

## **BSIF**

# **A Freeware Framework for Integrated Business Solutions Modeling Using Sparx Systems Enterprise Architect**

## CONTENTS

<b>1</b>	<b>Introduction .....</b>	<b>3</b>
	Agile EA .....	3
	The Platform .....	3
	The Process .....	5
	People 5	
	The Next Step .....	5
<b>2</b>	<b>An Overview of BSIF .....</b>	<b>7</b>
	BSIF Goals: .....	7
	Supported Notations & Frameworks.....	7
	Supported Approaches .....	8
	The Critical Perspectives .....	8
	Structure .....	9
	Behaviour .....	9
	Requirements .....	9
<b>3</b>	<b>The Implementation of BSIF .....</b>	<b>10</b>
	BISF Integrated Toolboxes .....	10
	BSIF Elements.....	10
	BSIF Relationships .....	11
	BSIF Extensions .....	12
<b>4</b>	<b>Using BSIF .....</b>	<b>13</b>
	Processes.....	13
	People 13	
	Downloading BSIF .....	13

# 1 Introduction

Working with a number of large and small organisations that have implemented, or are in the process of, implementing Enterprise Architect as the tool of choice for the analysis and design of business solutions, there appears to be some common recurring themes.

- Most come from Requirement Centric approaches.
- Most are attempting to move a CMMI level - typically level 2.
- Most have had previous training in EA, UML and BPMN.
- All struggle to translate theory into practical and repeatable processes for the delivery of Business Solutions using Enterprise Architect.

They also relate the same objectives for implementing tools

- Improved end to end traceability
- The capability to deliver Business Solutions using Agile approaches
- A common vocabulary across teams involved in Business Solution delivery
- The ability to transform project deliverables into a current state view of the enterprise
- The ability to use multiple notations together more effectively
- The ability to use standardised and repeatable processes
- Automated document generation.

## Agile EA

It was in the context of addressing these challenges that an idea, that ended up as BSIF, was born. I called it Agile EA. Agile EA was made up of three parts (Platform, Process, People).

### The Platform

The first challenge was to create a hierarchy of requirement elements that assisted in the process of translating business needs into business solutions and this resulted in a number of elements.

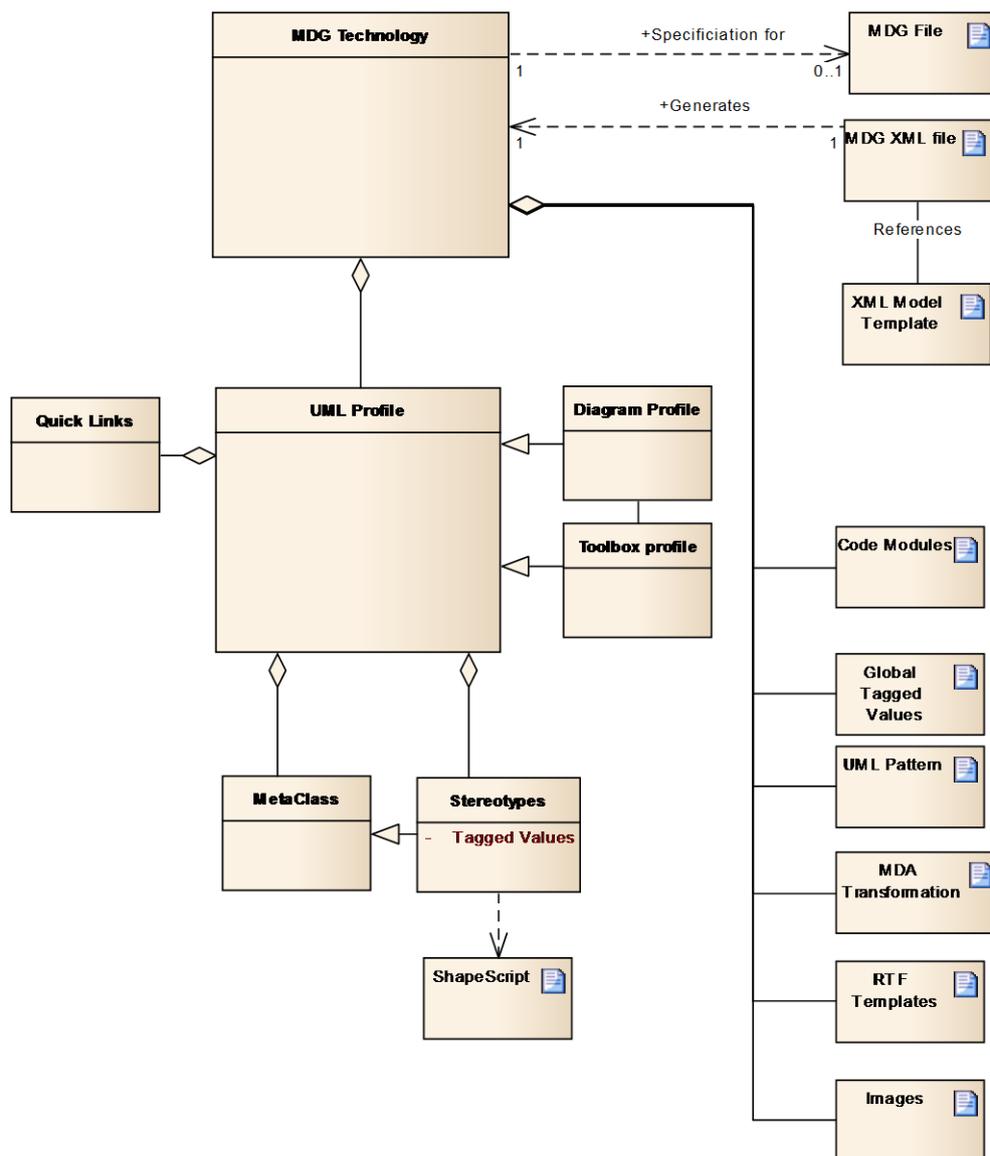
- Objectives,
- Functional Requirements
- Rules
- Non Functional Requirements.

