IT projects and SOA projects are no exception to this rule. However, we introduce additional attributes that we consider important to achieving business agility; attributes that may relate to identifying and analyzing which systems that may be impacted due to a proposed change.

Such attributes are required in order to establish the capabilities required for sensing changes and enacting on the corresponding measures to address potential changes. As a result, the solution's capabilities to address dynamic changes are enhanced and overall business agility can be achieved. Table 10 shows the attributes of the Impact Analysis factor and a brief description for their inclusion as part of this factor.

- **Impact Analysis Measurement:** In order to produce a single score for the Impact Analysis measurement, we assigned equal weight for every attribute that is part of the Impact Analysis Measurement factor.

**Definition:**

$$\text{Impact Analysis Score} = \sum IA_a$$

where  $IA_a$  is an attribute of the Impact Analysis Score

**Potential Values:**  $0 \leq \text{Impact Analysis Score} \leq 8$  (Not Normalized)

$0 \leq \text{Impact Analysis Score} \leq 8$  (Normalized)

#	Impact Analysis Factor	Reason
1-	Reporting analysis	This attributes serves as an indicator to the presence of practices that lead to impact analysis. Typically, proactively analyzing reports and searching for potential issues is a sign of an organization's maturity and seriousness for identifying issues before or as soon as they happen.
2-	Requiring SLAs	Service level agreements serve as a success criteria for a given solution component. Components that have SLAs associated with them are monitored for signs of achieving or failing their required SLAs. Moreover, components that advertise their SLAs go through more rigorous inspections to ensure that such SLAs are respected. Requiring SLAs is part of the necessary steps of gathering the evidence to whether solution components are behaving according to their contracts. From business agility perspective, failed components are proactively identified and failure conditions are corrected immediately.
3-	Measuring SLAs	The use of SLAs implies the measurement of achieved SLAs during execution times. This step is necessary to make educated decisions regarding the health status of overall application.
4-	Issuing audit reports	Issuing and storing audit reports provides a temporal view of the performance of the solution components over various conditions.
5-	Resource management and utilization	This attribute refers to the capabilities of existing infrastructure where solutions are deployed to handle extra loads without the need of human intervention. The use of monitoring of system loads and the existence of available policies to handle overload situations. Business agile organization should be able to shift resources and manage extended loads during unforeseen conditions.
6-	Proactive monitoring of thresholds	This attribute is necessary to determine proactively whether a solution component is about to reach its maximum load.
7-	Predictive impact analysis	The storage of historical audit reports, SLAs adherence data and workload conditions that is coupled with analytics will enable organizations to predict patterns of failure conditions before they happen. Therefore enhancing business agility on an organization.
8-	Historical impact analysis	The use of historical audit reports, SLA performance data and other attributes will enable organizations to understand better how their existing systems are meeting their stated objectives and requirements. Moreover, it allows organizations to better understand the operating environment and plan accordingly. While this view focuses primarily on past execution conditions, it provides an accurate view of the impact of earlier architectural decisions on the execution of systems and enables users to design better systems in the future [75].
9-	Lifecycle impact analysis	This attributes refers to the solution assets, mainly services, which may need to be versioned, sunsetted, renamed or redesigned. Each service will have many consumers that may get impacted whenever a change is applied to services in production. Business agility requires that such changes are managed and their impact is well-known in advance prior to enacting changes [76].

**Table 10 Impact Analysis Business Agility Factor and its Attributes**

Table 11 provides a summary of business agility factors and their associated properties.

<b>Business Agility Factor</b>	<b>Description</b>	<b>Associated Attributes</b>
SOA Architecture	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc. .	1.1 Support for rules 1.2 Support for events 1.3 Task automation 1.4 Support for alerts 1.5 Resource allocation 1.6 Support for monitoring 1.7 Support for analytics 1.8 Service architecture
BPM	BPM factor tracks the attributes of the business process management (BPM) component of a given SOA solution. This includes use of modeling and simulation.	2.1 Use of modeled processes 2.2 Externalized business rules 2.3 Runtime rules housed in engine 2.4 Process orchestration engine 2.5 Monitoring of KPIs 2.6 Use of policies
Governance	This factor tracks the level of governance in a given SOA solution.	3.1 Skilled Enterprise architect 3.2 Skilled App/SOA architect 3.3 Skilled Project Manager role 3.4 Skilled Release Manager role 3.5 Governance documented 3.6 Governance understood 3.7 Governance applied 3.8 Services in repository (discovery) 3.9 Control for requirements changes 3.10 Process for sunseting services 3.11 Percent services governed
Loose Coupling	This factor tracks properties that inject flexibility into a given SOA solution.	4.1 Service realization method 4.2 Service access method 4.3 Invokable processes thru ESB
Impact Analysis	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.	5.1 Reporting analysis 5.2 Requiring SLAs 5.3 Measuring SLAs 5.4 Issuing audit reports 5.5 Resource management and util. 5.6 Proactive monitoring thresholds 5.7 Predictive impact analysis 5.8 Historical impact analysis 5.9 Lifecycle impact analysis

**Table 11 Business Agility Factors and Associated Attributes**

A quick scan of the business agility factors in Table 11 shows that independent variables' values will vary for each SOA solution depending on the requirements and business objective for every solution. Making the right architectural decisions is always a trade-off

and a balancing act among many factors that every architect need to consider in order to achieve business goals. Based on our experience and observations of real SOA projects, each of the major architectural decisions that are represented through our metrics may have the potential to impact the business agility of a SOA solution on its own. However, the combined effect of such architectural decisions or metrics is far more significant than any individual architectural decision.

### ***6.2.3 Definition of Measures of the Dependent Attribute***

The steps outlined for this level mimic those of the previous step for the independent attributes with the emphasis on the dependent attribute instead of the independent attribute. Briand et al. [101] point to potential difficulty of representing dependent attributes through mathematical measures. In such scenarios, the careful inspection of the collected data and how it was used as part of the overall process may compensate for the lack of mathematical representation. However, regardless of the mathematical representation of dependent variable, empirical validation steps were applied.

As stated earlier, one of the objectives of this research is to measure the impact of various architectural decisions and other attributes on the business agility outcome in projects that use SOA as the primary method for building solution architectures. Therefore, business agility is the dependent variable that we devise a measurement approach for to help with establishing the relationship with SOA solutions. The business agility dependent variable values are not binary. In other words, we do not consider benefits of SOA projects as either delivering business agility benefits or not. The business agility variable is more of a continuous variable that will have different values based on the outcome of SOA projects and the architectural decisions that comprise the overall SOA solution. The values can range from low business agility benefits to high business agility benefits with some midpoint to indicate medium business agility level. This classification would make the dependent variable of ordinal scale. In section 3.7, we provided a description of our own devised method to measure business agility outcome of SOA projects through the

use of the Business Agility Index (BAI) and described how it can be used to differentiate across SOA projects.

#### ***6.2.4 Hypothesis Refining and Verification***

The refinement step implies changes to our originally stipulated hypotheses. The refinement to the original hypotheses was a direct result of the early analysis of the list of initial properties of SOA solutions that may have an impact on achieving business agility. The application of this step to the research is documented in the documentation of the independent variables section 6.2.2.

### **6.3 Data Analysis and Model Building**

The data analysis and model building step focuses on the activities required to establish the proper data analysis method. Activities related to validating assumptions and appropriateness of the data are also documented in this section. Finally, the steps taken to building the predictive model are explained in this section along with the required validation steps taken to validate the results.

#### ***6.3.1 Selection of Data Analysis Method***

The selection of the data analysis technique is greatly influenced by the type of the dependent variable. For the purposes of this research, parametric statistics methods will be the primary method for investigating relationships between dependent and independent variables. Multiple regression analysis will be used to construct a model that will capture the proper relationships between one dependent variable, business agility, and multiple independent variables that represent the primary contributors to achieving business agility such as SOA architectural decisions. The selection of multiple regression requires the data to meet certain conditions, e.g. normal distribution and other factors, which must be validated prior to conducting this type of analysis. The use of too many independent variables also introduces additional challenges that must be addressed.

Exploratory factor analysis is usually recommended to address this issue and was recommended in many references such as [90] [155] [156]. DeCoster [156] actually recommends the use of Principal Component Analysis (PCA) if the objective is to perform data reduction on the independent variables to build a set of principal components that map to basic correlated measurements. Briand and Wust [155] warn that a model that relies heavily on principal components may not be transferrable across different systems. For the purposes of this research, exploratory factor analysis was performed to reduce the number of overall independent variables. Similar to multiple regressions, there are some underlying assumptions for EFA to be executed effectively; for example, the ratio of factors to the number of entries in the sample size. Validating assumptions for both multiple regression and EFA will have important ramifications during the reporting and validations of the results.

#### **6.4 Predictive Model Building**

One of the primary objectives of this research is to build a predictive model that is capable of predicting business agility benefits of SOA solutions before they are completed, referred to earlier as predicted business agility index or PBAI. This section provides a brief overview to the steps taken to create the model. Given the nature of the questions for this research and the independent variables established, we chose multiple regression analysis for building the predictive model. Multiple regression analysis requires a certain set of assumptions to be true in order to reach valid results. For example, independent samples, normal distribution, no collinearity among independent variables, etc. are all required to be validated prior to initiating this type of analysis. The execution of the steps for building the predictive model and the results are documented in section 8.7.

**6.5 Data Validation and Construct Validity**

In order to generalize predictions for business agility benefits as a result of building SOA solutions, a data set will be required to validate the prediction capabilities of the model. According to Fenton [4], this data set should not be part of the dataset that was used to build the original model. This will provide additional validity to the model to showcase its ability to predict business agility for a new dataset. In order to accomplish this, cross-validation: leave one out validation method was used in this research. In this method, the model was built using the sample size minus one data set. The resulting model that was built with n-1 data sets was evaluated to see how well it predicted the business agility benefits from the data set that was removed. The process is repeated while removing a different data set. The results of data validation are included in section 8.10.

The reliability of the empirical validation will ultimately rest on many factors that are based on experimental design process, validity of measurements and the use of appropriate statistical techniques to analyze the data and reach proper conclusions. This research addresses potential threats to validity that may stem from construct validity, internal and external validity. Construct validity refers to whether the dependent and independent variables are suitable for evaluating the hypothesis, and therefore of answering the stated research questions. Internal validity refers to the rigor with which the study is designed and executed, and the degree to which researchers have taken into account alternative explanations for observed relationships. External validity refers to the extent to which results from the study can be generalized across the entire domain of study. We address threats to validity in section 8.13.

**6.6 Summary**

This chapter provided a description of the used research methodology and the set of basic hypotheses that were investigated as part of this research. GQM-MEDEA methodology was chosen for this research due to the rigor and guidelines it provides for asking the right questions and reaching proper conclusions. Based on GQM-MEDEA, we

documented a set of hypotheses that relate SOA architectural attributes to attaining business agility. Moreover, we documented a novel method on how to measure business agility for completed SOA projects. We established criteria for the data collection process and applied statistical knowledge to select proper statistical approaches for analyzing the data to reach proper conclusions. Exploratory factor analysis and regression analysis were the primary methods used to analyze the data. Finally, we addressed our approach for cross validation, threats to validity and model interpretation.

## Chapter Seven: Data Collection Process and Application

### 7.1 Introduction

This chapter provides an overview of the data collection processes that were used to collect the primary data for this research. This research employed the use of extensive surveys that were designed with the help of numerous SOA experts to capture the right level of details about surveyed SOA projects. The next sections cover the details of the processes, project profiles, participants, collection procedures and survey overview.

### 7.2 Data Collection Process

The data collection process and application step was primarily focused on collecting the empirical data that was used during the analysis step. This activity consisted primarily of the following steps:

- **Population:** This step identified the population of SOA projects that was used for collecting project properties that are relevant to this research's main questions.
- **SOA Project Profile:** This step further defined the criteria for which projects to include from the overall population during the data collection activity.
- **Participants:** This step identified the profile of participants during the data collection process.
- **Data Collection Activity and Instrumentation:** This step identified the mechanism for collecting the empirical data from the identified participants and from the population based on the defined SOA project profile.

The following sections provide the details for each step and its associated activities:

### **7.3 Population**

The primary population for this research is the set of completed SOA projects that were completed globally through IBM and non-IBM projects. The data for this study was collected through an IBM Academy of Technology virtual conference. A call for papers was issued in early May 2010 to request participation with case studies and data collection. The first round lasted for three months from June through September of 2010, and participants were asked to evaluate factors in their project guided by a series of questions that were to be answered with respect to a single SOA project. 215 data items were collected for each project, out of which 80 were used for the purposes of this study. Data collection took an average of 12 hours per project. All projects in the study were required to have SOA as the primary architecture style and meet a project profile criteria described in section 7.4.

The majority of the architects who completed the data collection steps can be considered experts in SOA. Participants were given about two months to complete the data collection steps due to the significant amount of required details. This was also necessary since data collection participants interacted with business people on participating projects to help with filling out data collection questions.

### **7.4 SOA Project Profile**

In order for a project submission to be accepted as part of the observed data, the project had to meet the following requirements:

- Project was production-bound that had an impact on the business
- Project used service oriented architecture (SOA) as the primary architectural style to reach the solution architecture
- Project measurements were as accurate as possible and did not include “guestimates”

- Web services (JEE or .Net) were the primary building blocks for realizing business capabilities. Projects that used service-oriented methods to discover services were allowed to participate, e.g. CORBA.
- Project sizes and complexity varied. At a minimum, projects that contained less than 5 services were excluded. The number 5 is arbitrary and was used to indicate that the primary focus is those projects that leverage more business services. There were no restrictions on project duration.
- Projects that used web services that did not map properly to business services were also considered for this study. This was done in order to detect any differences among projects that used true business services as advocated by SOA as compared to projects that used web services.
- No assumptions were made regarding the success of projects or business agility results.

#### ***7.4.1 Participants***

Conference and data collection participants included IBM architects, project managers, business analysts and technical staff that have participated in the completion of SOA projects. Additional participants outside of IBM with similar roles contributed to the data collection process. The same project profile criteria were used for both IBM and non-IBM projects. The majority of conference participants were senior project managers, architects and business executives.

Participants of the conference had no previous knowledge regarding the objectives of this research as to investigate the impact of SOA projects on achieved business agility. The questions used as part of the data collection process were included as part of a regular IBM conference that was in its third iteration. The IBM conference was executed in 2008 and 2009 with the objective of collecting SOA best practices. Keeping the participants in

the dark regarding establishing links to business agility and SOA was important to ensure that participants did not feel compelled to react in one way or another to any question that was related to achieving business agility. The data collection process was totally anonymous and confidential. Participants had the option to reveal their identities if they preferred.

No incentives or monetary compensations were given to participants. All participation was voluntary and at will. The majority of participants of the survey were totally motivated IBM employees who participated in the execution of SOA projects with real IBM customers and/or internal IBM SOA projects and external non-IBM SOA projects. The association of the conference with IBM's Academy of Technology guaranteed the participation of many senior and high quality people that despite the arduous task of completing a presentation and a long data collection process. The participants were promised, however, to get a free copy of the conference report and the follow up analysis of the conference results.

#### ***7.4.2 Data Collection Activity and Instrumentation***

The majority of the sample data details were collected through an IBM-sponsored virtual conference that addressed the various factors that impacted business agility including the role of business and IT alignment and SOA. As indicated earlier, there were two rounds of data collection activity. The first round lasted for three months from June through September of 2010, and participants were asked to evaluate factors in their project guided by a series of questions that were to be answered with respect to a single SOA project. 215 data items were collected for each project, out of which 80 were used for the purposes of this study. Data collection took an average of 12 hours per project. A second round of data collection started in late January 2011 and lasted for one month. A total of 9 non-IBM projects were used for the data collection process, and participants answered only the 80 questions related to our study.

Participants answered a 215-question online survey. Appendix B shows the questions of the survey used for data collection purposes. In addition, participants submitted a case study presentation to provide information about SOA best practices used and lessons learned. Before participants completed the online survey, they were invited to a virtual meeting to discuss the purpose, objectives, survey terms and rights of survey participants. In addition, the same information was available online during the survey process. Participants of the survey had the option of stopping the survey and returning to complete it at any point during the survey time. Participants were assigned IBM experts to help with the data collection process and help participants answer survey questions. Both IBM experts and participants completed an orientation session to go over the important details of the data collection steps and the definition of key terms. Moreover, participants had access to conference organizers to clarify any questions in the survey when needed. Appendix D provides a sample of the orientation session content that was delivered to IBM experts and survey participants.

The survey was initially executed through a pilot survey process before it went to mass data collection to experiment with the process, collect feedback and tweak accordingly prior to initiation of collecting mass project data. The survey contained 215 questions and took about 3 hours to complete for each project entry. However, the data collection for the survey items took an average of 12 hours. Survey participants were given 60 days to complete the survey and were encouraged to contact other customer resources to help in answering survey questions. All survey participants were responsible for acquiring approvals from IBM customers to participate in this research. No customer names were ever collected throughout the survey process.

The survey questions were grouped logically. Earlier sections of the survey asked general information about the surveyed project such as project name, number of developers, number of functional points, etc. Other sections of the survey were focused on the architectural traits of the surveyed SOA solution. Additional sections of the survey contained questions about business agility within the organization to use the collected

data for establishing business agility index baseline. Other sections of the survey contained questions about the software development process activities that were used to develop the SOA solution. The majority of the questions in the survey were mapped to the metrics proposed as part of this research. Some measurements were direct and were used as is in the calculation of metrics values. Other measurements were used to calculate the value of the metric through the combination of two different direct measurements. For example, the number of services is a direct measurement that plugs directly into a metric that tracks such value. On the other hand, the percentage of business services to web services is an indirect measurement that maps to a metric which tracks such data. To calculate this metric, two direct measurements are collected through the survey: number of business services and number of web services and an indirect measurement is created through the creation of a percentage of business services to web services.

Each of the identified business agility factors received a score that was based on aggregating the values of the answers to questions that are specific to that factor. As different questions were asked in different ways, the answers needed to be encoded. For example, ‘Yes/No’ questions of nominal scale were encoded as 0 and 1 respectively, while questions with answers on an ordinal scale were assigned an appropriate number of points that maps to the value of the ordinal scale of the question. Appendix C provides a brief description of the encoding process for the various types of survey questions.

### **7.5 Survey Overview**

The survey consisted of 14 sections that were used to collect all required information from profiled projects. Table 12 provides an overview of survey sections. Data was initially collected from 36 projects; however three projects were rejected because they did not meet the inclusion criteria, while three were rejected due to significant amounts of missing data that made their collected data questionable. A second round of data collection started in late January 2011 and lasted for one month. A total of 9 non-IBM projects were used for the data collection process, and participants answered only the 80 questions related to our study. As a result, 39 projects were ultimately included in the

data analysis. Of these, 30 projects were executed by IBM professionals, while 9 were non-IBM projects. In both cases the majority of project participants were from the US.

#	Survey Section	Description
1-	Project and Team	Project team, organization, industry and duration
2-	Governance	Governance practices, documents and procedures
3-	Requirements	Requirements gathering methods, roles, top functional and non-functional requirements and complexity
4-	Architectures	Major architectural components, decisions and practices
5-	BPM/SOA	Process and SOA related questions
6-	Cloud	Cloud computing usage and cloud-related questions for
7-	Standards and Open Source	Application of standards and which open sources tools were used as part of surveyed project
8-	Information Architecture	Information architecture decisions and how data was represented in each project
9-	Analytics	Whether analytics was used in the project and what type of information was gathered to facilitate analytics
10-	Implementation	Implementation questions on how services and major components of the system were realized
11-	Operations and Change	What type of change control procedures were used and what type of impact analysis was applied
12-	Product Feedback	Questions related to products used on the project and whether products met expectations
13-	Lessons Learned and Best Practices	What type of best practices were applied and what are the major lessons learned as a result of the project
14-	Assets	What type of assets were reused and what type of assets that were created as a result of this project
15-	Business Agility	What type of requirements for business agility and how this project helped in achieving business agility

**Table 12 Survey Sections**

**7.6 Summary**

This chapter provided an overview of the data collection procedures, population, SOA project profile and data collection activities including survey description. In the next chapter 8, we provide a preliminary overview of the data analysis results and how they relate to our hypotheses. Finally in chapter 9, we provide a detailed discussion into the results and their meaning. We also address the set of recommendations that can be concluded based on the analysis of the data.

## Chapter Eight: Analysis of Data

### 8.1 Introduction

This chapter presents the data analysis steps that were taken to investigate the research questions defined in chapter 1. As a reminder, the primary research questions are as follows:

1. How can attained business agility be measured in completed SOA projects?
2. What are the significant measurable technical and process factors of a SOA solution that are correlated with the attainment of business agility?
3. How can proposed business agility metrics be used in to predict the business agility outcome of in-progress SOA solutions?
4. Are SOA solutions that achieve business agility generally more complex than those that do not?

As mentioned earlier in chapter 7, the primary statistical methods that are used as part of the data analysis steps are exploratory factor analysis and regression analysis. Exploratory factor analysis is used primarily to identify the most significant independent variables that provide the maximum amount of variability in the data. Regression analysis is used to test the hypotheses that were established as part of chapter 7. We start first with descriptive statistics to provide a quick overview of collected data. Later on, we discuss factor analysis and provide some details of the steps taken to reach the reduced set of factors that were later grouped into meaningful logical groupings, referred to as contributors.

8.2 Descriptive Statistics

Categorized by industry, 19% of the projects were from government, 12.5% from Healthcare, 9% each from Banking, Financial Services, Telecommunications and Insurance, and the remainder from other industries. 6% of the projects reported completion within three months. 28% reported durations of three months to one year, 25% took between 1 and 2 years, and the remaining 41% took over two years. In all cases, the study participants were the tech leads and/or the architects that presided over the project during SOA solution implementation. The data in Table 13 and Figure 14 shows the projects by industry and duration and how projects are spread across various industries ensuring a good representation of various project types during the survey process.

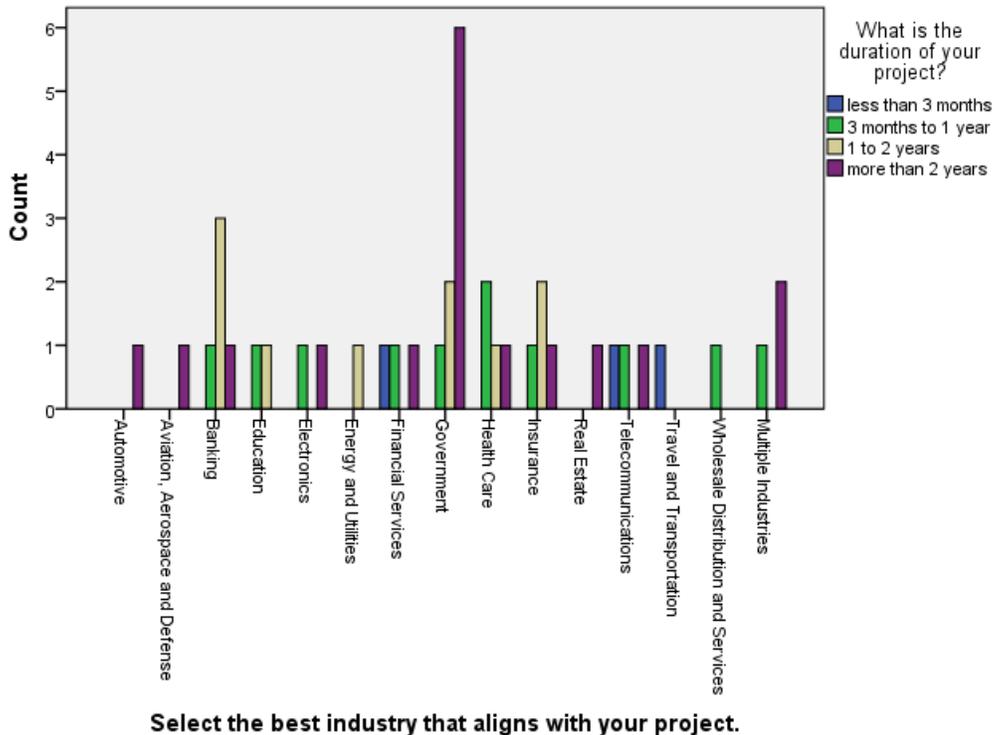
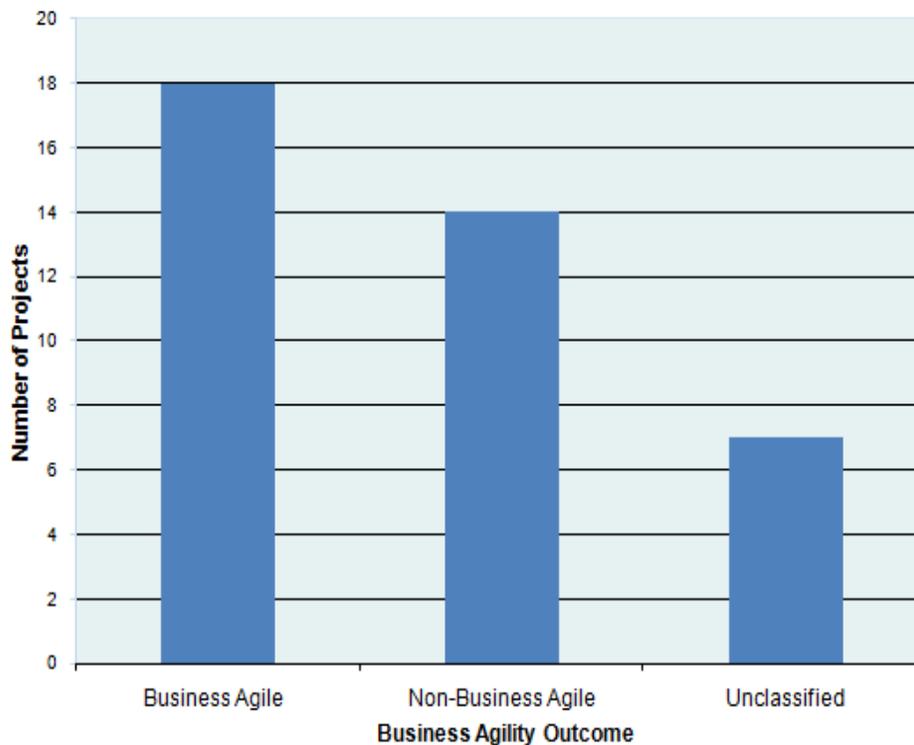


Figure 14 Projects Distribution by Industry and Duration

Select the best industry that aligns with your project	What is the duration of your project?				Total
	Less than 3 months	3 months to 1 year	1 to 2 years	More than 2 years	
Aviation, Aerospace and Defence	0	0	0	1	1
Banking	0	0	2	1	3
Education	0	1	1	0	2
Electronics	0	0	0	1	1
Energy and Utilities	0	0	1	0	1
Financial Services	0	1	0	1	2
Government	0	1	2	3	6
Healthcare	0	2	1	1	4
Insurance	0	1	1	1	3
Real Estate	0	0	0	1	1
Telecommunications	1	1	0	1	3
Travel and Transportation	1	0	0	0	1
Whole Sale Distribution and Services	0	1	0	0	1
Multiple Industries	0	1	0	2	3
<b>Total</b>	<b>2</b>	<b>9</b>	<b>8</b>	<b>13</b>	<b>32</b>

Table 13 Projects by Industry and Duration

Categorized by attainment of business agility, as indicated in Figure 15 , 46% of the projects claimed to attain business agility. On the other hand, 36% of projects did not claim any business agility benefits. The remaining 18% of projects were unclassified regarding the outcome of their project on business agility. The results show a good mix of project outcomes and are not biased towards a single outcome.



**Figure 15 Business Agility Outcomes of Surveyed Projects**

### 8.3 Calculating BAI Values for Collected Data

In chapter 3, we provided an overview of the Business Agility Index (BAI) and how it was devised to differentiate among projects with respect to attainment of business agility. As indicated in section 3.7, the BAI scale consists of eight equally weighted true/false questions.

The BAI index was computed by assigning one point to each question answered positively for a project. The BAI scale therefore ranged from 0 (no indication of business agility) to 8 (strong indication of business agility). As part of the data collection process, each study participant was responsible for engaging an extensive set of business stakeholders for their project in providing a simple yes/no answer to the more direct question: “Did this project achieve business agility?” They were also responsible for answering the eight BAI questions. Results are reported in Table 14, and show that 18 of the projects were classified as business agile, while 14 were classified as non business agile. The remaining 7 projects were unclassified. Results show that 12 of the business agile projects achieved BAI scores of 5 or higher, while 6 achieved scores of 4 or lower. Furthermore, 13 of the non-business agile projects achieved BAI scores of 4 or lower. Consequently, for the SOA projects included in our study, 100% of projects receiving high BAI scores were in fact perceived as business agile. In contrast, 70% of projects receiving low BAI scores were classified as non-business agile. Overall, it appears that BAI is an effective predictor of business agility, at least as it is determined by the participating stakeholders. Later in the threats to validity section of this chapter, we discuss the limitations of using human judgment for operationally defining business agility.

BAI	Business Agility Achieved?			Total
	No	Yes	Unclassified	
0	6	0	1	7
1	0	1	2	3
2	1	1	1	3
3	1	3	1	5
4	6	1	2	9
5	0	1	0	1
6	0	5	0	5
7	0	4	0	4
8	0	2	0	2
<b>Total</b>	<b>14</b>	<b>18</b>	<b>7</b>	<b>39</b>

**Table 14 BAI Assessment of 39 SOA Deployments**

#### 8.4 Factor Analysis

As part of the data collection process, we collected more than 150 attributes that may have some varying relationships among them. Factor analysis is usually ideal with 20 data entries for each factor use [112] [156] . However, other references indicate that factor analysis can be successfully performed with 3-5 data entries for each factor used [157]. Therefore, we performed factor analysis multiple times to reach to the final set of aggregate factors that will be used with the next statistical approach, i.e. regression analysis.

Given the 150 candidate attributes, we performed factor analysis to reduce the overlapping attributes in order to eliminate redundancy among attributes. Moreover,

factor analysis allowed us to detect which attributes had the greatest potential for explaining the variability in the data set. Since this step was applied several different times, we show a sample of one iteration which was used to determine the attributes for the impact analysis factor. Factor analysis assumptions were validated first (i.e. KMO and Barlett's test) to determine whether the resulting data met the preconditions for using the factor analysis model. Both tests demonstrated that it could be appropriately applied. Five factors emerged due to their ability to explain a good percentage of data variability. Due to the small sample, the results were not as reliable as desired but were still useful for reducing the number of attributes to a smaller set. The factor analysis confirmed the use of attributes: support for reporting analysis, measuring SLAs, proactive monitoring thresholds, predictive impact analysis, Historical impact analysis, and Lifecycle impact analysis; while additional attributes of requiring SLAs, Issuing audit reports, Resource management and utilization were added as additional attributes by our group of SOA experts given the additional benefits that are introduced as a result of having dynamic resource utilization and management capabilities. The same factor analysis procedure was repeated for architecture, governance, loose coupling, and BPM, in order to identify a set of relevant attributes for each of those factors. It is worth noting that attributes were excluded if factor analysis confirmed their exclusion and experts did not disagree with factor analysis results.

### **8.5 BAI Scores for Business Agility Factors**

In this section we provide a more detailed discussion of the SOA claims that different dimensions of practice contribute to the achieved business agility. For each of these areas we then report the survey results and discuss the implications of the results upon business agility. As a reminder, we show Table 15 from previous chapters to recap the business agility factors.

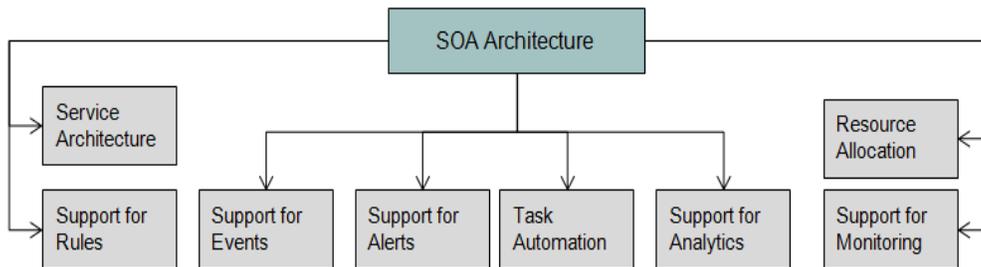
<b>Business Agility Factor</b>	<b>Description</b>	<b>Rationale</b>
SOA Architecture	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc. .	Poorly architected systems fail to fulfill their potential in meeting stakeholder’s objectives. [62] [1] [18] [3] [6] [7]
BPM	BPM factor tracks the attributes of the business process management (BPM) component of a given SOA solution. This includes use of modeling and simulation.	The use of modeling & simulation helps drive the discovery of additional requirements that may impact business agility, identifies bottlenecks, & expose areas that are ripe for streamlining.
Governance	This factor tracks the level of governance in a given SOA solution.	Governed services go through a well-defined process to ensure they are aligned to an organization’s business objectives, catalogued or registered, and monitored[6]
Loose Coupling	This factor tracks properties that inject flexibility into a given SOA solution.	Loosely coupled architectures facilitate faster change with little impact to other components.
Impact Analysis	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.	Underlying systems are equipped with the proper structures that can sense changes immediately and provide corrective measures to potential problems.

**Table 15 Business Agility Factors**

### **8.5.1 SOA Architecture**

**Claim:** A typical SOA system is expected to include certain components such as an enterprise service bus[1] [7] [24]. The architecture score is calculated through a set of attributes that all have equal weights. As depicted in Figure 16, the SOA score includes Service Architecture which includes architectural deliverables and major architectural decisions as part of SOA project such as the use of an enterprise service bus. The SOA Architecture also includes support for rules, support for events, support for alerts, task automation, support for analytics, resource allocation and support for monitoring. Figure 16 shows the attributes of the SOA Architecture factor that were described in chapter 3.

**Rationale:** Poorly architected systems will not fulfill their potential in meeting stakeholder’s objectives [158]. Given that there is no way to enforce SOA best practices [7][159], a SOA score is required to measure how each project adheres to top best practices.

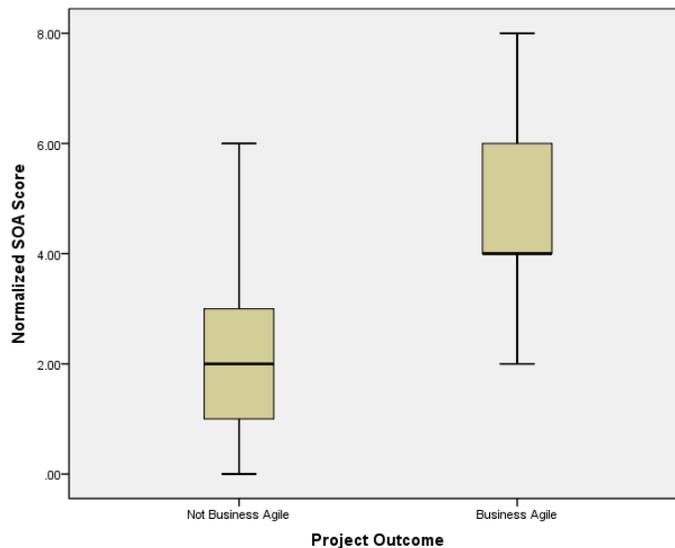


**Figure 16 SOA Architecture Business Agility Factor Attributes**

**Example:** A good architecture that is attempting to enhance business agility should have provisions for handling events and alerts easily.

**Survey Results:**

Figure 17 shows the relationship between projects that achieved, or did not achieve



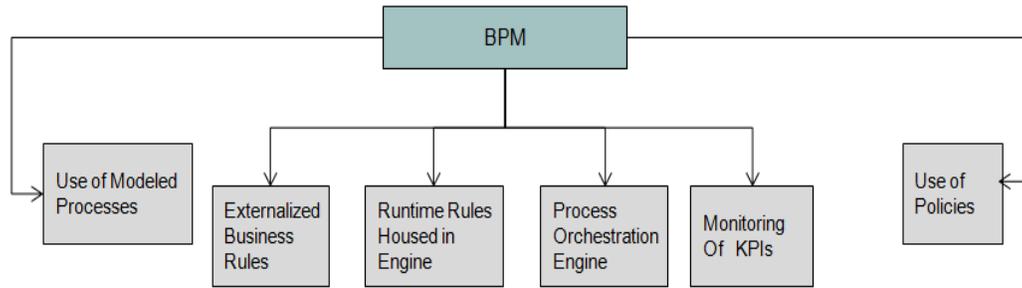
**Figure 17 Project’s Business Agility Outcomes and SOA Architecture Score**

business agility and their corresponding SOA Architecture score. The SOA Architecture score is normalized on a scale from 0 through 8, with 0 meaning that few architectural practices were observed, and 8 meaning that all identified architectural practices were observed. The results show a fairly strong correlation between SOA architecture and business agility, with projects claiming business agility primarily scoring above 5, while those that did not claim business agility had generally lower scores.

An independent-samples t-test was conducted to compare SOA scores for business agile and non-business agile projects. There was a significant difference in SOA scores in business agile projects ( $M=4.61$ ,  $SD=1.68$ ) and SOA scores for non-business agile projects ( $M=2.28$ ,  $SD=1.93$ );  $t(30)=3.63$ ,  $p=0.001$ . The results suggest that SOA architecture factors are significant in predicting business agility outcomes of projects. Specifically, the results indicate that as projects adhere more to SOA architecture guidelines, the better business agility outcome of the project.

### ***8.5.2 Business Process Management (BPM)***

**Claim:** According to the Association of Business Process Management Professionals (ABPMP), BPM is defined as: “a disciplined approach to identify, design, execute, document, monitor, control, and measure both automated and non-automated business processes to achieve consistent, targeted results consistent with an organization's strategic goals” [160]. ABPMP emphasizes the role of collaboration, technology-driven, improvement and streamlining and the end-to-end management of business process to meet business goals and objective with “more agility.” It is generally believed that using BPM components and solutions in combination with SOA enables an agile enterprise and facilitates business agility. Figure 18 shows the attributes of the BPM factor that were described in chapter 3.

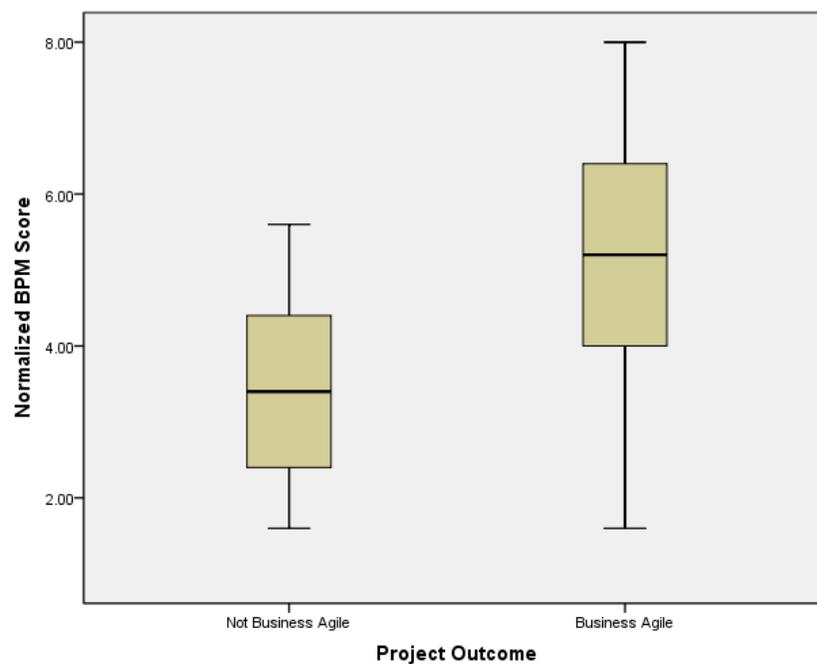


**Figure 18 BPM Business Agility Factor Attributes**

**Rationale:** The optimization of business processes as part of building SOA solutions is one of the key factors for realizing business agility requirements [11]. The use of modeling and simulation in SOA helps drive the discovery of additional requirements especially those that may impact business agility. It can also uncover bottlenecks of the business process and expose additional areas that are ripe for streamlining.

**Example:** The level of business process modeling that is accomplished for a project, level of simulation used, level of Key Performance Indicators monitoring, and the level of use externalized business rules and service implementations.

**Survey Results:** Figure 19 shows that projects that achieved business agility were likely to score higher on the BPM score. The majority of projects that received higher BPM scores reported achieving some business agility benefits for their projects. Projects that did not claim business agility benefits tended to score lower on the BPM scale. An independent-samples t-test was conducted to compare BPM scores for business agile and non-business agile projects. There was a significant difference in BPM scores in business agile projects ( $M=5.13$ ,  $SD=1.61$ ) and BPM scores for non-business agile projects ( $M=3.46$ ,  $SD=1.24$ );  $t(30)= 3.21$ ,  $p = 0.003$ . The results suggest that BPM factors are significant in predicting business agility outcomes of projects. Specifically, the results indicate that as projects adhere more to BPM guidelines, the business agility outcome of the project also increases.

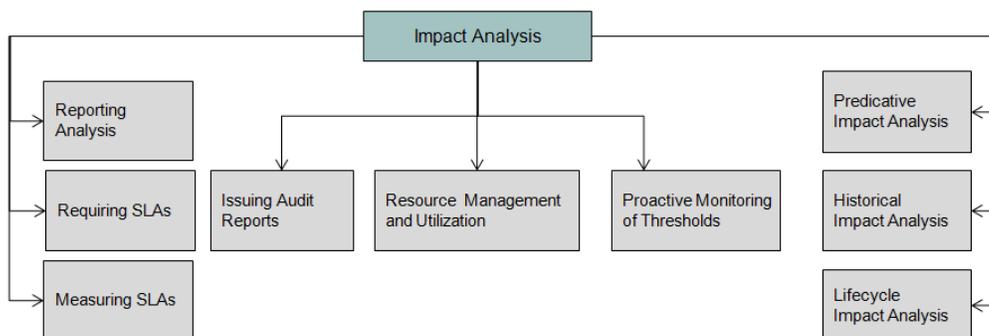


**Figure 19 Project's Business Agility and BPM Score**

### 8.5.3 Impact Analysis

**Claim:** According to Bohner et al. [73], change impact analysis involves "identifying the potential consequences of a change, or estimating what needs to be modified to accomplish a change". Identifying changes and assessing their impact is a key requirement for achieving business agility. In SOA, Key Performance Indicators (KPIs) represent the strategic quantified goals of an organization and its services. KPIs provide specific performance objectives, often stated in terms of response times, throughput, latency, security, reliability, usability, accuracy, or cost. As such, KPIs can be used to evaluate whether a deployed system is currently achieving its stated business goals. The concept of service level agreements (SLAs) also plays a role in the implementation of an overall strategy for change impact analysis in SOA solutions. Therefore, observed runtime deviances from advertised KPIs and SLAs triggers change impact analysis steps. Figure 20 shows the attributes of the Impact Analysis factor that were described in chapter 3.

**Rationale:** Business agility's demand for proactive sensing mandates that underlying systems are equipped with the proper structures that can sense changes immediately and provide corrective measures.

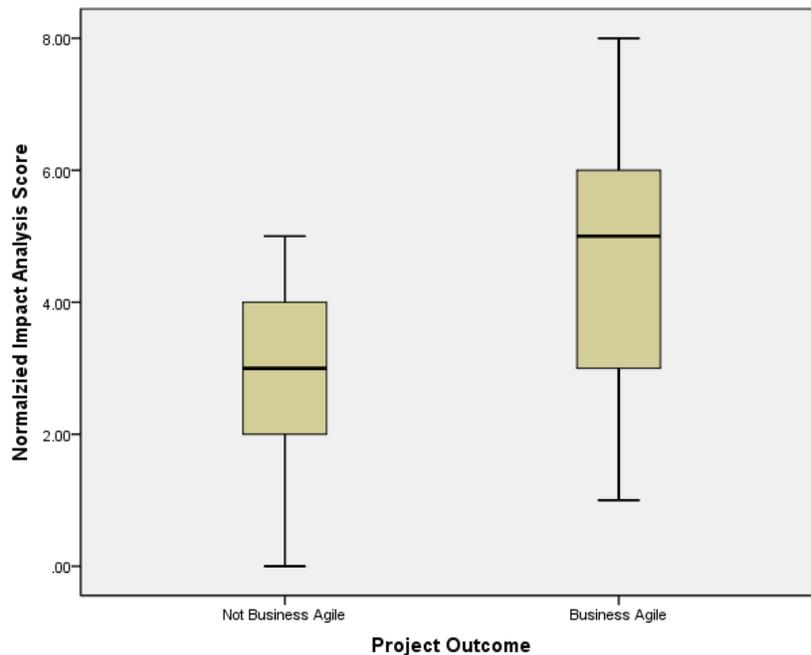


**Figure 20 Impact Analysis Business Agility Factor Attributes**

**Examples:** The ability to measure SLAs and assign thresholds in order to take corrective measures proactively. Services might not deliver on its previously defined SLAs (e.g.,

performance, availability, or cost), and a decision must be made as to what to do with underperforming services.

**Survey Results:** The data for Impact Analysis shown in Figure 21 shows that projects which achieved business agility were likely to score higher on the Impact Analysis score than those that did not achieve business agility. An independent-samples t-test was conducted to compare Impact Analysis scores for business agile and non-business agile projects. There was a significant difference in Impact Analysis scores in business agile projects ( $M=4.56$ ,  $SD=2.17$ ) and Impact Analysis scores for non-business agile projects ( $M=2.92$ ,  $SD=1.33$ );  $t(30)= 2.46$ ,  $p = 0.020$ . The results suggest that impact analysis factor is likely to be significant in predicting business agility outcomes of projects but to a lesser degree than previously mentioned SOA and BPM factors. Specifically, the results indicate that projects which adhere more closely to impact analysis guidelines tend to achieve better business agility outcomes.