In addition to this I needed to address documentation - how to generate documents from within EA. The approach I ended up using was to create some additional documentation elements that would allow version history tables and distribution lists to be modelled in EA and so these were added to the profiles I had created.

## MDG Technologies

At this stage things started to come together and with a lot of late nights spent learning about Profiles, Shape Script and MDG technologies. MDG technologies were the key to it all. They allowed the elements I had created to be put in toolboxes that were then associated with the custom diagrams I created. I could also include template references, RTF templates, Tagged values and other goodies within my MDG technology. I then discovered that if I created and then deployed to MDG I could use it to build a model template - one that had my diagrams and elements and Document Models built in. This was fantastic stuff and the first version of Agile EA was produced and put into production.

Below is the domain model for MDG Technologies



## The Process

The Agile EA MDG was great and I then found that I could reuse the elements on new assignments making a few changes here and there to suit the clients need. However a platform is only useful once you know how to use it. Interestingly this is the very problem that stopped the organisations I worked with making effective use of EA. And so the approach was expanded to include guides that explained the toolboxes and the model templates and how to use the two to populate a project model and then to generate the documents required. And so I created detailed guides that required only a basic understanding of modelling techniques to enable the delivery of quality standardised outputs from my models. And once again I found that, with a few changes I could reuse the guides.

It was around this time that I realised that I was developing an approach to using EA that not only provided a solution to clients that achieved all their goals but also an approach to implementing EA in organisations.

Up until this point the process was focused on project deliverables but this then changed and the repeatability and reuse we were achieving in these areas was applied at the enterprise level.

Standard project templates were used to capture information in a structured manner and this allowed us to create processes that harvested this information and moved it into Current State models of the enterprise. We developed processes that supported reuse key information between projects and Agile EA was becoming a powerful approach.

## People

The final part of the equation was building capability in the people using the framework and a realisation that traditional approaches to training in UML, BPMN and EA didn't work.

The result was a layered and modular approach that focused on showing people how to use the notations to produce real deliverables using the templates I had built for the organisation.

Agile EA was a huge success and was implemented in over 11 organisations. Each time new elements and processes were added to improve the approach.