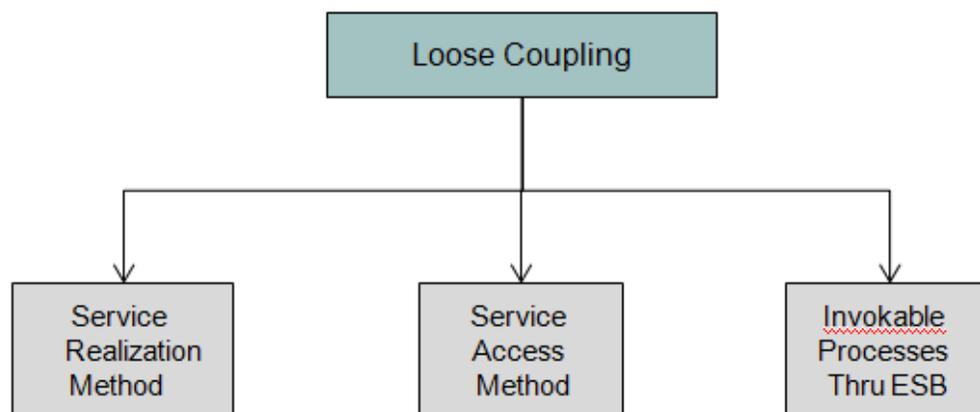


**Figure 21 Project's Business Agility and Impact Analysis Score**

### 8.5.4 Loose Coupling

**Claim:** Coupling refers to the degree of interconnections between software solution modules [79]. According to [161], loosely coupled systems are easier to maintain and understand, as changes can be made independently to different modules without significant ripple effects. Figure 22 shows the attributes of the Loose Coupling factor that were described in chapter 3.

**Rationale:** Loosely coupled architectures facilitate faster change with little impact to other solution components. Business agility is grounded in faster and more efficient handling of events and requires underlying structures to be capable of quick change with minimal or no impact to existing systems. Fiammante [7] points to some issues that plagued some of the early adopters of SOA and BPM due to their lack of experience in using the right tools and making the right architectural decisions. For example, some companies that adopted SOA failed to realize the power of loose coupling among business process components and the underlying services. This in essence led to the creation of rigid business process that largely failed to achieve the desired business agility benefits.

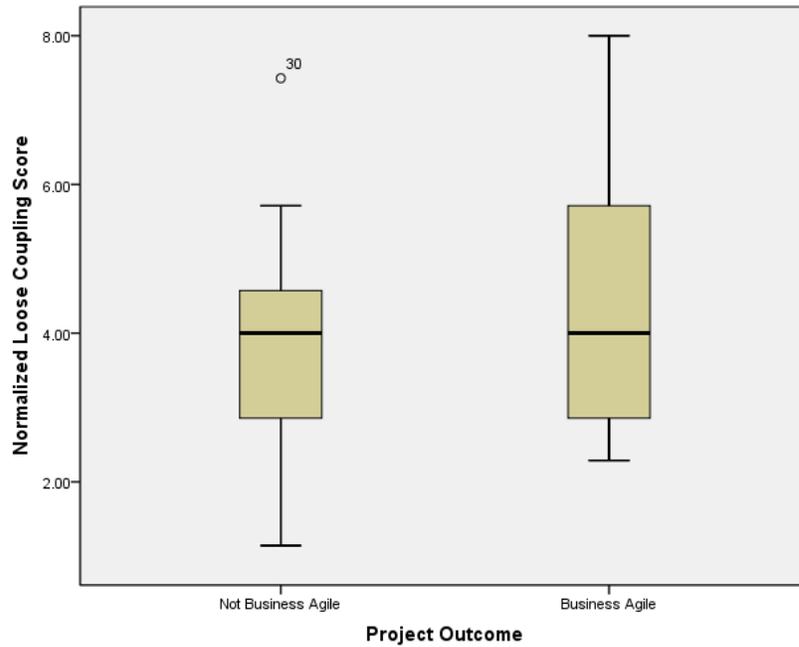


**Figure 22 Loose Coupling Business Agility Factor Attributes**

**Examples:** There are many service realization patterns that can be used for exposing and using services including the two primary patterns of Direct Exposure (DE) and Access Services. DE refers to exposing current IT systems or modules as a service without having to go through an intermediary component.

Access services, on the other hand, refer to exposing current IT systems or a module as a service by going through an intermediary component such as an EJB. Access services are more loosely coupled than direct exposure services. Moreover, the use of virtualization layers, or enterprise service bus, injects additional loose coupling into the overall structure of a SOA solution.

**Survey Results:** Figure 23 shows that projects which reported achieving business agility benefits tended to score higher on the Loose Coupling scale. Furthermore, the concentration of projects that achieved business agility reported scores of Loose Coupling that were closer to the mean. This indicates that some form of loose coupling is probably sufficient to enhance the outcome of business agility. Extreme loose coupling may have the potential to increase the complexity of the overall solution. In fact, SOA best practices advocate the need for building ESBs as major components of any SOA solution [159]. Independent-sample t-test results did not indicate significance for this factor. However, the data trends for business agile projects do suggest that business agile projects scored slightly better when comparing the means of business agile and non-business agile projects.



**Figure 23 Project’s Business Agility and Loose Coupling Score**

### 8.5.5 Governance

**Claim:** According to Brown [6], “SOA Governance is an extension of IT Governance that is focused on the business and IT lifecycle of services to ensure business value”. Properly governed SOA services are those services that are funded properly for an obvious business reason and tie directly to business goals and objectives. Moreover, governed services are advertised, managed, secured and deployed appropriately to an infrastructure that will meet execution demands. Therefore, ensuring that services are well-governed is a key attribute of well-built SOA solutions [6] [7]. Figure 24 shows the attributes of the Governance factor that were described in chapter 3.

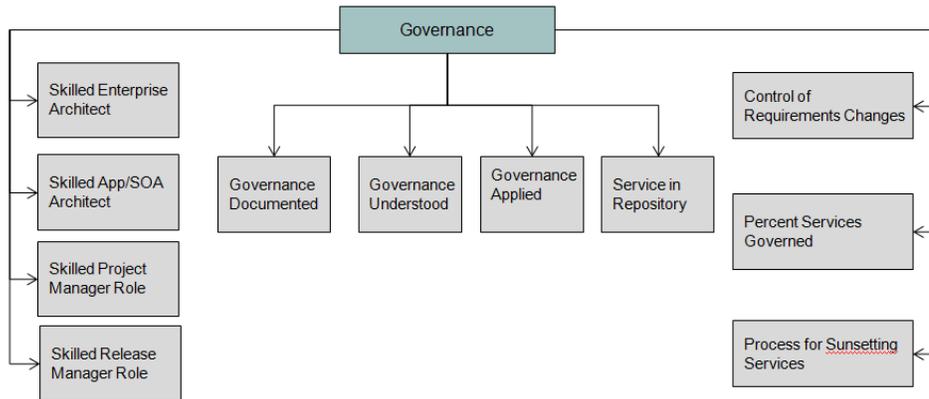


Figure 24 Governance Business Agility Factor Attributes

**Rationale:** Agile enterprises are efficient and control their resources. Therefore, better business agility can be achieved through governance of services including their proactive monitoring.

**Examples:** Existence of an enterprise architect and project manager, tracking of requirements changes, rules for managing and creating services.

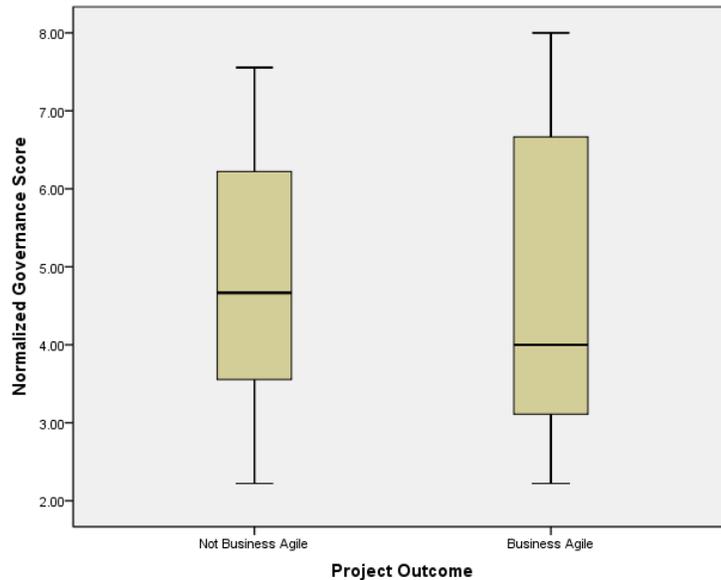


Figure 25 Project’s Business Agility and Governance Score

**Survey Results:** Surprisingly, the results from our survey, depicted in Figure 25 provided no support for the claim that better governance practices led to an increase in business agility. We found that projects which achieved business agility tended to include a wide degree of governance practices, with some projects claiming business agility scoring quite high, and some scoring quite low on the governance scale. These initial observations were supported by the fact that independent-sample t-test results did not indicate significance for this factor. These results warrant further investigation to determine if other factors such as the size of the project impact the importance of governance within a project.

### **8.6 Preliminary Results**

The earlier analysis in the previous section suggests that projects which reported achieving business agility benefits tended to score higher on the hypothesized factors of SOA Architecture, BPM, Impact Analysis and Loose Coupling. Surprisingly, the investigated data did not point to a strong relationship between achieving business agility and Governance. In fact, a closer look at the results as summarized in Table 16, which shows the factors and their meanings after being normalized on a scale from 0 through 8, reveals that projects that claimed contributions to business agility had mean value for SOA score 102.9% higher than those projects that did not claim achieving business agility. A less significant result was reported for BPM, Impact Analysis scores and Loose Coupling score with values 48.27%, 56.16% and 9.50% respectively. As for Governance there was a difference of (-6.39%) in the mean between projects that achieved business agility versus those that did not.

The significance of the SOA score is expected. Reaching the right architectural decisions and having the right skilled roles within a SOA project has always been an accepted approach for achieving business agility. The lack of divergence among projects with respect to Governance shows that while Governance may be considered to be important, it is not a predictor of business agility outcome. Further detailed statistical analysis is required to determine the level of significance of Governance as a contributor to business

agility. The other factors such as BPM score, Loose Coupling and Impact Analysis all showed some divergence between the means of business agile, versus non-business agile projects, indicating that such factors play a role in achieving business agility in SOA projects.

	<b>SOA</b>	<b>BPM</b>	<b>IA</b>	<b>LC</b>	<b>Gov</b>
Business Agile Mean	4.61	5.13	4.56	4.38	4.69
Not Business Agile Mean	2.28	3.46	2.92	4.00	5.01
Change Percentage	102.19	48.27	56.16	9.50	-6.39

**Table 16 Means for Normalized Business Agility Factors Scores**

The analysis thus far has identified a set of independent variables, business agility contributors or factors, which can be used to analyze their feasibility in predicting our dependent variable business agility as measured by the BAI. In the next section, we employ the use of regression analysis to further test our hypothesis and build a predictive model to help in predicting the business agility outcome of SOA projects before they are completed.

### **8.7 Building a Predictive Model for Business Agility using Regression Analysis**

Although the BAI is able to largely differentiate between business and non-business agile projects, its usefulness is limited because it requires business expertise and can only be assessed after the fact, once the SOA system has been built, deployed, and its business agility tried and tested. The BAI therefore does not provide predictive value. In contrast the intent of our work is to develop a concrete set of metrics that are easily collectable by technical project personnel during early phases of a project, and which have the capability of accurately differentiating between projects which are likely to attain business agility and those which are not. To accomplish this we constructed the PBAI by

identifying technical factors which could be collected and measured during early phases of the project, and which were shown to be highly correlated with the attainment of business agility.

For statistical purposes, the five composite scores shown in served as independent variables, while the BAI scores served as dependent variables. 38% of projects we studied scored below the midpoint of the BAI scale, while the remaining 62% performed above it, meaning that the overall sample provided sufficient representation of both business agile and non-business agile SOA deployments. The SOA, Impact Analysis, BPM, Loose Coupling, Governance scores, and the BAI index were computed for each project.

An initial analysis, reported in Table 16 reveals that projects claiming contributions to business agility had a mean value for the SOA score 102.19% higher than those projects that did not claim achieving business agility. A less significant result was reported for BPM, Impact Analysis and Loose Coupling scores with values 48.27%, 56.16% and 9.50% respectively. As for Governance there was a difference of -6.39% in the mean between projects that achieved business agility versus those that did not.

The significant difference in business agile and non-business agile SOA scores is expected, as making and implementing sound architectural decisions has always been perceived as critical for attaining business agility. The lack of divergence among projects with respect to Governance surprisingly suggests that at least in the projects studied, governance is not a predictor of business agility outcome. The other factors such as BPM score, Loose Coupling and Impact Analysis all showed some differences between the means of business agile, versus non-business agile projects, indicating that such factors have the potential for playing a role in achieving business agility in SOA projects.

## 8.8 Building the Predicted Business Agility Index (PBAI)

In this research, the business agility predictor model was constructed in several stages. First, a Business Agility Index (BAI) was developed to measure after-the-fact attainment of business agility in each of the studied SOA deployments. Using the BAI requires expertise in both business and technology, and it can only be applied after the SOA project has been deployed for a period of time and its business agility has been put to the test. In the second stage of model construction we developed a Predicted BAI (PBAI) which is designed for use in early stages of a SOA project to predict future business agility based on the degree to which various attributes are present in the project. The PBAI was developed as a next step in this research to answer the research question regarding the relationship between business agility and SOA deployments. We continue to use GQM-MEDEA as our guiding method to complete the activities of this research. The earlier results have identified a set of independent variables that were used to establish the relationship to the BAI scores, dependent variable, in a process to build PBAI and to validate it against the BAI.

### 8.8.1 Primary Hypothesis

As stated earlier in section 8.8.1 the fundamental hypothesis is similar for all the investigated factors as part of this study. The null hypothesis  $H_0$  states that the coefficients relating our factors (independent variables) to the BAI are equal to zero. Or simply stated, that business agility is not impacted by SOA deployment factors.  $H_0: \beta_1 = 0$ . The alternative hypothesis  $H_1$  states that the coefficients relating our factors (independent variables) to the BAI (dependent) variable are not equal to zero.  $H_1: \beta_1 \neq 0$ . Stated differently, there is a relationship between business agility and our hypothesized factors (SOA Score, BPM Score, Impact Analysis Score, Loose Coupling Score and Governance Score).

### ***8.8.2 Building the Predictor Model***

To construct the PBAI we utilized multiple linear regression analysis to investigate the relationship between composite factors (i.e. the independent variables) and the business agility index (i.e. the dependent variable). This relationship is represented as  $y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + e_i$ , where  $\beta_0$  is the constant term and  $\beta_1$  to  $\beta_p$  are the coefficients relating the  $p$  explanatory variables to the dependent variable.

The descriptive statistics for the identified factors were generated and examined. Projects with missing data were simply dropped from the analysis, reducing the number of analyzed projects from 39 to 32. Regression assumptions such as normality and collinearity were checked and found to be appropriate for the analysis. Correlation analysis between each of the factors and the dependent variable BAI showed positive correlations for each factor. Table 17 reveals that SOA, IA and BPM are the most significantly correlated factors to achieving business agility in SOA solutions. Loose coupling is also significant, however, to a lesser degree than previous factors. SOA Governance, on the other hand, is not significant based on the collected data. This further indicates that data used for this study meets the requirements for multiple regression.

	<b>SOA</b>	<b>BPM</b>	<b>IA</b>	<b>LC</b>	<b>Gov</b>
BAI Pearson Correlation	.675	0.50	.52	0.41	0.187
Sig. (2-tailed)	.000	.004	0.002	0.019	0.305
N	32	32	32	32	32
All correlations are significant except for Gov					

**Table 17 Simple Correlations between BAI Scores and Business Agility**

### Factors

	<b>Const</b>	<b>SOA</b>	<b>BPM</b>	<b>IA</b>	<b>LC</b>	<b>Gov</b>
<b>B</b>	-1.98	.65	.41	.44	.18	.12
<b>Sig.</b>	.034	.000**	.027**	.005**	.133*	.309*
** Significant      * Not Significant						

**Table 18 Multiple Linear Regression Results of Business Agility Factors**

The results of multiple regression showed that our independent variables produced an adjusted R2 of .64 (  $F(3,31) = 19.46, p = .000$ ) for the prediction of achieving business agility. All of the tested predictors turned out to be significant (  $p < .05$ ) except for the Governance Score (  $p = .13$ ) and Loose Coupling (  $p = .18$  ). Based on the results documented Table 18 , the regression analysis indicated that SOA, BPM and IA are reliable factors for predicting BAI. However, for Gov and LC, given the reported

insignificant levels, we excluded them from our model. Using our model parameters, we write the predicted business agility index PBAI calculation equation as follows:

$$PBAI = 0.65 SOA + 0.41 BPM + 0.44 IA - 1.98$$

Where: SOA is the SOA Score, BPM is BPM Score and IA is Impact Analysis Score.

### 8.9 Results Analysis

The final model includes the SOA Score, Impact Analysis Score and BPM Score, which were all shown to make positive contributions to the attainment of business agility. A closer look at the individual elements contributing to the SOA Score reveals that inherent support of a given architecture for rules, events, task automation, alerts, monitoring and analytics is essential to achieving the desired business agility improvements. Given that business agility is primarily concerned with the continuous sensing and adapting to new conditions, the significance of this factor seems logical. Similarly it is not surprising to find that impact analysis and BPM capabilities contribute to business agility. Impact analysis ensures that SLAs are monitored proactively and therefore it contributes directly to the business agility goals of responding to ad-hoc changes. Similarly the optimized business processes and proper alignment with IT that happens as a result of BPM's best practices ensures that business processes are well thought out and not created randomly. BPM attributes such as modeling, monitoring of key performance indicators and the use of rule engines to externalize business rules add significant amount of flexibility to SOA solutions. It is worth noting that monitoring SLAs or KPIs is not restricted to SOA solutions. Any IT solution can be architected to incorporate aspects of dynamic and predictive impact analysis components as well as BPM best practices

Our analysis could not conclusively identify loose coupling as a significant factor. However, our results show a positive correlation between loose coupling and the BAI as documented in Table 18. In particular, ESBs are widely considered important in

developing a loosely coupled solution primarily because they provide a virtualization layer [1] [7] and also enable many features that contribute to the sensing and handling of ad-hoc situations. We therefore experimented with having an ESB composite factor as a separate part of the model, and our analysis showed that use of ESBs is a particularly strong indicator for achieving business agility in a SOA deployment. However, given the significance levels reported in regression analysis results, we chose not to include this factor in the overall predictive model.

The Governance score predictor was found to be insignificant. The governance score includes a mix of capabilities including establishing and tracking project management and architect roles, advertising and sunseting services, and tracking requirements and requests for change. One reason that the factor may not have been identified as significant in our model is because all projects that scored high in impact analysis also tended to score high on the governance factor implying the adoption of strong governance practices. This overlap will be explored further in our future work.

One of the unexpected results from our study was the inability to draw conclusions concerning reuse. Attributes related to reuse were eliminated in early stages of factor analysis. This could be explained by the fact that the project data was collected from projects that were completed over the past four years, while effective reuse is often associated with multiple mature projects. Furthermore, a closer inspection of our collected data revealed a significant amount of missing data with respect to re-use questions. As a result of dropping projects with missing data, it is possible that re-use was not given full consideration as a predictive factor. Table 19 shows the final set of business agility contributors based on our analysis.

<b>Business Agility Contributors</b>	<b>Description</b>
SOA Architecture	This factor tracks major architectural decisions that are known to support flexible environments such as events, monitoring, analytics, etc.
BPM	BPM factor tracks the attributes of the business process management (BPM) component of a given SOA solution.
Impact Analysis	This factor tracks the properties that are known to enhance the ability of proactive sensing and adjusting to events.

**Table 19 Business Agility Contributors**

### ***8.9.1 Validating Regression Analysis Assumptions***

Before carrying out regression analysis, a validation of regression analysis assumptions is required. Regression analysis requires the underlying collected data to meet the following conditions before regression analysis can be effectively applied.

- **Sample is random and independent**

Based on our data collection method using the IBM conference and non-IBM projects, and based on our selection criteria of automatically accepting any SOA project that meets our project profile criteria, we can assume that our sample is random. Observed data and measurements from projects are used once in any given time and were not reused for the same type of measurement and can therefore be considered independent.

- **Variables are normally distributed**

In order for regression to be used, it assumes that variables have normal distributions. The size of our sample  $> 30$  guarantees that the central limit theorem (CLT) applies and data will be normally distributed due to the size of our sample.

- **Liner relationship between dependent and independent variables**

The use of multi regression is restricted to those variables that are related to each other in a linear fashion [90][112]. Non-linear relationships between dependent and independent variables will result in under estimating the relationship between variables and therefore skewing the test results [162]. There are many methods available to detect linear relationships among variables. We will validate this condition of linearity between dependent and independent variables later in this chapter using a method that is readily available in statistical software packages, residual plot analysis.

- **Normally distributed and independent residuals**

Residuals are normally distributed since they tend to follow the sample size. If we have  $> 30$  sample size, this implies we will have  $>30$  residuals, therefore our CLT applies for the assumption of normality of the list of residuals. We can do formal test for normality of residuals using histograms and probability plots but since the data set is  $> 30$ , we can assume that they are normally distributed.

Independent residuals test need to be executed if we have a component of time in the data collected. If the data is static, we can just assume that residuals are independent. A Durbin-Watson test that is available through software packages can formally test for the independence of residuals in the data.

- **Homoscedasticity and Multcolinearity**

Homoscedasticity means that the variance of errors is the same across all levels of the independent variable [112]. In other words, the residuals behave randomly throughout the entire data set and there should be no pattern forming between any of the variables and the residuals when plotted. If any pattern forms or any outliers are detected, then homoscedasticity assumption is violated which implies

the model is heteroscedastic. If it homoscedasticity is violated, it may indicate the use of insignificant variables.

Multicollinearity is a problem with being able to separate the effects of two (or more) variables on an outcome variable [112]. Multicollinearity happens when a large number of independent variables are used in a regression model. This problem must be eliminated while building the model to ensure that no two variables are similar. As noted earlier in section 4.3.4, Fenton[94] has shown this to be the case to in various previous metrics research and therefore reduced the level of trust in the resulting reported models. The issue of multicollinearity casts doubt into the impact of an independent variable on the variance of the dependent variable. This situation typically happens when a set of independent variables are more correlated with each other than they are with the dependent variable [112]. Multicollinearity leads to unrealistic coefficient estimates and ultimately to bad interpretations of the model. The easiest and best way to remove multicollinearity is to remove the redundant variable from the model [112]. Other treatments include adding more observations and selecting different independent variables for the regression model.

Our analysis for the data showed that no patterns were formed when plotting the residuals and the variables suggesting that homoscedasticity was not violated.

Similarly, our use of grouping of factors and using factor analysis ensured that our independent variables that were used in the regression model were not collinear.

#### - **Detect Influential Observations**

This test is performed to identify influential observations that may have large impact on the regression model. To achieve this, an observation, i.e. case study entry, from model was deleted and the model was re-computed to identify such influential observations. This process was repeated n times deleting exactly one observation from the model each time. In general, whenever an influential observation or an outlier is present in the model, it becomes necessary to

investigate the issues behind their presence instead of immediately discarding them. If their presence is due to data error, then this is an opportunity to fix the error and retain another observation [90]. Applying this to our data revealed that there were no influential observations that required any special attention.

### **8.10 Predictive Model Cross Validation**

In order to generalize predictions for business agility benefits as a result of building SOA solutions, a data set will be required to validate the prediction capabilities of the model. According to [4], this data set should not be part of the dataset that was used to build the original model. This will provide additional validity to the model to showcase its ability to predict business agility for a new dataset. In order to accomplish this, cross-validation: leave one out validation method was used in this research.

Because of the significant cost and effort of collecting data, we did not have any additional projects available for evaluating the PBAI. We therefore conducted a leave-one out cross-validation experiment in which the entire regression analysis was repeated and a new PBAI equation was generated from a training set of 31 projects, and then used to predict the BAI of the remaining project. The experiment was repeated 32 times until a prediction had been made for each project. There were two interesting questions addressed by this experiment.

The first evaluated the stability of the PBAI equation over the 32 computations. In all of the experiments, the generated PBAI model factors were similar 90% of the time and included the same major factors reported in the results of this chapter. The other 10% showed that LC factor nudged BPM factor with a slight margin of significance. The Governance factor was consistently excluded from all generated PBAI models.

The second evaluation compared the computed PBAI scores against the BAI scores and the true/false business agility ranking for each project. Based on our previous observations that a BAI value of five or greater is considered to be business agile, results,

which are reported in Table 17, show that PBAI predicted 15 projects to be business agile (i.e. scoring 5 or higher in the PBAI) of which 14 had been originally classified as business agile. It also predicted 17 projects to be non-business agile, of which 13 were originally classified as non-business agile, and 4 as business agile. Furthermore, as depicted visually in Figure 26, our results showed that in 87.5% of the cases the PBAI scores agreed with the BAI scores. This result was confirmed by a Pearson correlation analysis which returned a correlation of 0.82 between the predicted PBAI scores and reported BAI scores. In general these results indicate that the PBAI was able to differentiate between business agile and non-business agile projects.

#	BAI	PBAI	Agile	PBAI agreement with BAI
12	7	7.88	Y	True
10	6	7.54	Y	True
29	8	7.38	Y	True
28	6	7.12	Y	True
17	7	6.51	Y	True
2	7	6.17	Y	True
7	8	5.74	Y	True
18	7	5.26	Y	True
22	4	5.19	N	False
11	5	5.04	Y	True
14	3	4.76	Y	True
19	6	4.65	Y	False
6	3	4.54	Y	True
15	6	4.36	Y	False
1	4	4.10	Y	True
32	4	3.81	N	True

#	BAI	PBAI	Agile	PBAI agreement with BAI
20	6	3.62	Y	False
3	1	3.22	Y	True
21	4	3.18	N	True
5	4	3.17	N	True
4	2	3.16	Y	True
23	4	2.51	N	True
25	0	2.41	N	True
31	4	2.34	N	True
30	0	2.08	N	True
9	3	1.74	Y	True
26	0	1.59	N	True
24	2	1.31	N	True
8	0	1.26	N	True
27	0	1.15	N	True
16	3	0.71	N	True
13	0	0.49	N	True

Table 20 Predicted BAI vs. Reported BAI for All Projects

(Values greater than 4 are predicted as business agile)

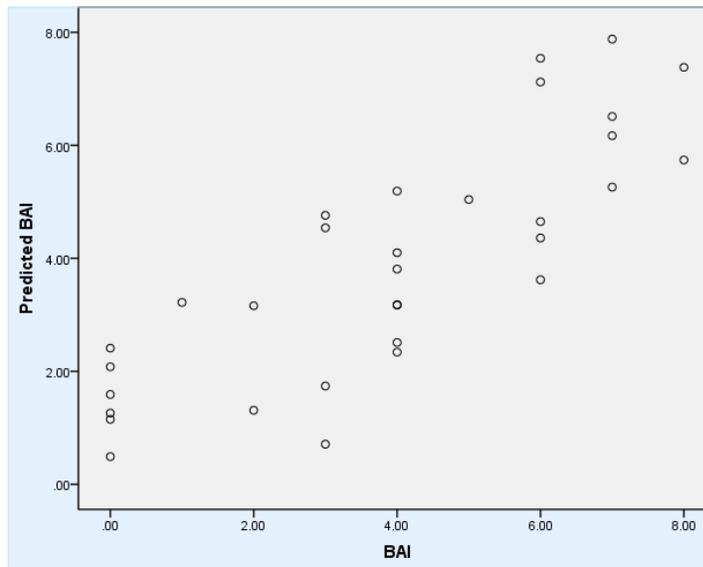


Figure 26 BAI vs. Predicted BAI for All Projects

### 8.11 Impact of Achieving Business Agility on Complexity

The results included so far did not address our final research question that is related to the level of complexity associated with achieving business agility. To investigate this relationship, we devised a measurement tool that quantifies the overall complexity of a given SOA solution and used it to differentiate among analyzed SOA projects based on their complexity. As indicated in section 1.2, the solution complexity index (SCI) is a scale that is normalized and can range from 0, lowest, to 10. A project with high SCI values is considered a complex project. Alternatively, SOA projects with low SCI values are less complex relative to the ones with higher SCI values. The SCI is a relative index and is constructed as a means to compare projects against the same criteria. The components of the SCI are described in Table 21. For each component of the SCI, a score was created to enable the calculation of a final value for the SCI. For example, project duration is a question on the survey that was calculated in months for the project. The results were mapped to numbers that were later used in the calculation of the SCI index.

<b>SCI Component</b>	<b>Description</b>
<b>Project Duration</b>	The total duration of project in months.
<b>Number of Technical People on Project</b>	The number of technical people that were involved with the project regardless of their affiliation (IBM or non-IBM)
<b>Number of business people on Project</b>	The number of business people that were involved with the project regardless of their affiliation (IBM or non-IBM)
<b>Number of Services per Project</b>	Number of services that were created for the project
<b>Number of Functional Requirements per Project</b>	Total number of functional requirements for the project

**Table 21 SCI Components**

The correlation analysis between attained business agility (BAI) and solution complexity (SCI) showed positive weak correlation. Table 22 reveals SCI is weakly positively correlated with BAI with Pearson correlation of .465 that is significant at the .01 level. Furthermore, as depicted visually in Figure 27, our results showed that projects that achieved higher levels of business agility as indicated through their BAI values tended to score higher on the complexity scale. However, there are several cases in which business agility was achieved at low levels of complexity. Conversely, the three cases of greatest complexity, all achieved high business agility.

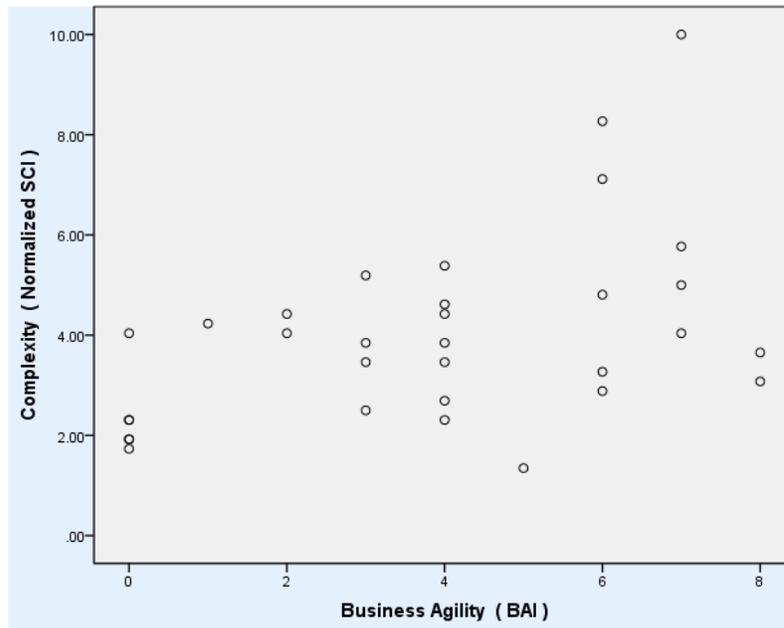


Figure 27 BAI vs. SCI for All Projects

	<b>SCI</b>
BAI Pearson Correlation	.465**
Sig. (2-tailed)	.007
N	32

\*\* Correlation is significant at the .01 level (2-tailed)

**Table 22 Simple Correlation between BAI and SCI**

### 8.12 Applying PBAI for Not Included Projects

As indicated earlier, some of the survey respondents did not provide an answer to the question that indicates to whether their project impacted business agility positively or negatively. Therefore such projects were not included in the analysis or building of the predicted business agility model. Table 23 shows the list of the unclassified projects along with their calculated scores for the business agility factors and BAI values.

#	SOA	BPM	IA	LC	Gov	BAI
1-	2	5.2	5	2.86	3.56	0
2-	0	2.8	3	0	4.44	1
3-	3	5.2	3	5.71	2.22	1
4-	1	2	6	5.71	5.33	2
5-	0	4.4	5	2.86	3.11	3
6-	6	6.8	4	0	3.56	4
7-	3	6	3	4.57	4.44	4

**Table 23 Business Agility Factors and BAI Scores for Unclassified Projects**

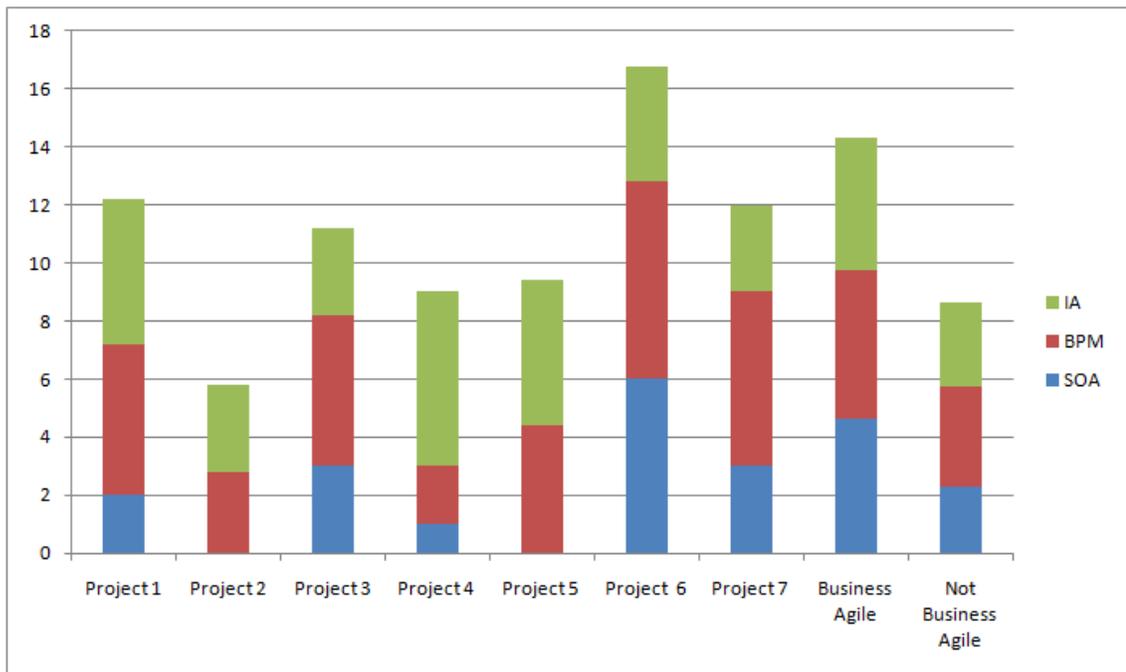
The data in Table 24 show the means for project profiles that contributed to business agility and those that did not. A cursory look at the data in Table 23 shows two groups of projects. One group of projects (projects 1 thru 5) that clearly did not contribute to business agility based on their calculated BAI scores of less than 4. The other group (projects 6 and 7) show BAI values of four which is considered border line to contributing to business agility.

	<b>SOA</b>	<b>BPM</b>	<b>IA</b>	<b>LC</b>	<b>Gov</b>
Business Agile Mean	4.61	5.13	4.56	4.38	4.69
Not Business Agile Mean	2.28	3.46	2.92	4.00	5.01

**Table 24 Means for Business Agility Factors for both Agile and Not Business Agile**

Further plotting of projects 1 thru 7 is shown in Figure 28 where the business agility contributors' values for the projects are drawn side by side next to the profile of those projects that reported contributions to business agility and those that did not. The diagram shows that projects 1 thru 5 closely match the profile of Not Business Agile profile. Project 7 can be construed either way given its BAI value of 4. However, a closer look at Project 7 stacked bar shows that IA and SOA business agility contributors match those of the Not Business Agile profile. Therefore, Project 7 aligns more with the Not Business Agile Profile. Project 6 is more interesting since its BAI value is 4 and its business agility contributor values show the right representation of the values required for attaining business agility and closely matches that of the Agile profile. Applying the

PBAI model to the data for the unclassified projects confirmed the differentiation between projects 6 and 7 where both had similar BAI values.



**Figure 28 Projects 1-7 Business Agility Contributors Profile vs. Agile and Not Agile Profiles**

The PBAI values in Table 25 clearly show that project 6 is the only project that should be classified as contributing to business agility. The data also show that PBAI was able to correctly classify those projects that did not contribute to business agility.

#	SOA	BPM	IA	BAI	PBAI
1-	2	5.2	5	0	3.65
2-	0	2.8	3	1	0.49
3-	3	5.2	3	1	3.42
4-	1	2	6	2	2.13
5-	0	4.4	5	3	2.02
6-	6	6.8	4	4	6.5
7-	3	6	3	4	3.75

**Table 25 BAI vs. PBAI for Unclassified Projects**

### 8.13 Threats to Validity

The reliability of the empirical validation will ultimately rest on many factors that are based on experimental design process, validity of measurements and the use of appropriate statistical techniques to analyze the data and reach proper conclusions. Many authors have voiced their concerns to the validity of published results in the field of software metrics especially for studies that attempted to empirically validate complexity and object oriented (OO) metrics [90] [94] [101]. Other authors echoed similar concerns that were common in five of the key influential usability studies in the human-computer-interaction (HCI) field [163]. The critique is centered on the threats to validity of the experimental design.

The threats to the validity of this research that are related to the survey and design of the study are grouped as suggested by Wohlin et al. [164] and presented in the following section.

### *8.13.1 Construct Validity*

Threats to construct validity originate from the use of SOA experts to construct the BAI model and to evaluate each of SOA projects against the BAI and with respect to its attained level of agility. It is possible that the SOA experts did not include all pertinent factors in the BAI model, or in the list of 150 attributes assessed for each of the projects. This problem was at least partially overcome by the rigor with which the study was designed and executed. For example, we employed the practices defined by GQM-MEDEA to identify a relatively complete and appropriate set of factors for the PBAI model. Furthermore, all of the statistical techniques applied in this thesis were carefully implemented to ensure that all preconditions were met, and a systematic process was followed for selecting projects, collecting project data, and ensuring that the data was valid.

It should be noted however, that no such threat exists in collecting PBAI data, as this data represented actual measurements taken for each project. Although the study was presented as a study on best practices of SOA development and participants were neither informed of the intent to construct a PBAI nor pressured to claim business agility, it is clearly feasible that some participants overstated (or possibly understated) the degree of attained business agility.

The use of a large number of properties and factors as independent variables is typically associated with collinearity issues that weaken conclusions. This issue to the inter-correlation that may exist among independent variables [90][112]. The grouping of attributes into business agility factors alleviated this condition and further collinearity validations were within the norms of acceptable bounds.

**8.13.2 Internal Validity**

Threats to internal validity are present due to the concentration of IBM projects and subjects that participated in the study. However, this risk has been mitigated by including additional set of projects that were executed by non-IBM personnel. Furthermore, the survey takers did not have any clue to the objective of measuring business agility as an outcome for SOA. The survey was presented as a process to collect SOA best practices which guaranteed little to no bias for answering business agility outcome questions.

**8.13.3 External Validity**

Given the low number of overall projects used in the survey process, the power of generalizing the results of this survey may not be perfect. However, the results for predictive powers of PBAI when compared to actual BAI results do show that the presented results are useful. Furthermore, the leave-one-out experiment demonstrated that the results were at least applicable across this broad sample of projects

**8.13.4 Conclusion Validity**

Given that only 39 SOA deployments were included in our study, threats to conclusion validity may be present due to the relatively low number of samples. However each SOA project involved many hours of hands-on data collection and a significant investment in time. To the best of our knowledge, no previous study (excluding simple surveys) has evaluated so many SOA projects as extensively as our study. Nevertheless, the relatively small sample size did impact our ability to explore certain factors such as the re-use factor, for which there was a problem with missing data. We therefore are careful not to make claims for factors that we were unable to fully analyze. Finally, the relatively low number of samples also impacted our ability to conduct adequate factor analysis which is far more effective when at least 10-15 samples are present for each factor that is to be explored.

### 8.14 Limitations

Like any statistical regression based analysis, we do not claim any causation relationship between the significant factors we established and business agility. To conclude that the presence of the right mix of identified factors in a given SOA solution will guarantee the achievement of business agility is a gross misunderstanding of the results of this research. On the other hand, the results of this research do establish a useful measurement for business agility that can be used to assess SOA solutions. More importantly, the developed PBAI model is useful in predicting the potential attainment of business agility and therefore serves as a useful tool for project executives, architects and stakeholders.

While the primary objective of this research is to investigate the relationship between SOA solutions and business agility, the impact of attaining business agility on complexity is a secondary component that will not be thoroughly investigated. Therefore, this research will not provide any metrics that can be used for determining the complexity of SOA solutions. Nor will this research establish any metrics that will provide information on how to estimate cost and effort of SOA projects. Nevertheless, some of the proposed metrics can be incorporated as part of an overall framework of cost and complexity estimation for SOA solutions while taking other cost and complexity drivers into the overall estimation effort. For example, functional requirements, skills, development practices, education, corporate culture, funding models and human behavior factors, to name a few, are all potential factors that impact cost and complexity, and their impact cannot be neglected in any effort that would attempt to estimate the cost and complexity of overall SOA projects.

The complexity indicators used to build the Solution Complexity Index (SCI) are not thoroughly investigated for varying level of complexities within the indicators themselves. For example, two projects reporting 500 functional requirements are treated the same from the functional requirements perspective. This is done primarily due to the lack of gathered complexity details. Nonetheless, the information is useful from a high level comparative perspective. Therefore, the SCI values do provide a relative high level complexity scale that is used to compare analyzed projects' complexity using the same

objective criteria. Individual scales on the SCI scale are meaningless outside the context of this research.

Finally, the results reported in this study may not be applicable to non-SOA projects. The fundamental question that needs to be asked is what makes a SOA system? If a given solution architecture uses many of the architectural components of a SOA system and enforces the alignment between business and IT, then one can argue that SOA is being used. In this situation, the results may be applicable since the identified factors that contribute to business agility do overlap with concepts that are applicable to many architectural styles. For example, the concept of Impact Analysis and its contributions to business agility is still relevant regardless of the underlying architectural style used to build a given solution.

### **8.15 Summary**

The PBAI model described in this thesis was developed through observing over 39 successful SOA deployments, some of which succeeded in achieving business agility and others which did not. The PBAI provides practitioners with a relatively quick method of assessing an early or late stage project to predict its potential for attaining business agility. All metrics used by the PBAI are relatively easy to collect. The empirical validation reported in this thesis substantiates many of the ideas that have previously been anecdotally claimed as best practices. However, the PBAI can be further enhanced through additional empirical evaluations that may take additional business agility perspectives into consideration. For example, it has been reported through our data collection process that training and industry knowledge contribute to business agility. This probably explains the model's moderate explanation of the variability of the data reported through adjusted R<sup>2</sup> of .64. Extending the BAI to include components of additional entities that contribute to business agility will be an interesting area for further research. Moreover, investigating the impact of reusability and loose coupling will be another area for exploration.

It is worth noting that striving for business agility as a result of SOA solutions comes at an expense, and often requires development of additional SOA components that need to be integrated to accomplish business agility. Therefore, the level of complexity of SOA solutions that produce higher levels of business agility may be higher than those that are less business agile as indicated by our correlation analysis of complexity and agility attainment.

## **Chapter Nine: Conclusions and Implications**

### **9.1 Introduction**

This final chapter provides a general summary of the work presented in this thesis. The chapter is divided into three major sections: summary of results, using PBAI in real SOA projects and future work.

### **9.2 Summary of Results**

In the proceeding chapters we presented the primary objectives of this research through a set of research questions. The research questions focused on the following topics:

- How can business agility be measured in completed SOA projects?
- What are the significant measurable technical and process factors of a SOA solution that are correlated with the attainment of business agility?
- How can proposed business agility metrics be used in to predict the business agility outcome of in-progress SOA solutions?
- Are SOA solutions that achieve business agility generally more complex than those that do not?

The first research question was answered through the creation of the business agility index (BAI) that was used to measure business agility attainment of completed SOA projects, chapter 3. The second question was answered through the identification of a set of business agility factors that had strong presence in projects that reported higher business agility results. Those factors included Service Oriented Architecture, Business Process Management, Impact Analysis, Loose Coupling and Governance. The use of regression analysis revealed that only three of these factors made a significant contribution to business agility. This analysis resulted in the creation and validation of a predictive model (PBAI) that was able to successfully predict business agility outcomes

for the majority of projects in our case study. The PBAI model is very useful since it allows project stakeholders to predict the business agility outcome before the project completion date, and to take corrective measures before it is too late in the development lifecycle. Finally, the last research question that investigated the relationship between attaining business agility and complexity resulted in a weak positive correlation between the two indicating that attaining business agility does not come for free. The additional solution components that inject such behavior are likely to require additional time and resources in order to attain agility benefits.

### **9.3 Using the PBAI in Real SOA Projects**

As previously explained, the PBAI is designed to provide an early prediction of the potential business agility outcome in SOA projects. All of the associated metrics are easily collected by IT personnel on the project. In addition to generally identifying projects with low potential for achieving business agility, the factors contributing to the BAI can be used to recommend areas of remediation. We randomly selected one project from the set of projects scoring PBAI values of 3 or lower for further analysis.

The selected project was found to score only 2 on the architectural score. An analysis of the individual attributes in the architectural category of the PBAI showed that on the positive side, this project (i) provided runtime support rules for driving the workflow, and (ii) included a service architecture which properly documented services, interfaces, and operations. On the other hand this project failed to provide runtime support for (i) alerts that summon human intervention, (ii) events that trigger the workflow, (iii) services that automate the workflow. (iv) dynamic resource allocation to alleviate bottlenecks, (v) monitoring and dashboarding to provide real-time feedback, and (vi) analytics that inform or enrich the process. These lacking areas could be recommended as possible elements for increasing the potential business agility of the SOA deployment. The fact that these factors could be analyzed early in the project, provides significant opportunities for making corrections in early planning stages.

## **9.4 Future Work**

There are plenty of opportunities to extend the work in this thesis given the wide scope of business agility. Opportunities are described below along with additional enhancements that can be applied to enhance the validity of results and extending the usefulness of the business agility predictive model.

### ***9.4.1 Additional Metrics***

This thesis has focused on understanding the dynamics of business agility within the context of SOA and architectural traits. Given that business agility may be impacted by other factors, there is a need to investigate additional dimensions and their impact on business agility. Dimensions may include, but not limited to, organization, skills, training, reuse, and information architecture. As indicated earlier, we could not make any conclusions regarding the role of reuse in attaining business agility. This was due to the fact that our collected data has several missing entries for the reuse section in the survey. A new round of data collection and better emphasis on the quality of collected data could provide the link required for drawing conclusions regarding this often touted factor in SOA implementations.

Similarly, information architecture and use of canonical models is a hotly debated area to the usefulness of SOA and its contribution to business agility. Our survey respondents did not provide enough information for the proper analysis of this dimension in attaining business agility.