## The Next Step

While Agile EA was successful it had a number of limitations mainly due to the fact that it evolved as I went. I reached the point where it had ended its useful life and needed to be redeveloped or replaced. I decided to replace it and resigned from the consulting company where I had created Agile EA to build something better.

The main problems with Agile EA were its lack of integration and usability

- It wasn't linked to the notations it was used in conjunction with
- It was not aligned with common approaches used such as IIBA and Volere
- It lacked a nice and flexible presentation in EA
- It was focused on discrete areas of business solutions
- It had a limited vocabulary as a result.

Five months later and I now have my replacement to Agile EA called BSIF. BSIF is the Business Solutions Integrated Framework. It is an integrated requirements model designed to assist organisations implementing Enterprise Architect as the tool of choice for Analysis and Design. BSIF takes all of the strengths of Agile EA and addressed all of the limitations

How do they compare - well If Agile EA was a steam train then BSIF is a Diesel. It does the same thing but is much more powerful!

# 2 An Overview of BSIF

## BSIF Goals:

The goal of BSIF was to create a new requirements framework that could span and integrate the existing notations and frameworks that modellers are using. BSIF is not designed to be a replacement for any of the other notations - It was designed to complement and to facilitate the exchange of information and improve traceability between Analysis, Architecture and Design practitioners. It was also designed to be much more intuitive to use.

## Supported Notations & Frameworks

To support this goal we selected to integrate BSIF with the internationally accepted Notations and Frameworks that when combined provide complete coverage of all areas of Business Solution Delivery.

## TOGAF

The Open Groups Architectural Framework (TOGAF) is an Architectural Framework, which provides for the four architectural domains that are the accepted subsets of Enterprise Architecture

- The **Business Architecture** defines the business strategy, governance, organization, and key business processes
- The **Data Architecture** describes the structure of an organization's logical and physical data assets and data management resources
- The **Application Architecture** provides a blueprint for the individual application systems to be deployed, their interactions, and their relationships to the core business processes of the organization
- The **Technology Architecture** describes the logical software and hardware capabilities that are required to support the deployment of business, data, and application services. This includes IT infrastructure, middleware, networks, communications, processing, standards, etc.

TOGAF provided a number of key components to the framework.

- TOGAF Business Architecture supplied traceability at the Business Motivation level
- The ADM provides a structure that can be used to store information captured at different levels facilitating cooperation between architects and other practitioners
- The TOGAF Enterprise Continuum is a well defined structure for storing other frameworks and solutions
- TOGAF was designed to be adapted and integrated.

## Archimate

Archimate presents a unified way of modelling enterprise architectures, integrating the architecture domains for Business Architecture, Information Architecture, Application Architecture and Infrastructure Architecture in a way that makes the models easy for decision makers to understand and read. Archimate also includes a big focus on Services within the various architecture domains, which supports the adoption of Service Oriented Architecture approach.

Archimate and TOGAF go hand in hand and being able to integrate solution delivery with the architectural models and frameworks.

### **Business Process Modelling Notation (BPMN)**

BPMN provided the International notation for Business Process modelling that supports SOA implementation as well as simple process mapping.

BPMN is ideal at the lower levels of process modelling but needed alignment with the higher level constructs such as Business Services, Functions and Process frameworks. As well as this we needed to plumb it into our requirements hierarchy.

### **Unified Modelling Language (UML)**

UML provided a single integrated modelling notation that supported software development activities from analysis and design through to implementation phases of the SDLC.

Agile EA was focused on UML and so it was more of a matter of automating the linkages that had been identified.

### **Supported Approaches**

For BSIF to be of any practical use it was also important that we supported the approaches that organisations were adopting for solution delivery.

### **IIBA**

The requirements framework is closely aligned with the requirements types identified in BABOK 2.0 and supports the techniques that related to modelling based requirements management.

### **Volere**

Requirements elements in BSIF capture the attributes that Volere uses allowing Volere practitioners to use the process and template within Enterprise Architect.