### ***9.4.2 Extensive Empirical Validation***

As noted earlier, we managed to collect data from 39 SOA projects but only used 32 projects to carry out the empirical validations of this research. 39 projects are considered significant in the software engineering research given that other researchers used 10 subjects or less to conduct their analysis. However, there is still a significant opportunity to extend the data collection process to a larger number of projects. The data collection

step is not a trivial task and would require a significant amount of planning to select the right candidate projects. Appendix B provides a complete set of questions that can be used for further data collection procedures. The use of a significant sample would strengthen the internal validity of this research and allow us to generalize the results with better accuracy.

#### ***9.4.3 Embedding in Existing Methodology and Tools***

For any set of metrics suite to be successful, it should be easy to collect the required data where metrics can be calculated and used. The business agility factors identified in this research can be divided into two sets. One set that requires a human interviewer with the project personnel to ask certain questions on various roles and activities. The second set can be easily automated through the collection of data that is pertinent to deliverables that should have been created as part of the SOA development lifecycle. To make full use of the research outlined in this thesis, a SOA methodology can be enhanced to include the set of activities required to collect the data required to produce the business agility predications as evidenced by the PBAI model.

An online tool can be created to automate the human activity process and allow for importing the deliverables required to detect the presence of lack of the required business agility factors. The tool will generate a report based on the input to provide business agility recommendations based on the calculations of the PBAI values. The report will provide project stakeholders with the diagnosis regarding the business agility health check for the project. Moreover, the report will include a set of recommendations on how to enhance or attain the desired business agility outcome.

# APPENDIX A GQM-MEDEA

## Introduction

In order to accomplish the objectives of this research, Goal Question Metric: METric Definition Approach (GQM-MEDEA) [89] was used as the overall governing process. The GQM-MEDEA is well suited for the purposes of this research due to the similarities between what the method provides and the objectives of this research. The GQM-MEDEA[89] approach, illustrated in Figure 29, provides high level steps that formulate the basis for defining and validating the right set of software engineering metrics. The approach assumes and encourages the gathering of data from different sources such as management and project teams. It also builds on experiences gained from different domains and projects.

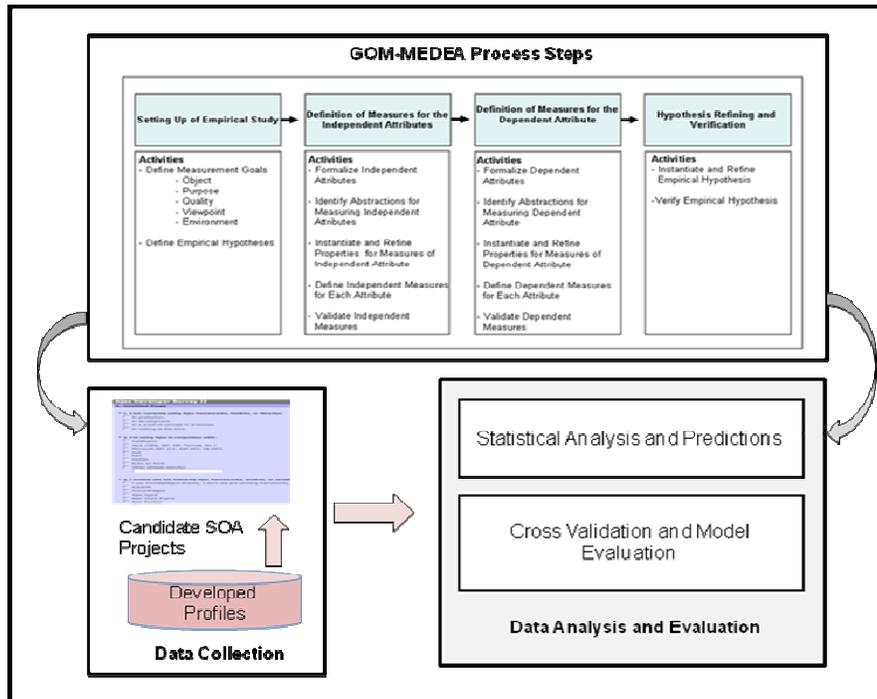


Figure 29 Overall Research Methodology that Shows Application of GQM-MEDEA

Process Steps to Our Research Questions – Author’s Image

Briand et al. [89] provide a detailed overview of the GQM-MEDEA approach. We present a high level overview of the approach in this section. In subsequent sections we discuss how it was used as described in Figure 30 to help with the definition of hypotheses and metrics identified in this research. The methodology used in this research consists of three primary steps:

### **1- GQM-MEDEA Process Steps**

The GQM-MEDEA offers a set of recommendations and guidelines on how to conduct activities to help with the definition of properties under investigation. It also provides a structure for the identification of entities and attributes impacting overall goals. The outcome of this step is a set of hypotheses and metrics that will be investigated further and empirically validated as part of steps 2 and 3.

### **2- Data Collection**

This step generates the required data that were used during the analysis step to empirically validate the hypotheses stipulated in step 1. The outcome of this step is the definition of a project profile for participating projects in the data collection step. In addition, an extensive data collection survey is created to meet the data demands of the empirical validation for every hypotheses stipulated in step 1. The data collection survey is executed to collect required data for the purposes of this research.

### **3- Data Analysis and Evaluation**

This step addresses the empirical validation of the hypotheses stipulated in step 1 based on the data that was collected during step 2. GQM-MEDEA provides guidelines that influence the selection of the proper statistical methods that can be used. The outcome of this step is the validation of the business agility index that is used to assess business agility outcome of SOA deployments. In addition, this step

provides the empirical validation of the stipulated hypotheses that were outlined in step 1. The empirically validated hypotheses are then grouped into a model that is used to predict business agility for in progress SOA projects. Finally, the results were validated and evaluated to ensure the stability of the model and its usefulness in predicting the expected outcome.

### **GQM-MEDEA Process Steps**

The GQM-MEDEA process consists of four main activities that lead researchers to the discovery of the factors and attributes that may impact the area under investigation. The four activities, their applicability and application to this research is documented below. Figure 30 describes the major steps that are defined by GQM-MEDEA that are later explained in the following sections.

### **Setting Up of Empirical Study**

This step consists of two primary steps:

#### **1- Define Measurement Goals**

During this step, objectives are documented and measurement goals are listed for further investigation. To reach the measurements goal in a systematic manner, the GQM templates [118] were used. The template provides guidelines for defining goals and refining them to reach concrete questions that lead to definition of measures.

Here is a summary of templates and how they can be used:

- **Object of study:** Determines entities that need to be modeled and the hypotheses that are relevant.

The identified entities will make up the majority of a SOA solution such as: BPM, connectivity, business analytics, etc. where architectural decisions will be present and represent the major points of interest for this research.

- **Purpose:** Defines the measures to be used in the experiment.

The defined measures will be used to predict the overall impact on business agility. Defining prediction as the purpose will have direct impact on the data types to be collected and its accuracy to reach meaningful predictions. Moreover, the amount of data to be collected will be also impacted since prediction requires more linear data where relationships can be statistically detected [89]. Furthermore, prediction models require empirical validations in order to build credibility in the predictive powers of models.

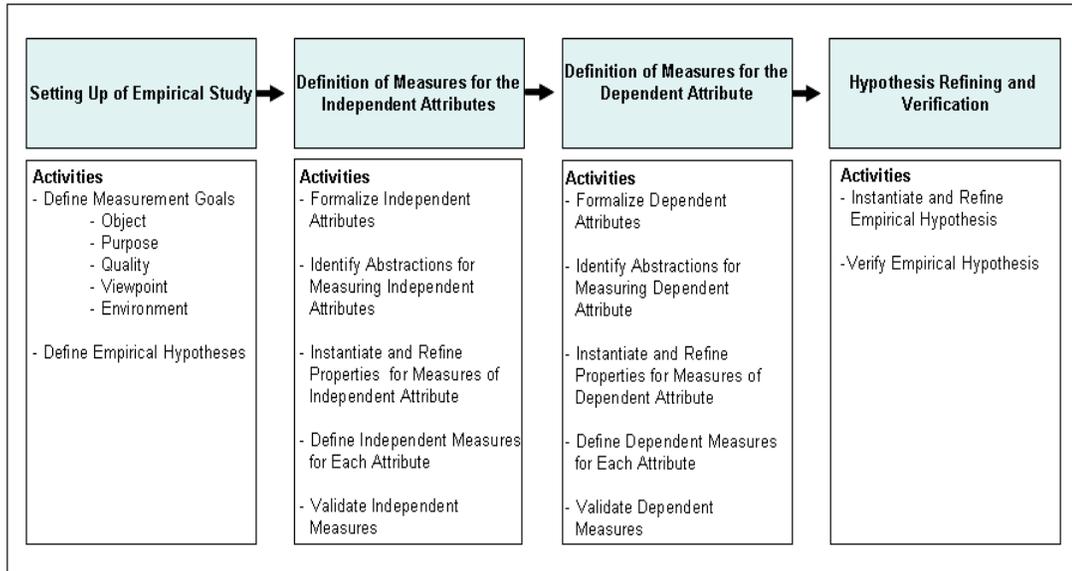


Figure 30 QM-MEEDA Major Steps and Main Activities - Author's image

- **Quality:** Determines dependent attribute that is at the center of the prediction model along with the hypotheses that stipulate relationships between dependent and independent variables.

- **Viewpoint:** Documents the perspective from which predictions are made.

In our study, business agility as a dependent variable may be viewed by organizations as a set of attributes that combine both technical and business entities. However from a customer’s perspective, business agility may be viewed as how fast a company can react to fixing a situation that directly impacts a customer, e.g. power interruption.

- **Environment:** Determines the context and scope of the study and provides the context to which results are valid and how can they be generalized.

Executing this step provides the outcome that is described in Table 26. The table shows that the objective of the study is to provide a set of metrics for SOA solutions that can be used in predicting business agility outcome.

<b>Goal</b>	1
<b>Object of Study</b>	Creation of a new set of metrics that can provide clues to the business agility benefits that can be expected from SOA solutions.
<b>Purpose</b>	Prediction
<b>Quality Focus</b>	Business agility as a result of building SOA solutions
<b>Viewpoint</b>	Project leaders and architects
<b>Environment</b>	Completed SOA projects over the past 3 years for IBM customers

**Table 26 Measurement goals per GQM-MEDEA**

**2. Define Empirical Hypotheses**

Research hypotheses were formulated based on our observations, intuitions and experiences in building SOA solutions. The hypotheses were verified empirically to show which hypotheses may hold the test by showing strong correlations between

independent and dependent variables. For example, an empirical hypothesis may indicate the relationship between business services and business agility by stipulating that the larger the number of business services, the higher business agility that can be attained from building a SOA solution. According to the GQM-MEDEA method, hypotheses may further get refined towards the end of the process once measurements are clearly defined.

Stipulating hypotheses as part of this step to determine the impact on our dependent variable, business agility, resulted in the identification of the set of potential independent variables and their attributes. For the objectives of this research, it is stipulated that building SOA solutions results in attaining better business agility for organizations. It is often the case that organizations adopt SOA methodologies to reach better business agility levels. Therefore, business agility is the corporate objective for our study and would translate to some attributes that must be measured to define what business agility means. The definition of research hypotheses will tie in the independent variables to the dependent variable and is the subject of our investigation.

### **Definition of Measures of the Independent Attributes**

The objective of this step is to define the set of independent attributes or variables that will be correlated to the dependent variable. This step calls for the formalization of the independent variables via sets of generic properties that characterize their measure. For example, do independent variables have any mathematical properties such as additivity? Briand et al. [89] point to the lack of proper definition of a universal set of properties in the software engineering field with only few properties that received enough attention such as coupling and cohesion. Therefore, it becomes important to define important independent variables to better understand their meanings. The authors of GQM-MEDEA advocate the use of a property-based software engineering framework [120] to formally define and validate the independent variables. It is also worth noting that while the GQM-MEDEA advocates

the use of formal definitions of the mathematical properties of the measures, the authors of the method acknowledge that not all measures may be able to be represented in a formal way due to the human-intensive nature of the definition activity.

Given the fact that the majority of identified independent variables are of human intensive nature, formal definition and validation of the proposed metrics is not feasible and was not conducted for this study. The lack of formal definitions does not take away from the credibility of the identified properties nor does it affect the results of the research. As Briand [120] points out that the careful inspection of the collected data and how it was used as part of the overall process may compensate for the lack of mathematical representation.

For the purposes of this research, business agility metrics or predictors that are based on architectural decisions and other attributes are considered independent variables. The independent variables values will vary for each SOA solution depending on the requirements and business objective for every solution. Making the right architectural decisions is always a trade-off and a balancing act among many factors that every architect need to consider in order to achieve business goals. Based on our experience and observations of real SOA projects, each of the major architectural decisions that are represented through our metrics may have the potential to impact the business agility of a SOA solution on its own. However, the combined effect of such architectural decisions or metrics is far more significant than any individual architectural decision.

### **Definition of Measures of the Dependent Attribute**

The steps outlined for this level mimic those of the previous step for the independent attributes with the emphasis on the dependent attribute instead of the independent attribute. Briand et al. point to potential difficulty of representing dependent attributes through mathematical measures. In such scenarios, the careful inspection of the

collected data and how it was used as part of the overall process may compensate for the lack of mathematical representation. However, regardless of the mathematical representation of dependent variable, empirical validation steps were applied.

As stated earlier, the objective of this research is to measure the impact of various architectural decisions and other attributes on the business agility outcome in projects that use SOA as the primary method for building solution architectures. Therefore, business agility is the dependent variable that we devise a measurement approach for to help with establishing the relationship with SOA solutions. The business agility dependent variable values are not binary. In other words, we do not consider benefits of SOA projects as either delivering business agility benefits or not. The business agility variable is more of a continuous variable that will have different values based on the outcome of SOA projects and the architectural decisions that comprise the overall SOA solution. The values can range from low business agility benefits to high business agility benefits with some midpoint to indicate medium business agility level. This classification would make the dependent variable of ordinal scale. Throughout the text, the dependent variable will be referred to as Y.

### **Hypothesis Refining and Validation**

The refinement step implies changes to our originally stipulated hypotheses. The refinement to the original hypotheses was a direct result of the early analysis of the empirical results.

## APPENDIX B SURVEY QUESTIONS

### IBM Academy of Technology Conference 3<sup>rd</sup> SOA Best Practices with focus on Business Agility (c) copyright IBM 2010

#### Instructions

The questions in this survey refer to a "project." We are looking for experiences on projects with clients (internal or external). The key is client experiences, not just personal opinion. Projects can also include solution development and CIO projects, with a variety scopes.

Your presentation case study can be a merger of several projects. However, you should create a separate survey for each project.

2. The questions span the project lifecycle. Meet with your colleagues, sales teams, customer staff and the account team to determine the best responses. You can review the PDF version so that your time on-line is minimized.
3. We are mostly interested in projects that are related to IBM's strategies:
  - SOA, BPM, Cloud, Industry Solutions
  - Globally Integrated Enterprise, Business Agility, Smart Planet
  - IBM products and services
  - etc.
4. If the question 'does not apply' or 'is not applicable' to your project, then skip the question.
5. If the questions asks for an answer and your are not sure, select Unknown.
6. The survey should take a few hours. You may resume editing at your convenience. Please save / print copies just in case there are system issues.
7. We will correlate the results and present them during the Academy conference.
8. All client traceable information will remain IBM Confidential.
9. Have fun and thanks again!

## Survey Overview:

Project and Team  
Governance  
Requirements  
Architectures  
BPM / SOA  
Cloud  
Standards and Open Source  
Information Architecture  
Analytics  
Implementation  
Operations and Change  
Product Feedback  
Lessons Learned & Best Practices  
Assets  
Business Agility  
Finished

## Project and Team

Let's gather some background on your, the project background, the team, etc.

---

1) Please enter your IBM intranet email address (not Notes).

---

2) What is your primary working geography?

- North America
- South America
- Europe
- Middle East
- Africa
- China
- India
- Asia
- Australia
- World wide
- Other (please specify)

If you selected other, please specify

---

3) What is your primary IBM line of business?

- SWG
- GTS
- GBS
- S&D
- STG
- Research
- CHQ

4) What is the name of the client where you performed the project?

---

5) Select the best industry that aligns with your project.

- Automotive
- Aviation, Aerospace and Defense
- Banking
- Chemical and Petroleum
- Construction
- Consumer Packaged Goods
- Education
- Electronics
- Energy and Utilities
- Financial Services
- Food Services
- Government
- Health Care
- Industrial Products
- Insurance
- Life Sciences and Pharmaceuticals
- Manufacturing
- Media and Entertainment
- Professional, Technology and Business Services
- Real Estate
- Retail
- Telecommunications
- Travel and Transportation
- Wholesale Distribution and Services
- Multiple Industries

Additional comments

---

6) What is the approximate revenue of your project to IBM in \$US?

	\$0	\$1 to \$50k	\$50k to \$100k	\$100k to \$500k	\$500k to \$1M	\$1M to \$10M	\$10M to \$100M	\$100M to \$1B
Software	<input type="radio"/>							
Hardware	<input type="radio"/>							
Services	<input type="radio"/>							
Solutions	<input type="radio"/>							

7) In which geographies was this project designed, implemented or run?

- North America
- South America
- Europe
- Middle East
- Africa
- China
- India
- Asia
- Australia
- Other (please specify)

If you selected other, please specify

---

8) How would you characterize the project?

- SOA
- BPM
- Cloud
- SO
- Smart Planet
- Legacy Modernization
- Industry Solution
- Consulting

- Other (please specify)

If you selected other, please specify

---

9) What lifecycles did your project address?

- Requirements
- Business Architecture
- Technical Architecture
- Systems Engineering
- Development
- Deployment
- Testing
- Execute
- Manage
- Monitor

Additional comments

---

10) What is the scope of your project?

- Client POC / POT / Demo
- Client project / application
- Client program / line of business
- Client enterprise / organization
- Client transformation
- IBM internal solution
- IBM CIO
- Other (please specify)

If you selected other, please specify

---

11) What is the duration of your project?

- less than 3 months
- 3 months to 1 year
- 1 to 2 years
- more than 2 years

12) Was this your first BPM or SOA project?

- Yes
- No
- Unknown

13) Was this your client's first BPM or SOA project?

- Yes
- No
- Unknown

14) How many client and IBM technical people were on the project?

enter a number:

---

15) How many client and IBM business people were on the project?

enter a number:

---

16) Were the following team member roles defined and used on your project?

	Yes	No	Unknown
Enterprise Architect	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Application/SOA Architect	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Data Architect	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Security Architect	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Analyst	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Modeler/Architect	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Project Manager	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Testing Lead	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Lead	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Operations	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Release Manager	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

17) Did team members have appropriate training?

	Yes	No	Unknown
Business	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architects	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Engineers	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Development	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Testing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Operations	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Project Management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

18) What was the primary method for training team members?

	Class Room	Conference	Online / Self Paced	On The Job	None
Business	<input type="radio"/>				
Architects	<input type="radio"/>				
Engineers	<input type="radio"/>				
Development	<input type="radio"/>				
Testing	<input type="radio"/>				
Operations	<input type="radio"/>				
Project Management	<input type="radio"/>				

19) When defining and changing the project, did you have interaction between IT and Business people?

- Yes
- No
- Unknown

20) How frequently did the IT and Business people interact?

- every day
- few times per week
- few times per month
- quarterly
- yearly
- unknown
- never

21) What was the quality of that interaction?

- High Quality - constructive dialog about business strategy and how they map to IT requirements
- Narrowly focused - refining solution requirements without much insight to business intent or future inflection points
- Incidental - interaction primarily focused on updating the Business on development progress, but with little focus on requirements refinement
- Contentious - frequent non-constructive criticism in either direction
- Other (please specify)

If you selected other, please specify

---

## Governance

Let's gather some background on the project methodologies, tools, governance, techniques, etc.

---

22) What methods and work products did you use on the project?

- Team Solution Design
- Unified Method
- RUP
- SOMA
- CBM
- ISIS
- Client Specific
- Agile
- SGMM
- Ascendant
- Model Driven Architecture / Design
- Visualization
- Industry specific ( e.g. DODAF, FEA)
- None
- Chaos
- Unknown
- Other (please specify)

If you selected other, please specify

---

23) What Team Solution Design or Unified Method work products did you create?

- Architecture Overview
- Architecture Decisions
- Context Diagram
- Operational Model
- Issues, Risks, Concerns
- Use Cases Scenarios
- Component Model
- Non Functional Requirements
- Estimation Report
- Functional Requirements
- Other (please specify)

If you selected other, please specify

24) Are the general governance processes documented, understood and applied?

	Yes	No	Unknown
Documented	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Understood	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Applied	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

25) Governed services are those services that are documented properly with clear objectives, proper funding and ownership. They are also properly registered in a registry and monitored proactively at runtime for achieving their service level agreements. What percentage of business services are governed properly within the project?

enter a number: \_\_\_\_\_ %

26) What percentage of business services are in a repository (e.g. WSRR or RAM) and can be easily discovered?

enter a number: \_\_\_\_\_ %

27) What percentage of business services are NOT governed properly within the project?

enter a number:

\_\_\_\_\_ %

28) Is there a defined and deliberate process for sunseting services?

- Yes
- No
- Unknown

## Requirements

Let's gather some information on the project requirements.

\_\_\_\_\_

29) What is the total number of functional requirements for this project?

enter a number:

\_\_\_\_\_

30) What are the top 5 functional requirements?

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

31) What are the top 5 Non-functional requirements?

\_\_\_\_\_  
\_\_\_\_\_

32) Do you have a standard set of work products to capture functional and non-functional requirement?

- Yes
- No
- Unknown

33) Where would you go to find business requirements?

- strategic plan
- interviews
- workshops
- industry SME experts
- CBM engagements
- Business Process
- Business Architecture
- Other (please specify)

If you selected other, please specify

---

34) What techniques did you use to trace requirements, to architecture, to implementation , to tests and to operations?

	Yes	No	Unknown
ReqPro	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
DOORS	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Spreadsheet	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Requirement Traceability Verification Matrix (RTVM)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Rational Team Concert	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Rational Requirements Composer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Other	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
No requirements tracing	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
RAM Repository	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Proprietary system	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

35) Did you use business processes and use cases to document your requirements?

- Yes
- No
- Unknown

36) How are requirements changes being addressed in your project?

- Well defined process
- Loose process
- No process
- Unknown

37) What are the primary challenges in collecting requirements?

- No clear vision
- Stake holders are not committed
- No baseline exists
- Lack of skills to solicit requirements
- Lack of methods and tools to solicit requirements
- None
- Other (please specify)

If you selected other, please specify

---

38) Do stakeholders assign a priority or weighting to each functional and non-functional requirement?

- Yes
- No
- Unknown

39) What do each of the non functional requirements leverage?

	existing infrastructure in the enterprise	components to be added to the enterprise infrastructure	components to be used only for this project	Unknown
Security	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Performance (resp. time)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Load balancing	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Scalability	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
System monitoring	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

40) Are the following roles defined and assigned in the project?

	Yes	No	Unknown
Requirement Manager	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Change Manager	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Configuration Manager	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

## Architectures

Let's gather some background on the project business architectures, technical architectures, etc.

For business architecture, we are generally referring to the items created before development starts.

41) Please indicate how you address business strategy and capabilities:

	Created?	Applied?	Governed?	In Repository?	Traced across lifecycle?
business strategy	<input type="checkbox"/>				
business capabilities	<input type="checkbox"/>				

42) Please indicate how you address requirements and use cases:

	Created?	Bounds the scope of project?	Governed?	In Repository?	Traced across lifecycle?
requirements	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
use cases	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

43) Please indicate how you address legacy analysis:

	Human effort?	Automated?	Governed?	In Repository?	Traced across lifecycle?
code analysis	<input type="checkbox"/>				
data analysis	<input type="checkbox"/>				

44) Please indicate how you address IT architectures:

	Created?	Applied?	Governed?	In Repository?	Traced across lifecycle?
Enterprise architecture	<input type="checkbox"/>				
Deployment architecture	<input type="checkbox"/>				
Security architecture	<input type="checkbox"/>				
Application architecture	<input type="checkbox"/>				
Context diagrams	<input type="checkbox"/>				
Component diagrams	<input type="checkbox"/>				
Architecture overview	<input type="checkbox"/>				

45) Please indicate how you address BPM / SOA architectures:

	Created?	Reused?	Governed?	In Repository?	Traced across lifecycle?
Business Processes	<input type="checkbox"/>				
Business Rules	<input type="checkbox"/>				
Business Services	<input type="checkbox"/>				
Business Events	<input type="checkbox"/>				
KPIs	<input type="checkbox"/>				

46) Please indicate how you address UI architectures:

	Created?	Reused?	Governed?	In Repository?	Traced across lifecycle?
Screen mockup	<input type="checkbox"/>				
Navigation flow	<input type="checkbox"/>				
Business Dashboards	<input type="checkbox"/>				

47) Please indicate how you address information architectures:

	Created?	Reused?	Governed?	In Repository?	Traced across lifecycle?
Business Glossary	<input type="checkbox"/>				
UI Data Models	<input type="checkbox"/>				
Service Data Models	<input type="checkbox"/>				
Messages	<input type="checkbox"/>				
Transformation Maps	<input type="checkbox"/>				
Enterprise Information Architecture	<input type="checkbox"/>				

48) How easy or difficult is it for you to create parts of the business architecture?

	A breeze - easy	Takes Time - medium	Painful - hard
Strategy and Capabilities	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Requirements and Use Cases	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Legacy Analysis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
IT Architectures	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Rules	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Information Architecture	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
UI Mockups	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Glossary	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Transforms / Mappings	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

49) Where did you publish and overall view of the business architecture?

- Asset Repository with Browser Access
- All over the place
- Published to a shared drive ( C: )
- Sent through email
- Other (please specify)

If you selected other, please specify

---

50) What is the scope of the business architecture?

- Enterprise wide
- Line of business
- Project based
- Business Use
- IT Use
- Other (please specify)

If you selected other, please specify

---

51) When using a technology based approach to creating business architecture, what is most important to you?

- Powerful Tools \_\_\_\_\_
- Easy to use Tools \_\_\_\_\_
- Integrated Tools \_\_\_\_\_
- Content generating Tools \_\_\_\_\_

52) How did you rank the quality of the business architecture? Answer all drop downs with a separate choice.

- Ability to generate code and implementation artifacts \_\_\_\_\_
- Ability to do impact analysis through traceability \_\_\_\_\_
- Ability to browse through pictures and documents quickly \_\_\_\_\_
- Ability to get through governance quickly \_\_\_\_\_
- Ability to reuse the architecture \_\_\_\_\_
- Ability to correlate to business agility \_\_\_\_\_

53) Who participates in the creation and approval of the business architecture?

- Business Executive
- IT Executive
- Business Analyst
- Operations
- Architects
- Engineers
- Development
- Testing
- Users
- Other (please specify)

If you selected other, please specify

---

54) Does the overall architecture include the capability of reporting and analysis of information to enable business monitoring and decision making?

- Yes
- No
- Unknown

55) Do you maintain Architectural Decisions?

- Yes
- No
- Unknown

56) Who signs off on the architecture deliverables before project startup?

	Yes	No	Unknown
Customers/Constituents	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
External partners	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business users	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Project sponsors	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Business Analysts	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architects	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
System Engineers	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Development	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Test	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Operations	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Stakeholders	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

57) Will the architecture be developed into an IBM standardized reference architecture for reuse on future projects?

- Yes
- No
- Unknown

58) Select the 3 most important security requirements implemented in your client's project today. Selections should be made in order of importance.

- People and Identity Management \_\_\_\_\_
- Data Access and Information Protection \_\_\_\_\_
- Application Vulnerability Testing \_\_\_\_\_
- Security Process Compliance & Audit \_\_\_\_\_
- Capabilities \_\_\_\_\_
- Network, Server, Endpoint \_\_\_\_\_
- Firewalling/Patch management \_\_\_\_\_
- /Virus/Intrusion Detection \_\_\_\_\_
- Physical facility security \_\_\_\_\_

## BPM / SOA

Let's gather some information on the BPM and SOA aspects of your project.

---

59) How well did the team understand the overall business context, intent and potential for market inflections to drive future changes in the execution of related business processes?

- Highly
- Moderately
- Poorly
- Unknown

60) How are business processes modeled on your project?

- Whiteboarding
- Rational Software Architect
- WebSphere Business Modeler
- Lombardi
- Microsoft Power Point
- Microsoft Visio
- WebSphere Compass
- Rational Requirements Composer Sketcher
- Rational System Architect
- 3rd Party Tool
- Lotus Symphony
- No modeling was performed
- Other (please specify)

If you selected other, please specify

---

61) What percentage of implemented business processes on your project were modeled?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

62) Was simulation used to achieve optimal business processes?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all
- Unknown

63) Were business process performance results and KPIs monitored in real time, measurable and enable taking actions?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all
- Unknown

64) How do you characterize business processes that were implemented on this project?

	Yes	No	Unknown
Invoked Human Tasks	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
New Function Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Accessed Legacy Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Accessed Third Party Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Used Business Rules	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Integrated with Other Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Called an ESB	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

65) Can business processes be easily changed and deployed quickly with new outcomes?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all
- Unknown

66) What is the average number of human tasks per business process that are modeled as part of the project?

enter a number:

---

67) How many business rules/decision tables/rule sets are modeled as part of the project?

enter a number:

---

68) What is the average number of process integration points for those processes that interact with other processes within the project?

enter a number:

---

69) Business rules can be either hard coded in the actual code of the service or the underlying components that may implement the service, e.g. an EJB, or may be stored in a more flexible manner, e.g. business rules engine, to facilitate the dynamic application of business rules during changing runtime conditions. To what extent did you use a rules engine such as ILOG to externalize rules from your business logic?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all
- Unknown

70) Is the use of an process engine accompanied with business rules engine important for SOA projects within the organization?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all
- Unknown

71) What is the total number of business processes that were implemented as part of this project?

enter a number:

---

72) Does the architecture provide runtime support for the execution of business processes such as:

- rules that drive the workflow
- events that trigger the workflow
- services that automate a task
- alerts that summon human intervention
- dynamic resource allocation that alleviates bottlenecks
- monitoring and dashboarding that provide real time feedback
- analytics that inform or enrich the process condition
- None, there is no runtime support for business processes.

73) Did you create a service architecture that properly documents services, interfaces, operations, etc.?

- Yes
- No
- Unknown

74) What is the total number of services that were implemented as part of this project?

enter a number:

---

75) A Candidate Service can later be verified and turned into true services through well-documented criteria such as business alignment, composable, stateless, etc. What is the total number of candidate services?

enter a number:

76) What percentage of services are reused in this project?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

77) An IT Service is an abstraction on a technical implementation (e.g. EJB, CICS) and is not necessarily a business services. What percentage of services are created as IT services?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

78) An Adaptable service is a service that possesses the right set of policies, business rules or infrastructure that would allow the service to adapt dynamically based on varying conditions and policies. What percentage of services are considered adaptable?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

79) An Atomic Service is a service that can stand on its own and does not require the use of other services (as opposed to Composite services). What percentage of services are atomic for your project?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

80) What percentage of services are considered composite for this project?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

81) Stateless services are services that are totally independent and do not retain instance state across service invocations. Services that use database lookups are also considered stateless. What percentage of services created for this project are considered stateless?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

82) Service endpoints are usually stored in a WSDL document that provides the attributes regarding how to invoke a service. Service endpoints can be looked up statically from a well-know WSDL document through design time, or can be looked up dynamically through a registry mechanism. What percentage of services are going to be looked up statically within this project?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

83) What percentage of services and business processes (or sub-processes), that were themselves exposed as invocable services, were implemented as part of this project?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

84) There are many service realization patterns that can be used for exposing and using services including the two primary patterns of Direct Exposure (DE) and indirect exposure that is referred to as Access Services. DE refers to exposing current IT systems or modules as a service without having to go through an intermediary component. For example, a stored SQL procedure could be turned into an information service directly by wrapping it through a web service and exposing the web service to consuming clients. What percentage of services are going to be realized through direct exposure?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

85) Access Services realization refers to exposing current IT systems or a module as a service by going through an intermediary component such as an EJB. What percentage of services are going to be realized through access services?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

86) Services can be invoked directly through point to point, or through an intermediary like an enterprise service bus that virtualizes the access to services. What percentage of services are going to be invoked through the Enterprise Service Bus or through an intermediary broker?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

87) What percentage of services are going to be invoked through point to point connectivity?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

88) What percentage of services have multiple versions in production?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

89) What percentage of services have one interface (operation)?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

90) What percentage of services have more than 1 and less than 6 interfaces (operations)?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

91) What percentage of services have more than 6 interfaces (operations)?

- More than 80%
- 60% - 80%
- 40% - 60%
- 20% - 40%
- Below 20 %
- Unknown

92) Are services and service interfaces stable in general? Stable means they change very rarely due to their use of standards-based interfaces and well-defined and approved message structures?

- Yes
- No
- Unknown

## Cloud

Let's gather some information on your cloud approach.

---

93) Is your project leveraging a cloud environment/offering? (If you answer Yes, you will see more cloud questions)

- Yes
- No
- Unknown

94) Is your client's cloud project using a public cloud service?

- Yes
- No
- Unknown

95) Is your client's cloud project leveraging/creating a private cloud service?

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)
- Business Process as a Service (BPaaS)
- No
- Unknown

96) In your client environment, what are reasons that prevent or slow down cloud adoption? Please select the top 5 reasons.

- Security (Data security/privacy, Compliance & Regulations/Governance & Auditability)
  - Lack of standardization
  - Performance concerns - Reliability and High Availability
  - Lack of Modularity (Stand-alone or loosely coupled nature)
  - Budgetary considerations (Difficulty forecasting, TCO, Lack of quick ROI, IT spending slowdown)
- Lack of clear value proposition
- Complexity of technology support and needed skills after cloud transformation
  - Inability to customize and integrate with internal IT
  - Loss of internal control over IT resources
  - Market and technology immaturity (lack of enough major provider choices)
  - Lack of stringent SLA adherence (including response times)
  - Software licensing practices
  - Legacy migration
  - Risk management (vendor lock-in, vendor long term viability, hard to bring back in-house)
- Lack of Skills
- Other (please specify)

If you selected other, please specify

---

97) Pick the 3 most important ITIL processes in cloud service management for your client's project today. Selections should be made in order of importance.

- Service Strategy (Portfolio, Risk, Demand, Financial Mgt) \_\_\_\_\_
- Service Design (Catalog, SLA, Capacity, ... Mgt) \_\_\_\_\_
- Service Improvement (Reporting, Measurement) \_\_\_\_\_
- Service Transition (Asset, Change, Release, Testing) \_\_\_\_\_
- Service Operation (Event, Incident, Problem, Request, ...) \_\_\_\_\_

98) Is your client using a standardized technology solution stack for cloud service delivery?

- Yes
- No
- Unknown

## Standards and Open Source

Let's gather some information on the so called openness of your projects.

---

99) Are open standards important to your client?

- Yes
- No
- Unknown

100) List the important open standards for your project.

---

---

---

---

101) Is there any integration of industry regulatory compliance requirements (e.g HIPAA) in your project?

- Yes
- No
- Unknown

102) List the important open source tools/products for your project.

---

---

---

## Information Architecture

Let's gather some information on, well, information data, models, etc.

103) Have you partitioned the data model in different business information domains, each with separated business ownership, so that a change in a domain does not propagate to the other domains?

- Yes
- No
- Unknown

104) Have you leveraged your business data to be more agile (i.e. to identify new business services, new business processes or new behaviors to address market conditions)?

- Yes
- No
- Unknown

105) What drives the Information Architecture?

- Service, process and rules definition
- UI Data
- Enterprise logical data model
- Business glossary
- Partner message mappings
- COTS product specifications
- Other (please specify)

If you selected other, please specify

---

106) Which industry standard data models available, from IBM or industry consortiums, are you using?

Answer: \_\_\_\_\_

107) Are you implementing a data context that establishes an approach to the categorization of data assets using taxonomies and other descriptive information?

- Yes
- No
- Unknown

108) Are you implementing a data sharing that describes the access and exchange of data, where access consists of recurring requests (such as a query of a Data Asset), and exchange consists of fixed, recurring information exchanges between parties?

- Yes
- No
- Unknown

109) Are you separating the data design model from the data sharing model and data context, so that they can vary independently while ensuring consistencies?

- Yes
- No
- Unknown

110) Are you implementing a data design that provides a means to uniformly capture the business semantic and syntactic structure of data?

- Yes
- No
- Unknown

111) Are you implementing information driven processes which behavior automatically adapts to information content changes?

- Yes
- No
- Unknown

112) Is your data model managed at the Enterprise Architecture level and consistent with the other aspects of EA (business model, infrastructure model and application model)?

- Yes
- No
- Unknown

113) Are you implementing a canonical model?

- at Enterprise level
- at Community of Interests level ?
- For Business partners (B2B) ?
- none

114) Is the solution data model created from existing application-specific data models?

- Yes
- No
- Unknown

115) Have you introduced flexibility in the data model so that business processes or services are not impacted by evolution of entity characteristics or attributes?

- Yes
- No
- Unknown

116) Which variability techniques for information and data model are you using?

- Polymorphism,
- Back box approach such as with xsd:any.
- Extensibility with name/value pairs,
- Characteristic Specification/Value Pattern (e.g.. Telco SID model)
- Separation of UI messaging from Legacy information
- Other (please specify)

If you selected other, please specify

---

117) Do you formally measure the impact of the any of the following?

	Yes	No	Unknown
Information model changes on services and process models	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architecture changes on project timeline	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Service model changes impact on processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Process changes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

118) What assets do you derive from a business object model?

	Yes	No	Unknown
XML/XSD	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Rules	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
UI	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Events	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architecture	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Data Architecture	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Data Sources	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Transformations	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Message Definitions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

119) Do you have separate data models for the following?

	Yes	No	Unknown
MDM	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Data warehouse	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Operational Data Stores	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Content Management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Datamarts	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Cubes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Archives	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

120) For global coherence purposes, is your business rules vocabulary/artifacts derived from a Business Object Model?

- Yes
- No
- Unknown

121) Do you have a formal method to evaluate the impact of internationalization of the data model (addressing size, performance, integration of mainframe and distributed environments, functional impact on code – sorting and searching, etc..)?

- Yes
- No
- Unknown

122) How do you capture your information data model?

- XML
- UML
- Entity/relationship diagram
- Text
- Unknown
- Other (please specify)

If you selected other, please specify

---

123) Do you have data quality and master data management tools/methods to ensure that the growth of your information model will not degrade its consistency?

- Yes
- No
- Unknown

124) Did you implement a business glossary?

- Yes
- No
- Unknown

## Analytics

Let's gather some project information on analytics, etc.

---

125) Did you use dashboards as part of your solution to track and analyze business results? (If you answer Yes, you will see more analytics questions)

- Yes
- No
- Unknown

126) Since you answered yes, are dashboards customizable -- can you easily create new and different dashboards for other business user and IT roles?"

- Yes
- No
- Unknown

127) How many dashboards are included in your solution?

enter a number:

---

128) Which LOBS did your dashboards support?

- Financial
- HR
- IT
- Marketing
- Sales
- Customer Relations
- Security
- Operations
- Other (please specify)

If you selected other, please specify

---

129) What are the identified roles that are going to access the dashboards?

- Executives
- Middle Management
- Business People
- IT Administration
- IT Staff
- Other (please specify)

If you selected other, please specify

---

130) What is the average number of KPIs per dashboard?

enter a number:

---

131) Dashboards are being used to measure the following as part of this project?

	Yes	No	Unknown
Performance of business processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Performance of IT Infrastructure	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Correlation of Business Events and automated actions	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Operation Metrics	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
IT Development Metrics	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architecture Metrics	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

132) Does the dashboard support business and IT alignment?

- Yes
- No
- Unknown

133) Does the solution include archiving capabilities for the dashboards' historical data?

- Yes
- No
- Unknown

134) Which of the following activities were achieved by the use and analysis of dashboard data?

	Yes	No	Unknown
Enhancing business process based on business KPIs information	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Enhancing IT Infrastructure based on IT KPIs information	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Adjusting the strategy of the business	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Enhancing Security measures and processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Generation of new requirements	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Identification of new business opportunities	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

135) Did you archive historical data for mining business trends and potential issues?

- Yes
- No
- Unknown

136) Did this project help in creating content that can be fed to analysis tools to provide executives with information to make better business decisions?

- Yes
- No
- Unknown

## Implementation

Let's gather some information on your implementation, products, platforms, etc. What did your project use?

---

137) Application Servers

- WebSphere Application Server
- CICS
- IMS

138) Business Integration and Optimization

- WebSphere Adapters
- WebSphere DataPower SOA Appliances
- WebSphere Enterprise Service Bus
- WebSphere Message Broker
- WebSphere MQ
- WebSphere Service Registry and Repository
- WebSphere Transformation Extender (WTX)
- WebSphere ILOG JRules
- WebSphere Business Events
- WebSphere Business Modeler
- WebSphere Business Monitor
- WebSphere Integration Developer
- WebSphere Business Services Fabric
- WebSphere Partner Gateway
- WebSphere Process Server
- Lombardi BlueWorks
- FileNet Business Process Manager

139) Business Intelligence and Commerce

- Cognos 8 Business Intelligence
- Cognos 8 Go!
- Cognos Now!
- OmniFind
- WebSphere Commerce
- Other (please specify)

If you selected other, please specify

---

140) Data Management and Warehouse

- DB2 Enterprise Server
- DB2 solidDB
- Informix Dynamic Server
- DB2 Alphablox
- InfoSphere Warehouse
- Other (please specify)

If you selected other, please specify

---

141) Enterprise Content Management

- Content Manager
- FileNet Content Manager
- Lotus Web content Management
- FileNet Records Manager

142) Host Transaction Processing

- CICS Transaction Server
- IMS
- TPF

## 143) Information Integration and Master Data Management

- InfoSphere Data Architect
- InfoSphere Information Server
- InfoSphere Master Data Management Server

## 144) Organizational Productivity, Portals and Collaboration

- Lotus Forms
- WebSphere Portal
- WebSphere Portlet Factory
- Lotus Quickr
- Lotus Sametime
- Lotus Connections

## 145) Security

- Tivoli Access Manager
- Tivoli Federated Identity Manger
- Tivoli Directory Server
- Tivoli Directory Integrator
- Tivoli Identity Manger
- Rational AppScan

## 146) Architecture &amp; Software Development

- Rational Application Developer for WebSphere Software
- Rational Software Architect
- Rational ClearCase
- Rational ClearQuest
- Rational Team Concert
- Rational Asset Manager
- Rational Requirements Composer
- Rational RequisitePro
- Rational System Architect
- Rational Method Composer
- Rational Functional Tester
- Rational Performance Tester
- IBM DOORS

## 147) Storage Management

- Tivoli Storage Manager
- Tivoli Business Continuity Manager
- DITTO

## 148) Systems and Asset Management

- Tivoli Composite Application Manager for Applications
- Tivoli Composite Application Manager for SOA
- Tivoli Monitoring
- Tivoli OMEGAMON XE
- Tivoli Performance Analyzer
- Tivoli Provisioning Manager
- Maximo Asset Management
- Tivoli Enterprise Console
- Tivoli CCMDB

## 149) Operating Systems

- Windows
- Macintosh
- AIX
- Linux
- HP-UX, SunOS, Solaris
- z/OS

## 150) Programming Languages

- Java
- C/C++
- PHP
- Perl
- JavaScript
- C#
- Ruby/Rails
- COBOL
- PL/1
- Other (please specify)

If you selected other, please specify

---

151) Hardware

- pSeries
- xSeries
- zSeries
- Blade
- DataPower
- iSeries
- Lenovo desktop
- Other (please specify)

If you selected other, please specify

---

152) Have you used appliances or pre-built systems?

- WebSphere DataPower SOA Appliances
- WebSphere CloudBurst Appliance
- InfoSphere Balanced Warehouse
- IBM Smart Analytics System
- Other (please specify)

If you selected other, please specify

---

153) Third Party Middleware Components

- BEA/Oracle ESB
- TIBCO
- MS BizTalk
- Sun/Oracle
- Vitria
- Open Source ESB (OpenESB, Mule, etc. )
- JBoss ESB
- Sonic
- Other (please specify)

If you selected other, please specify

---

154) Third Party Applications

- Siebel
- SAP
- Oracle Financials
- Sales Force Automation
- Hogan
- Proprietary
- Other (please specify)

If you selected other, please specify

---

155) Third Party Hardware

- Oracle Sun
- Intel
- HP
- Other (please specify)

If you selected other, please specify

---

156) What other IBM Software did you use in your project?

---

---

---

---

157) What other non-IBM Software did you use in your project.

---

---

---

---

158) What other non-IBM Hardware used in your project?

---

---

---

---

159) How many project releases did you have over a year?

enter a number:

---

160) What is the total number of emergency releases and fixes that you execute during a year?

enter a number:

---

161) What is the percentage of releases that are deployed on schedule?

enter a number:

---

%

162) What is the percentage of releases that are deployed late?

enter a number:

---

%

163) The process for providing access control to services and resources within the organization is well defined and enforced?

- Yes
- No
- Unknown

164) Security compliance issues are usually detected proactively and reported automatically through the right channels?

- Yes
- No
- Unknown

165) How are you automating your install/configure/deployments?

- PlusOne
- Compatibility Advisor
- Solution Assembly Toolkit
- Tivoli Service Provisioning Manager (TSPM)
- SPIN
- Risk Management Provisioning (RMP)
- VMWare
- Hypervisor
- PowerVM
- BuildForge
- Zephyr
- Custom Scripts
- Maven/Open Source
- IBM Installation Manager
- We only do regular product installs
- Unknown
- Other (please specify)

If you selected other, please specify

---

## Operations and Change

Let's gather some information on operations, change management, QoS, SLAs, etc.

---

166) The organization has defined processes to enhance business and IT operations in the following areas?

	Yes	No	Unknown
Measuring QoS	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Requiring SLAs for all major services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Issuing periodical audit reports	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Enhancing resource management and utilization	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Proactively monitoring thresholds and predicting issues before becoming problems	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

167) Describe the legacy systems and how they are integrated with other projects?

	Yes	No	Unknown
Through Service Interfaces	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Through Business Service Interfaces	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
SLAs are attached to services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

168) Does the solution use a discrete process orchestration engine for process automation?

- Yes
- No
- Unknown

169) The process for creating and deploying applications is?

	Yes	No	Unknown
Documented	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Applied	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

170) The process for releasing application changes to the environment is?

	Yes	No	Unknown
Documented	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Applied	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

171) Do you monitor?

	Yes	No	Unknown
Services	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Resources	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

172) What percentage of services and resources have policies associated to address SLA issues?

Services %: \_\_\_\_\_  
 Resources %: \_\_\_\_\_

173) What percentage of services and resources have SLAs and are proactively managed and monitored?

Services %: \_\_\_\_\_  
 Resources %: \_\_\_\_\_

## Product Feedback

Let's gather some information on issues with products and how you addressed them

---

174) Where did you find gaps in IBM products (services, software, hardware)?

- Conformance to standards
- Integration issues
- Complete solution
- Automation
- Missing information
- Defects
- Missing function
- Unsupported capabilities
- Methodology incomplete
- Work products not sufficient
- Immature offering
- No Gaps
- Other (please specify)

If you selected other, please specify

---

175) How did you get around product (software, hardware, services) gaps?

- Created custom coding
- Designed around problem
- Drove changes (PMRs, CRs) to product owners
- Create a new engagement approach
- Escalated
- We were stuck
- Brought in more people
- Lost the opportunity
- Other (please specify)

If you selected other, please specify

---

176) Describe your product feedback below. Include a hyperlink to external documentation if needed.

---

---

---

---

## Lessons Learned & Best Practices

Let's gather some information on the top 5 lessons learned and top 5 best practices. A lesson learned is usually a bad experience that you managed to deal with. A best practice is repeatable way forward that usually addresses a gap.

---

### Lessons Learned

177) Enter your top 5 lessons learned:

1: \_\_\_\_\_

2: \_\_\_\_\_

3: \_\_\_\_\_

4: \_\_\_\_\_

5: \_\_\_\_\_

178) Enter your top 5 best practices:

1: \_\_\_\_\_

2: \_\_\_\_\_

3: \_\_\_\_\_

4: \_\_\_\_\_

5: \_\_\_\_\_

## Assets

Let's gather some information on how you created and reused assets.

---

179) How many assets did you create?

enter a number:

---

180) Why did you create the assets?

- Unique Implementation
- First of a Kind
- Gaps in products
- Required to create assets
- Other (please specify)

If you selected other, please specify

---

181) Describe the assets you created.

---

---

---

---

182) Did you put your assets in iRAM?

- Yes
- No
- Unknown

183) How many assets did you reuse?

enter a number:

---

184) What value or costs did you realize in the assets you reused?

	High	Medium	Low
Asset Reuse led to cost savings	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Customization of Assets Needed	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gave feedback to Asset Owners	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Asset was easy to use	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Assets gave us problems	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Reduced risk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Improved odds of winning	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

185) What asset taxonomy approach do you use to enable ease of discovery and retrieval?

- No catalog
- Naming convention
- Metadata
- Configurable metadata
- Use social computing metaphor (tagging, community, etc)
- UI navigation
- Other (please specify)

If you selected other, please specify

---

186) How did you track reuse of assets?

- No tracking
- Manual report compilation
- Database query
- Reporting analytics system
- Unknown
- Repository does it for us
- Other (please specify)

If you selected other, please specify

---

187) Did you follow one of the IBM reference architectures or equivalent?

- Yes
- No
- Unknown

188) Select the cross Industry frameworks, solutions or assets you used in the project.

	Yes	No	Unknown
Product Development Integration Framework (PDIF)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Banking Industry Framework	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Integrated Information Framework (IIF)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Government Industry Framework (GIF)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Network Centric Operations	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

(NCO)			
Health Integration Framework (HIF)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Insurance Industry Framework	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Media Enterprise Framework (MEF)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Retail Industry Framework	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Service Provider Delivery Environment (SPDE) Framework	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Solution Architecture for Energy & Utilities (SAFE) Framework	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
WebSphere Accelerators	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Integrated Architecture Modeling & Management (IAM2)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Smart Architecture & Engineering (Smart AE)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
WebSphere Commerce	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
IBM Information Framework (IFW)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



192) Using the Open Group v1.8 maturity model, where did your project finish?

	Silo	Integrated	Componentized	Services	Composite Services	Virtualized Services	Dynamically Re-configurable Services
Business View	<input type="radio"/>						
Governance & Organization	<input type="radio"/>						
Methods	<input type="radio"/>						
Applications	<input type="radio"/>						
Architecture	<input type="radio"/>						
Information	<input type="radio"/>						
Infrastructure & Management	<input type="radio"/>						

193) What were your SOA Entry Points?

- People: enables efficiency through interaction and collaboration
- Process: offers tools and services to help streamline business process management
- Information: enables access to complex, heterogeneous data sources
- Connectivity: links people, processes and information for your business
- Reuse: extends the value of your previous asset investments

194) Which of the following impact analysis benefits did you achieve on your project? Given a change in a requirement, business processes, service, strategy, business rule, asset, team, etc., what type of Impact Analysis did you automatically provide?

- Static - determines and reports on what other requirements, processes, etc. are related to the change.
- Predictive - determines and reports on estimated cost to support the change
- Historical - determines and reports on the cost it already took to implement what may change
- LifeCycle - determines and reports on end to end (business architecture to implementation) what else will change, what it may cost to change, and what it already cost to deploy
- Other (please specify)

If you selected other, please specify

---

195) Do you formally measure the impact of the any of the following?

	Yes	No	Unknown
Information model changes on services and process models	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Architecture changes on project timeline	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Service model changes impact on processes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Process changes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

196) Business agility is usually achieved through the continuous and proactive sensing of business events and having the right policies and rules to address changes in an automated fashion. Did this project bring your organization close to business agility through the introduction of new business rules and policies that may address potential ad-hoc business situations?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all

197) The use of dynamic business rules that can be applied on the fly without having to modify the business layer is considered an element that injects flexibility into solutions and therefore considered a promoter of business agility. Did this project bring your organization close to business agility through the introduction of dynamic business rules that can be applied on the fly without having to modify the business layer?

- Great extent, i.e. more than 80%
- Moderate extent, i.e. about 50%
- Very little extent, i.e. less than 20%
- Not at all

198) As a result of building and deploying properly-governed business services, the organization is able to create new business solutions that address market requests and competitors faster and more efficiently?

- Yes
- No
- Unknown

199) As a result of building and deploying properly-governed business services, the organization is able to get a better view of their business conditions and react faster to events that have the potential to cause disruption of their business.

- Yes
- No
- Unknown

200) Did business agility requirements for your project lead to a change of data base engine / technology?

- Yes
- No
- Unknown

201) Incorporating business rules and policies as part of this project helped your organization to adapt better to dynamic business situations as soon as they are identified?

- Yes
- No
- Unknown

202) As a result of building and deploying properly-governed business services, the organization is able to view consistent and trusted data?

- Yes
- No
- Unknown

203) As a result of building and deploying properly-governed business services, the organization is able to understand their customer base better and offer them more relevant services that are likely to meet their acceptance?

- Yes
- No
- Unknown

204) As a result of building services for this project, the organization is able to create services faster to the marketplace and therefore enhance their revenue stream?

- Yes
- No
- Unknown

205) This project and the identified business services used in this project will make your organization more business agile?

- Yes
- No
- Unknown

206) Did the following help your client's business agility?

	Yes	No
Business Dashboards & Analytics	<input type="radio"/>	<input type="radio"/>
Cross Organization Teaming	<input type="radio"/>	<input type="radio"/>
Dynamic Business Processes	<input type="radio"/>	<input type="radio"/>
Real Governance	<input type="radio"/>	<input type="radio"/>
Excellent Industry Knowledge	<input type="radio"/>	<input type="radio"/>
Skilled Staff	<input type="radio"/>	<input type="radio"/>
Actionable Architecture	<input type="radio"/>	<input type="radio"/>
SOA and Implementation	<input type="radio"/>	<input type="radio"/>
Virtualized Platforms	<input type="radio"/>	<input type="radio"/>
Impact Analysis	<input type="radio"/>	<input type="radio"/>
Open Standards	<input type="radio"/>	<input type="radio"/>
Flexible Runtime	<input type="radio"/>	<input type="radio"/>

207) What are the top 4 contributors from the project that helped your client increase business agility?

- Cross Organization Teaming \_\_\_\_\_
- Virtualized Platform \_\_\_\_\_
- Flexible Runtime \_\_\_\_\_
- Actionable Architecture \_\_\_\_\_
- Impact Analysis \_\_\_\_\_
- Open Standards \_\_\_\_\_

- Business Dashboards and Analytics \_\_\_\_\_
- SOA and Implementation \_\_\_\_\_
- Skilled Staff \_\_\_\_\_
- Real Governance \_\_\_\_\_
- Dynamic Business Processes \_\_\_\_\_
- Excellent Industry Knowledge \_\_\_\_\_

208) How does your client achieve repeatable and predictable outcomes in the face of changes in customers, market, IT, and innovation?

- By addressing change via change processes
- Via mergers and acquisitions
- Via outsourcing
- Flexible business and agile architecture
- Flexible and agile runtime
- Monitoring and managing KPIs and taking corrective actions
- Other (please specify)

If you selected other, please specify

\_\_\_\_\_

209) Did your project have specific requirements to gather metrics (KAIs) to measure increased business agility?

- Yes
- No
- Unknown

210) Give examples of the requirements for business agility.

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

211) What are the key measurements of business agility that you looked for in your project?

\_\_\_\_\_  
\_\_\_\_\_

---

212) Did you have to address those measurements directly in your project?

- Yes
- No
- Unknown

213) How did you ensure you were meeting those key business agility requirements?

---

---

---

214) How is business agility measured in the business?

---

---

---

---

**Finished!**

Congratulations.

---

215) You made it! How long did it take you to complete this survey?

- 1 hour
- 2 hours
- 3 hours
- 4 hours
- Other (please specify)

If you selected other, please specify

---

## APPENDIX C SCORING METHOD PER SURVEY QUESTION TYPES

Table 27 shows the details for the scoring method of survey question types where translation to numerical values is required.

Question Type	Scoring Method
Simple Yes/No questions	Yes = one point, No = zero points
Percentage value questions – Type 1	<ul style="list-style-type: none"> <li>○ More than 80% = 5 points</li> <li>○ 60% - 80% = 4 points</li> <li>○ 40% - 60% = 3 points</li> <li>○ 20% - 40% = 2 points</li> <li>○ Below 20 % = 1 point</li> </ul> Unknown = -1 to indicate missing
Percentage value questions – Type 2 (High, Medium, Low)	<ul style="list-style-type: none"> <li>○ Great extent, i.e. more than 80% = 4 points</li> <li>○ Moderate extent, i.e. about 50% = 3 points</li> <li>○ Very little extent, i.e. less than 20% = 2 points</li> <li>○ Not at all = 0 points</li> </ul>

**Table 27 Survey Questions Scoring**

# APPENDIX D SURVEY PARTICIPANTS ORIENTATION SESSION

IBM Academy

 IBM Academy of Technology



**3<sup>rd</sup> SOA Case Study Best Practices Conference,**  
with a focus on business agility, cloud computing, smarter planet and BPM.  
Sept 7-Sept 9

## **Orientation Session**

May 10, 2010



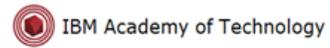
© 2010 IBM Corporation



---

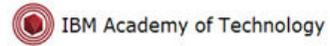
## Agenda

- Conference Summary
- Conference Strategy
- Business Agility
- Proposed Business Agility Definition
- SOA Project Profile
- Survey Participants Activities
- Important Conference Dates
- Uncompleted Surveys and Multiple Surveys



## Conference Summary

- Build on success of 2007 and 2008 Academy SOA Conference
- Bring together key SOA deployment experts to share client case studies
- Leverage standardized case study presentations
- Introduce Survey as a new approach to harvesting project information
- Harvest case study information through human and automated processing
- Share case study assets throughout IBM
- Identify findings and make recommendations through detailed analysis of survey and case study data

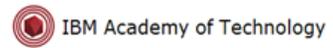


## Conference Strategy

- **Virtual conference** where practitioners share and present client case studies
  
- Focus on client approaches, lessons learned and best practices (SOA, BPM, Business Agility, Cloud and Smart Planet)
  
- Presentations follow standardized case study template
  - Accompanied by detailed survey the aligns to case study template
  - Measurement and harvesting of results when all submissions are in
  
- Results of conference
  - Drive product changes
  - Drive services changes
  - Share with clients
  - Our goal is not to prove a method or technology. Our goal is to harvest best practices and data from engagements, then apply it to support our offerings, or drive change.

## Business Agility

- The ability to “manage and apply knowledge effectively, so that an organization has the potential to thrive in a continuously changing and unpredictable business environment”. **[Dove]**
  
- “The ability of an organization to sense environmental change and respond efficiently and effectively to that change.” **[Gartner]**
  
- “The ability of enterprise elements to adapt proactively to unexpected and unpredicted changes”. **[Kidd]**
  
- **Common**
  - **The ability of an organization to adapt in response to changes.**



---

## Proposed Business Agility Definition

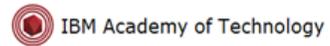
Organization's ability to effectively sense, manage, adapt and respond efficiently and effectively to produce a desired outcome to opportunities, challenges and competition, through leveraging business strategy, people and information technology.



---

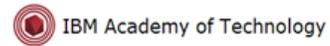
## SOA Project Profile

- Project is production-bound with impact on the business
- Project used service oriented architecture (SOA) as the primary architectural style to reach the solution architecture
- Project measurements were as accurate as possible and did not include "questimates"
- Web services (JEE or .Net) are the primary building blocks for realizing business capabilities. Projects that used service-oriented methods to discover services will be allowed to participate, e.g. CORBA.
- Project sizes and complexity of all aspects are welcome. At a minimum, projects that contained less than 5 services will not be accepted to ensure some level of project complexity is present. There are no restrictions on project duration.
- Projects that use web services that do not map properly to business services will be considered for this study.
- No assumptions regarding the success of projects or business agility results. Failed projects are also welcome.



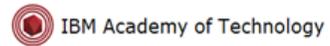
## Survey Participants Activities

- IBM experts will be assigned to help orient survey participants regarding the overall process and go over some of the important terms
- IBM experts will share the survey questions with survey participants to help them prep for the actual survey process
- Present the concept of Business Agility and its definition to survey participants along with explanations to any questions on the survey
- The survey requires an extensive knowledge of project details and therefore participants are encouraged to do the following:
  - Read the survey questions and prepare for the answers well in advance
  - Provide accurate information regarding responses to survey questions
  - Meet with project personnel to discuss business and/or IT questions that they may not know from their involvement with the account in order to answer survey questions



## Important Conference Dates

- **May 3, 2010**                      **Call for participation and abstracts**
- May 14, 2010                      Reminder for call for participation and abstracts
- **June 25, 2010**                      **First notification of early submitters to start case study and survey**
- **July 9, 2010**                      **Deadline for abstracts**
  
- July 13, 2010                      Final notification of participants of acceptance
- **July 15, 2010**                      **Second notification to start case study and survey**
- July 30, 2010                      Final notice to submit case study and survey
- **August 16, 2010**                      **Deadline for case study and survey**
  
- **September 7, 2010**                      **Conference starts**



## Common Survey Questions: Uncompleted Surveys and Multiple Surveys

- If you exit browser or your computer crashes before submitting survey, next time you attempt to go to the survey you will get this screen:

### Incomplete surveys

You have uncompleted surveys that you may take some action on.

**Resume:** Resume an existing survey submission at the last unanswered question.

**Resume from First Page:** Resume an existing survey submission at the first page with the ability to review existing answers.

**Delete:** Delete your existing survey submission.

2010

<http://ucdsurvey1.torolab.ibm.com/ss2/wsb.dll/113/10200.htm> [Resume](#) | [Resume From First Page](#) | [Delete](#)

[Terms of use](#)

- If you want to start from scratch again, then do Delete and reenter the survey URL
- If you want to resume where you left off, click Resume – Answers are stored
- You CAN enter multiple surveys, one for each project assuming you submitted more than one abstract
- When you submit a survey and go to survey site again, it will assume a new survey
- Only one survey per abstract submission is allowed

## Bibliography

- [1] N. Bieberstein, *Executing SOA: a practical guide for the service-oriented architect*. Upper Saddle River, NJ: IBM Press/Pearson Plc, 2008.
- [2] A. Arsanjani, A. Ghosh, A. Allam, T. Abdollah, S. Ganapathy, and K. Holley, "SOMA: A method for developing service-oriented solutions," vol. 47, no. 3, pp. 377–396, 2008.
- [3] R. High, S. Kinder, and S. Graham, "IBM SOA Foundation: An architectural introduction and overview." [Online]. Available: <http://www.ibm.com/developerworks/webservices/library/ws-soa-whitepaper/>. [Accessed: 21-Nov-2011].
- [4] N. Fenton, *Software metrics: a rigorous approach*. London, Boston: International Thomson Computer Press, 1995.
- [5] E. Newcomer, *Understanding SOA with Web Services*. Upper Saddle River, NJ: Addison-Wesley, 2005.
- [6] W. Brown, *SOA Governance: Achieving and sustaining business and IT agility*. Upper Saddle River, NJ: IBM Press/Pearson Plc, 2009.
- [7] M. Fiammante, *Dynamic SOA and BPM: Best Practices for Business Process Management and SOA Agility*. Upper Saddle River, NJ: IBM Press/Pearson Plc, 2010.
- [8] M. Ren and K. Lyytinen, "Building Enterprise Architecture Agility and Sustenance with SOA," *Communications of the Association for Information Systems*, vol. 22, no. 4, pp. 75–86, 2008.
- [9] R. Dove, *Response ability: the language, structure, and culture of the agile enterprise*. New York: J. Wiley, 2001.
- [10] D. McCoy and D. Plummer, "Defining, Cultivating and Measuring Enterprise Agility." [Online]. Available: <http://www.gartner.com/DisplayDocument?id=491436>. [Accessed: 21-Nov-2011].
- [11] Y. Yusuf, M. Sarhadi, and A. Gunasekaran, "Agile Manufacturing: The drivers, concepts and attributes," *International Journal of Production Economics*, vol. 62, no. 1–2, pp. 33–43, 1999.
- [12] S. L. Goldman and R. N. Nagel, "Management, technology and agility: the emergence of a new era in manufacturing," *International Journal of Technology Management*, vol. 8, no. 1–2, pp. 18–38, 1993.
- [13] P. Kidd, *Agile manufacturing : forging new frontiers*. Wokingham, England - Reading, Mass: Addison-Wesley, 1994.
- [14] B. Sherehiy and W. Karwowski, "A review of enterprise agility: Concepts, frameworks, and attributes," *International Journal of Industrial Ergonomics*, vol. 37, no. 5, pp. 445–60, 2007.
- [15] J. L. Zhao, M. Tanniru, and L. J. Zhang, "Services computing as the foundation of enterprise agility: Overview of recent advances and introduction to the special issue," *Information Systems Frontiers*, vol. 9, no. 1.

- [16] N. C. Tsourveloudis and K. P. Valavains, "On the measurement of enterprise agility," *Journal of Intelligent and Robotic Systems*, vol. 33, no. 3, pp. 329–342, 2002.
- [17] P. Kruchten, *The Rational Unified Process: An Introduction*, 3rd ed. Boston: Addison-Wesley, 2004.
- [18] F. Cummins, *Building the Agile Enterprise: with SOA, BPM and MBM*. Boston, Amsterdam: MK/OMG Press/Elsevier, 2009.
- [19] M. Hirzalla, P. Bahrs, J. Cleland-Huang, C. S. Miller, and R. High, "A Predictive Business Agility Model for Service Oriented Architectures," presented at the Int'l Conference on Service Oriented Computing (ICSOC), Paphos, Cyprus, 2011, pp. 653–660.
- [20] M. Hirzalla, P. Bahrs, J. Cleland-Huang, C. S. Miller, and R. High, "Analysis of Business Agility Indicators in SOA Deployments," in *The 2011 International Conference on Software Engineering Research and Practice*, 2011, pp. 400–407.
- [21] M. Hirzalla, A. Zisman, and J. Cleland-Huang, "Using Traceability to Support SOA Impact Analysis," presented at the Services, Washington, D.C., USA, 2011, pp. 145–152.
- [22] M. Hirzalla, J. Cleland-Huang, and A. Arsanjani, "A Metrics Suite for Evaluating Flexibility and Complexity in Service Oriented Architectures," presented at the ICSOC Workshop on Engineering Service Oriented Applications, Sydney, Australia, 2008, pp. 41–52.
- [23] D. K. Barry, *Web services and service-oriented architectures: the savvy manager's guide*. San Francisco, CA: Morgan Kaufmann; Elsevier Science, 2003.
- [24] T. Erl, *Service-oriented architecture: concepts, technology and design*. Upper Saddle River, NJ: Prentice Hall Professional Technical Reference, 2005.
- [25] D. Krafzig, K. Banke, and D. Slama, *Enterprise SOA: Service-Oriented Architecture Best Practices*, Fifth. Prentice Hall, 2006.
- [26] B. Roth, "Service-Oriented Architecture best practices," *Web Services Journal*, vol. 5, pp. 14–15, 2005.
- [27] T. Erl, *SOA: Principles of service design*, 1st ed. Upper Saddle River, NJ: Prentice Hall, 2008.
- [28] M. N. Huhns and M. P. Singh, "Service-Oriented Computing: key concepts and principles," *IEEE Internet Computing*, vol. 9, no. 1, pp. 75–81, 2005.
- [29] M. Papazoglo and A. D. Georgakopoulos, "Service-Oriented Computing," *Communications of the ACM*, vol. 46, no. 10, pp. 24–28, 2003.
- [30] T. Kohlborn, A. Korthaus, T. Chan, and M. Roseman, "Identification and Analysis of Business and Software Services—A Consolidated Approach," *IEEE Transactions of Services Computing*, vol. 2, no. 1, 2009.
- [31] M. Bell, *Service-oriented modeling: service analysis, design and architecture*. Hoboken, NJ: John Wiley & Sons, 2008.
- [32] T. Erl, *Service-Oriented Architecture: A Field Guide to Integrating XML and Web Services*. Upper Saddle River, NJ: Prentice Hall, 2004.
- [33] P. Bahrs and T. Nguyen, "Smarter Architecture & Engineering: Game Changer for Requirements Management," presented at the RESS11, 2011, pp. 1–5.

- [34] M. Keen, G. Ackerman, I. Azaz, M. Haas, R. Johnson, J. Kim, and P. Robertson, *Patterns: SOA Foundation - Business Process Management Scenario*. IBM Redbooks, 2006.
- [35] A. Ganek and K. Kloeckner, "An Overview of IBM Service Management," *IBM Systems Journal*, vol. 46, no. 3, pp. 375–385, 2007.
- [36] M. Chuba and D. Scott, "Top Challenges Facing IT I&O Executives," G00160214, Sep. 2008.
- [37] M. K. Haki and M. Wentland, "Proposal of a Service Oriented Architecture Governance Model to serve as a practical framework for Business-IT Alignment," presented at the 4th IEEE International Conference on New Trends in Information Sciences and Service Sciences, Gyeongju, 2010.
- [38] S. Izza and R. Imache, "An Approach to Achieve IT Agility by Combining SOA with ITSM," *International Journal Information Technology and Management*, vol. 9, no. 4, pp. 423–445, 2010.
- [39] I. Jacobson, *The Unified Software Development Process*. Reading, Mass: Addison-Wesley.
- [40] M. Papazoglo and W. Van Den Huevel, "Service-oriented design and development methodology," *International Journal of Web Engineering and Technology*, vol. 2, no. 4, pp. 412–442, Jul. 2006.
- [41] S. Jones and M. Morris, "A Methodology for Service Architectures." Contribution to OASIS, Oct-2005.
- [42] "OASIS SOA Reference Model." [Online]. Available: <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf>. [Accessed: 23-Nov-2011].
- [43] "OASIS UDDI Specification TC | OASIS." [Online]. Available: [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=uddi-spec](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=uddi-spec). [Accessed: 28-Apr-2012].
- [44] M. Keen, J. Bond, J. Denman, S. Foster, S. Husek, B. Thompson, and H. Wylie, *Integrating ESB in SOA*. IBM Redbooks, 2005.
- [45] "SOAP Specifications." [Online]. Available: <http://www.w3.org/TR/soap/>. [Accessed: 07-Jan-2012].
- [46] B. Hutchinson, M.-T. Schmidt, D. Wolfson, and M. Stockton, "SOA programming model for implementing Web services, Part 4: An introduction to the IBM Enterprise Service Bus," Jul-2005. [Online]. Available: <http://www.ibm.com/developerworks/webservices/library/ws-soa-progmodel4/index.html>.
- [47] C. Dudley, L. Rieu, M. Smithson, T. Verma, and B. Baswell, *WebSphere Registry and Repository Handbook*. IBM Redbooks, 2007.
- [48] "ABPMP - The Association of Business Process Management Professionals." [Online]. Available: <http://www.abpmp.org/>. [Accessed: 23-Nov-2011].
- [49] L. M. Sanchez and R. Nagi, "A review of agile manufacturing systems," *International Journal of Production Research*, vol. 39, no. 16, pp. 3561–3600, 2001.
- [50] C. W. Richards, "Agile Manufacturing: Beyond lean?," *Production and Inventory Management Journal*, vol. 37, no. 2, pp. 60–64, 1996.

- [51] D. Gerwin, "An Agenda for Research on the Flexibility of Manufacturing Processes," *International Journal of Operations & Production Management*, vol. 7, no. 1, pp. 38–49, 1987.
- [52] E. S. Bernardes and M. D. Hanna, "A Theoretical Review of Flexibility, Agility and Responsiveness in the Operations Management Literature," *International Journal of Operations & Production Management*, vol. 29, no. 1, pp. 30–53, 2009.
- [53] D. M. Upton, "Flexibility as process mobility: The management of plant capabilities for quick response manufacturing," *Journal of Operations Management*, vol. 12, no. 34, pp. 205–224, 1995.
- [54] M. Stevenson and M. Spring, "Flexibility from a supply chain perspective: definition and review," *International Journal of Operations & Production Management*, vol. 27, no. 7, pp. 685–713, 2007.
- [55] Y. Gong and M. Janssen, "Measuring Process Flexibility and Agility," in *Proceedings of the 4th International Conference on Theory and Practice of Electronic Governance*, 2010.
- [56] A. Martensson, "Producing and Consuming Agility," *Agile Information Systems: Conceptualization, Construction, and Management*, pp. 41–51, 2007.
- [57] C. Ling and L. Xin, "Achieving Business Agility by Integrating SOA and BPM Technology," in *International Forum on Information Technology and Applications*, Chengdu, China, 2009, pp. 334–337.
- [58] J. Choi, D. Nazareth, and H. Jain, "The Impact of SOA Implementation Strategies on Business Value and Agility: A Systems Dynamics Approach," presented at the 6th International Conference on Advanced Information Management and Service (IMS), Seoul, Korea, 2010.
- [59] P. Iocola, "When Legacy Meets SOA: Achieving Business Agility by Integrating New Technology with Existing Software Asset," presented at the 1st Annual IEEE Systems Conference, Honolulu, HI, USA, 2007.
- [60] R. Imache, S. Izza, and A.-N. Mohamed, "An enterprise information system agility assessment model," *Computer Science and Information Systems*, vol. 9, no. 1, pp. 107–133, 2012.
- [61] S. Izza, R. Imache, L. Vincent, and Y. Lounis, "An Approach for the Evaluation of the Agility in the Context of Enterprise Interoperability," in *In Proceedings of IESA*, 2008, pp. 3–14.
- [62] A. Rostampour, A. Kazemi, H. Haghghi, and F. Shams, "A Metric Suite for Measuring Service Alignment with Business Agility," presented at the Computer Science and Software Engineering, CSI Int'l Symposium on, 2011, pp. 121–128.
- [63] Business Rules Group, "Defining Business Rules: Why Are They Really?" [Online]. Available: [http://www.businessrulesgroup.org/first\\_paper/br01c1.htm](http://www.businessrulesgroup.org/first_paper/br01c1.htm). [Accessed: 26-Nov-2011].
- [64] "Event Processing Technical Society." [Online]. Available: <http://www.ep-ts.com/>. [Accessed: 28-Apr-2012].
- [65] W. R. Schulte, "The Growing Role of Events in Enterprise Applications," Gartner, AV-20-3900, Jul. 2003.

- [66] J. Ganci, M. Flugrath, A. Manekar, and M. Trasti, *Best Practices for SOA Management*. 2007.
- [67] B. Darmawan, P. Nambiar, P. Lall, and R. Gummadavelli, *Managing an SOA Environment with Tivoli*. 2008.
- [68] L. Vaquero, J. Caceres, and J. Hierro, *Open Source Cloud Computing Systems*. IGI Global, 2012.
- [69] T. H. Davenport, "Competing on Analytics," *Harvard Business Review*, 2007.
- [70] "OASIS Web Services Business Process Execution Language (WS-BPEL) 2.0." [Online]. Available: <http://docs.oasis-open.org/wsbpel/2.0/wsbpel-specification-draft.html>. [Accessed: 24-Nov-2011].
- [71] D. Parmenter, *Developing, Implementing and Using Winning KPIs*, 2nd ed. New Jersey: John Wiley & Sons, 2010.
- [72] "Policy - Definition and More from the Free Merriam-Webster Dictionary." [Online]. Available: <http://www.merriam-webster.com/dictionary/policy>. [Accessed: 28-Apr-2012].
- [73] S. A. Bohner and R. S. Arnold, *Software Change Impact Analysis*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1996.
- [74] S. L. Pfleeger and J. M. Atlee, *Software Engineering: Theory and Practice*, 2006th ed. Upper Saddle River, NJ: Prentice Hall.
- [75] "IBM Change and Configuration Management Database, Version 7.2.1." [Online]. Available: [http://publib.boulder.ibm.com/infocenter/tivihelp/v32r1/index.jsp?topic=%2Fcom.ibm.ccmdb.doc\\_721%2Fchange%2Ft\\_plan\\_AIA.html](http://publib.boulder.ibm.com/infocenter/tivihelp/v32r1/index.jsp?topic=%2Fcom.ibm.ccmdb.doc_721%2Fchange%2Ft_plan_AIA.html). [Accessed: 02-Jan-2012].
- [76] V. Gucer, W. T. Barbosa, M. Ferkoun, K. Kidambhi, M. Lambert, R. Mincov, R. Noppert, and U. Pradeep, *Integration Guide for IBM Tivoli Service Request Manager V7.1*. IBM Redbooks, 2008.
- [77] D. Kaye, *Loosley Coupled: The Missing Pieces of Web Services*, 1st ed. Marin County, California: RDS Press, 2003.
- [78] R. C. Thomas II, "A Practical Guide to Federal Enterprise Architecture." US Customs Service, Feb-2001.
- [79] R. Pressman, *Software Engineering: A Practitioner's Approach*, 7th ed. New York, NY, USA: McGraw-Hill Higher Education, 2010.
- [80] S. L. Lim, E. Ishikawa, and P. and K. Cox, "Towards Agile Service-oriented Business Systems: A Directive-oriented Pattern Analysis Approach," presented at the IEEE International Conference on Services Computing, Honolulu, HI, USA, 2008, pp. 231–238.
- [81] C. T. Lin, H. Chiu, and Y. H. Tseng, "Agility Evaluation Using Fuzzy Logic," vol. 101, no. 2, pp. 353–368, 2006.
- [82] C. T. Lin, H. Chiu, and P. Y. Chu, "Agility Index in the Supply Chain," *International Journal of Production Economics*, vol. 100, no. 2, pp. 285–299, 2006.
- [83] J. Ren, Y. Yusuf, and D. Burns, "A prototype of measurement system for agile enterprise," presented at the International Conference on Quality, Reliability and Maintenance, Oxford, UK, 2000, pp. 247–252.

- [84] B. M. Arteta and R. E. Giachetti, "A measure of agility as the complexity of the enterprise system," *Robotics and Computer-Integrated Manufacturing*, vol. 20, no. 6, pp. 495–503, 2004.
- [85] J. Buzacott, "The Structure of Manufacturing Systems: Insights on the Impact of Variability," *International Journal of Flexible Manufacturing Systems*, vol. 11, no. 1, pp. 27–46, 1999.
- [86] W. E. Diewert, *The Economic Theory of Index Numbers: A Survey.*, In Essays in the Theory and Measurement of Consumer Behavior in Honour of Sir Richard Stone, A. Deaton. London, Cambridge University: , 1981.
- [87] C. Kaner and W. Bond, "Software Engineering Metrics: What Do They Measure and How Do We Know?," presented at the 10th International Software Metrics Symposium, Metrics 2004, Chicago, IL - USA.
- [88] P. Goodman, *Practical Implementation of Software Metrics*. New York, NY, USA: McGraw-Hill, 1993.
- [89] L. C. Briand, S. Morasca, and V. R. Basili, "An operational process for goal-driven definition of measures," *IEEE Transactions on Software Engineering*, vol. 28, no. 12, pp. 1106–1125, 2002.
- [90] K. El Emam, "A Methodology for Validating Software Product Metrics," Canada, NRC 44142, 2000.
- [91] N. Fenton, "Software Measurement: A Necessary Scientific Basis," *IEEE Transactions on Software Engineering*, vol. 20, no. 3, pp. 199–206, 1994.
- [92] "IEEE SA - 1061-1998 - IEEE Standard for a Software Quality Metrics Methodology." [Online]. Available: <http://standards.ieee.org/findstds/standard/1061-1998.html>. [Accessed: 23-Nov-2011].
- [93] B. Kitchenham, S. L. Pfleeger, and N. Fenton, "Towards a Framework for Software Measurement Validation," *IEEE Transactions on Software Engineering*, vol. 21, no. 12, pp. 929–944, 1995.
- [94] N. Fenton, M. Neil, and N. Square, "A Critique of Software Defect Prediction Models," *Machine Learning Applications in Software Engineering*, vol. 25, no. 5, p. 72, 2005.
- [95] M. Halstead, *Elements of Software Science*. New York: , 1977.
- [96] T. J. McCabe, "A Complexity Measure," *IEEE Transactions on Software Engineering*, vol. 2, no. 4, pp. 308–320, 1976.
- [97] T. M. Kholshogoftaar, "Predicting Software Development Errors Using Complexity Metrics," *IEEE Journal Selected Areas in Communications*, vol. 8, no. 2, pp. 253–261, 1990.
- [98] S. S. Stevens, "On the Theory of Scales of Measurement," *Science*, vol. 103, no. 2684, pp. 677–680, 1946.
- [99] N. Fenton, "Software Metrics: Theory, Tools and Validation," *Software Engineering Journal*, vol. 5, no. 1, pp. 65–78, 1990.
- [100] L. C. Briand, K. El Emam, and S. Morasca, "On the application of measurement theory in software engineering," *Empirical Software Engineering*, vol. 1, no. 1, pp. 61–88, 1996.

- [101] L. C. Briand, K. El Emam, and S. Morasca, "Theoretical and Empirical Validation of Software Product Measures," ISERN-95-03, 1995.
- [102] H. Zuse, *Software Complexity: Measures and Methods*, vol. 4. Berlin, New York: , 1991.
- [103] S. L. Pfleeger, R. Jeffrey, B. Curtis, and B. Kitchenham, "Status Report on Software Measurement," *IEEE Software*, vol. 14, no. 2, pp. 410–422, 1992.
- [104] N. F. Schneiderwind, "Methodology for Validating Software Metrics," *IEEE Transactions on Software Engineering*, vol. 18, no. 5, pp. 410–422, 1992.
- [105] E. J. Weyuker, "Evaluating Software Complexity Measures," *IEEE Transactions on Software Engineering*, vol. 14, no. 9, pp. 1357–1365, 1988.
- [106] S. R. Chidamber and C. F. Kemerer, "A Metrics Suite for Object Oriented Design," *IEEE Transactions on Software Engineering*, vol. 20, no. 6, pp. 476–493, 1994.
- [107] H. F. Li and W. K. Cheung, "An Empirical Study of Software Metrics," *IEEE Transactions on Software Engineering*, vol. 13, no. 6, pp. 697–708, 1987.
- [108] F. Akiyama, "An Example of Software System Debugging," *Information Processing*, vol. 71, pp. 353–379, 1971.
- [109] T. Compton and C. Withrow, "Prediction and Control of Ada Software Defects," *Journal of Systems and Software*, vol. 12, pp. 199–207, 1990.
- [110] L. Hatton, "The Automation of Software Process and Product Quality," *Software Quality Management*, pp. 727–744, 1993.
- [111] B. Kitchenham, L. M. Pickard, and S. J. Linkman, "An Evaluation of Some Design Metrics," *Software Engineering Journal*, vol. 5, no. 1, pp. 50–58, 1990.
- [112] J. Cohen, *Applied Multiple Regression Correlation Analysis for the Behavioral Sciences*, 3rd ed. Mahwah, NJ: Erlbaum Associates, 2003.
- [113] R. Courtney and D. Gustafson, "Shotgun Correlations in Software Measures," *Software Engineering Journal*, vol. 8, no. 1, pp. 5–13, 1992.
- [114] J. R. Gaffney, "Estimating the Number of Faults in Code," *IEEE Transactions on Software Engineering*, vol. 10, no. 4, 1984.
- [115] L. C. Briand, E. Arisholm, S. Counsell, F. Houdek, and P. Thevenod-Fosse, "Empirical Studies of Object-oriented Artifacts, Methods, and Processes: State of the Art and Future Directions," *Empirical Software Engineering*, vol. 4, no. 4, pp. 387–404, 1999.
- [116] S. Benlarbi and W. L. Melo, "Polymorphism Measures for Early Risk Prediction," in *Proceedings of the 21st Int'l Conference on Software Engineering*, Los Angeles, CA, USA, 1999, pp. 334–344.
- [117] F. B. Abreu and W. Melo, "Evaluating the Impact of Object-oriented Design on Software Quality," in *Proceeding METRICS*, 1996, vol. 96.
- [118] V. R. Basili and G. Caldiera, "The goal question metric approach," *Encyclopedia of Software Engineering*, vol. 1, pp. 528–532, 1994.
- [119] K. K. Hefner, "An Experience-based Optimization of the Goal/Question/Paradigm," in *Proceedings of the California Software Symposium*, Irvine, CA, 1995.

- [120] L. C. Briand, S. Morasca, and V. R. Basili, "Property based software engineering," *IEEE Transactions on Software Engineering*, vol. 22, no. 1, pp. 68–86, Jan. 1996.
- [121] K. El Emam, "A Primer on Objected-oriented Measurement," in *Proceedings of the 7th IEEE Int'l Software Metrics Symp (METRICS '01)*, 2001, pp. 185–188.
- [122] W. Li and S. Henry, "Object-oriented Metrics that Perdict Maintainability," *Journal of Systems and Software*, vol. 23, pp. 111–122, 1993.
- [123] V. R. Basili, L. C. Briand, and W. Melo, "A Validation of Object-Oriented Design Metrics as Quality Indicators," *IEEE Transactions on Software Engineering*, vol. 22, no. 10, pp. 751–761, 1996.
- [124] M. Hitz and B. Montazeri, "Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective," *IEEE Transactions on Software Engineering*, vol. 22, no. 4, pp. 267–271, 1996.
- [125] S. L. Pfleeger, "Lessons Learned in Building a Corporate Metrics Program," *IEEE Software*, vol. 10, no. 3, pp. 67–74, 1993.
- [126] M. Umarji and H. Emurian, "Acceptance Issues in Metrics Program," in *Proceedings of the 11th IEEE International Software Metrics Symposium (METRICS'05)*, 2005, vol. 00.
- [127] M. Umarji and C. Seaman, "Gauging Accpetance of Software Metrics: Comparing Perspectives of Managers and Developers," in *Proceedings of the 3rd Int'l Symposium on Empirical Software Engineering and Measurement*, Orlando, FL - USA, 2009.
- [128] F. Davis, "Perceived Usefulness, Perceived Ease of Use, and End User Acceptance of Information Technology," *MIS Quarterly*, vol. 13, pp. 318–339, 1989.
- [129] D. J. Paulish and A. D. Carleton, "Case Studies of Software Process Improvement Measurement," *Computer*, vol. 27, no. 9, pp. 50–57, 1994.
- [130] J. Iversen and L. Mathiassen, "Lessons from Implementing a Software Metrics Program," in *Proceedings of the Hawaii Int'l Conference on System Sciences (HICSS)*, Maui, HI, 2000, vol. 18, p. 20.
- [131] S. Vinoski, "Old Measures for New Services," *IEEE Internet Computing*, vol. 9, no. 6, pp. 267–271, 2005.
- [132] M. Pereplechikov, C. Ryan, and K. Frampton, "Cohesion Metrics for Predicting Maintainability of Service-oriented Software," in *Seventh Int'l Conference on Quality Assurance (QSIC 2007)*, 2007.
- [133] M. Pereplechikov, C. Ryan, and K. Frampton, "Coupling Metrics for Predicting Maintainability in Service-oriented Designs," in *Proceedings of the 2007 Australian Software Engineering Conference*, 2007.
- [134] Q. Ma, N. Zhou, and H. Wang, "Evaluating Service Identification with Design Metrics on Business Process Decomposition," presented at the IEEE Conference on Services Computing, Bangalore, India, 2009, pp. 160–167.
- [135] D. Rud, A. Schmietendorf, and R. Dumke, "Product Metrics for Service-oriented Infrastructures," presented at the IWSM/MetriKon, 2006, pp. 161–174.

- [136] K. Qian, J. Liu, and F. Tsui, "Decoupling Metrics for Services Composition," presented at the 5th IEEE/ACIS International Workshop on Component Based Software Engineering, Software Architecture and Reuse (ICIS-COMSAR 06), Honolulu, HI, USA, 2006, pp. 44–47.
- [137] Y. M. Liu and I. Traore, "Complexity Measures for Secure Service-oriented Software Architectures," in *Proceedings of the Third Int'l Workshop on Predictor Models in Software Engineering*, 2007.
- [138] J. Salasin and A. Madani, "Metrics for Service-oriented Architecture (SOA) Systems: What Developers Should Know," *Journal of Integrated Design & Process Science*, vol. 11, no. 2, pp. 55–71, 2007.
- [139] A. Hock-koon and M. Oussalah, "Defining Metrics for Loose Coupling Evaluation in Service Composition," presented at the IEEE International Conference on Services Computing, Washington, D.C., USA, 2010.
- [140] S. Alahmari, E. Zaluska, and D. C. De Roure, "A Metrics Framework for Evaluating SOA Service Granularity," presented at the IEEE International Conference on Services Computing (SCC), 2011, pp. 512–519.
- [141] B. Shim, S. Choue, S. Kim, and S. Park, "A design quality model for service-oriented architecture," in *Proceedings of the 15th IEEE Asia-Pacific Software Engineering Conference*, 2008, pp. 403–410.
- [142] R. Sindhgatta, B. Sengupta, and K. Ponnalagu, "Measuring the Quality of Service Oriented Design," in *Proceedings Service Oriented Computing. 7th International Joint Conference, ICSOC-ServiceWave*, 2009, pp. 485–499.
- [143] T. Senivongse, C. Phacharintanakul, M. Ngamnitiporn, and M. Tangtrongchit, "A Capability Granularity Analysis on Web Service Invocations," in *Proceedings of the World Congress on Engineering and Computer Science*, San Francisco, CA, 2010, vol. 1, pp. 20–22.
- [144] H. Hofmeister, "Supporting Service-oriented Design with Metrics," in *12th International IEEE Enterprise Distributed Object Computing Conference*, Munich, Germany, 2008, pp. 191–200.
- [145] L. O'Brien, J. Gibson, and J. Gray, "A Method for Selecting SOA Pilot Projects Including a Pilot Metrics Framework," presented at the International Conference on Software Engineering (ICSE), Waikiki, Honolulu, HI, USA, 2011.
- [146] "Oracle SOA Maturity Model." .
- [147] S. Inaganti and S. Aravamudan, "SOA Maturity Model," *BPTrends*, Apr-2007.
- [148] "Zapthink's Service Oriented Architecture Roadmap 3.0."
- [149] S. Wadhawa, M. Mishra, and A. Saxena, "A Network Approach for Modeling and Design of Agile Supply Chains Using a Flexibility Construct," *International Journal of Flexible Manufacturing Systems*, vol. 19, no. 4, pp. 410–442, 2008.
- [150] R. Seethamraju and J. Seethamraju, "Enterprise Systems and Business Process Agility - A Case Study," presented at the The 42nd Hawaii International Conference on System Sciences, Hawaii, 2009.
- [151] J. Banks, *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice*. New York: John Wiley & Sons, 1998.

- [152] H. A. Reijers, "Best Practices in Business Process Redesign: An Overview and Qualitative Evaluation of Successful Design Heuristics," *Omega*, vol. 33, no. 4, pp. 283–306, 2005.
- [153] R. Grady and D. Caswell, *Software Metrics: Establishing a Company-wide Program*. Prentice Hall, 1987.
- [154] J. C. Wileden, "Software Interoperability: Principles and Practice," in *Proceedings of the 12th International Conference on Software Engineering (ICSE)*, 1999, pp. 675–676.
- [155] L. C. Briand and J. Wust, "Empirical Studies of Quality Models in Object-oriented Systems," *Advances in Computers*, vol. 56, pp. 98–167, 2002.
- [156] J. DeCoster, "Overview of Factor Analysis." 1998.
- [157] J. O. Kim and C. W. Mueller, *Factor Analysis: Statistical Methods and Practical Issues*. Newbury Park, CA: Sage University Paper Series on Quantitative Applications in the Social Sciences, series no. 07-014, 1978.
- [158] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Addison-Wesley, 1998.
- [159] P. Bahrs, M. Hirzalla, S. Pappé, and R. High, "2010 3rd SOA Case Study Best Practices Conference - with focus on business agility, cloud computing, smarter planet and BPM," Academy of Technology, IBM, 2011.
- [160] "ABPMP - The Association of Business Process Management Professionals." [Online]. Available: <http://www.abpmp.org/>. [Accessed: 18-Feb-2010].
- [161] W. P. Stevens, G. J. Myers, and L. Constantine, "Structured Design," *IBM Systems Journal*, vol. 38, no. 2/3, pp. 231–256, 1999.
- [162] J. Osborne and A. Overby, "The power of outliers (and why researchers should always check for them). Osborne, Jason W. & Amy Overbay," 6-2004. [Online]. Available: <http://pareonline.net/getvn.asp?v=9&n=6>. [Accessed: 12-Feb-2011].
- [163] W. D. Gray and M. C. Salzman, "Damaged Merchandise? A Review of Experiments that Compare Usability Evaluation Methods," *Human-Computer Interaction*, vol. 13, no. 3, pp. 203–261, 1998.
- [164] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnel, and A. Wesslèn, *Experimentation in Software Engineering: An Introduction*. Boston, MA: Kluwer Academic Publisher, 2000.