### **Agile**

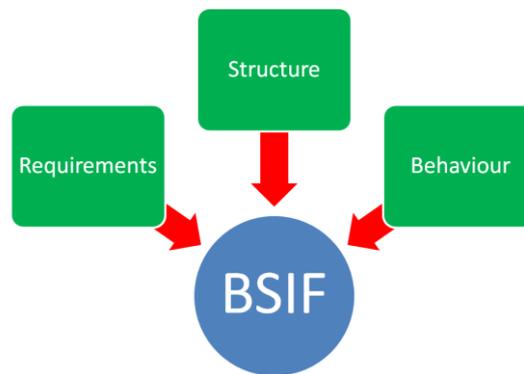
BSIF supports and encourages Agile approaches

- The inherent traceability that BSIF provides supports the management of change within solutions
- BSIF provides a common requirements vocabulary for all practitioners ensuring that people can effectively communicate and collaborate
- The effective decomposition of Business solutions into slices that can be delivered in an iterative manner ensure that the framework is ideal for Agile projects
- Lastly BSIF is an open framework that can be used as people see fit.

### **The Critical Perspectives**

When discovering the best approach to integrate the notations I used three perspectives that were common to all notations.

Structure - Behaviour - Requirements



In developing BSIF the aim was to provide traceability between them all and the key to BSIF was to introduce a robust requirements hierarchy that spanned all the notations and support linking of the elements into a single framework.

### **Structure**

We need to be able to describe the structure and static relationship of the business solution including.

- Organizational Structure
- Process Frameworks
- Business Entities
- Application Components
- Technological Components.

### **Behaviour**

We need to be able to model behavioural aspects of the business solution.

- Business Motivation
- Business Interaction
- Business Processes
- System Interaction
- Application Messaging
- Communication links.

### **Requirements**

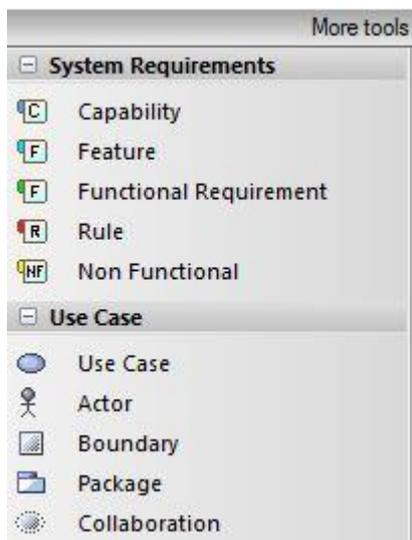
We also need to be able to communicate the requirements that exist and drive these other views of the business solution.

- Business Requirements
- Business Drivers
- Business Goals
- Objectives
- Capabilities
- Functional Requirements
- Business Rules
- Non Functional Requirements.

# 3 The Implementation of BSIF

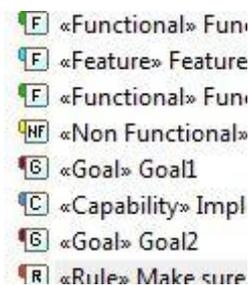
BSIF has been implemented using an Enterprise Architect MDG technology. This allows the elements, diagrams, Quick links and other customisation to be packaged and deployed to users of the tool.

## BSIF Integrated Toolboxes



## Features

- Colour Coded elements
- Clearly labelled
- Fully Integrated with the notations elements
- Easy to differentiate in the Project Browser



## BSIF Elements

### Standard Attributes

Standard fields have been built into the elements providing structured collection of the information that supports IIBA and Volere based approaches to requirements.

BSIF::BSIF Requirement (Rule6)	
Customer Dissatisfaction	0
Customer Satisfaction	0
Date Identified	
Fit Criteria	<memo>
ID	
Last Reviewed	
Priority	Proposed
Rational	<memo>
Rendering	BSIF
Source	
BSIF::Rule (Rule6)	
Rule_Type	Business

## Image Control

BSIF elements include a rendering tagged value that can be used to control the details they are shown on diagrams.

These include:



### Native

Renders elements in native EA format.

This enables Compartment visibility to be set using Feature Visibility



### BSIF

Elements are shown with colour coding and selected information



### BSIF+ Notes

This controls rendering of the notes section of the element

Notes:  
The notes Section



### BSIF+ Details

This controls rendering of selected details of the elements

Details:  
ID:  
Status: Proposed  
Source: Ross Park



### All

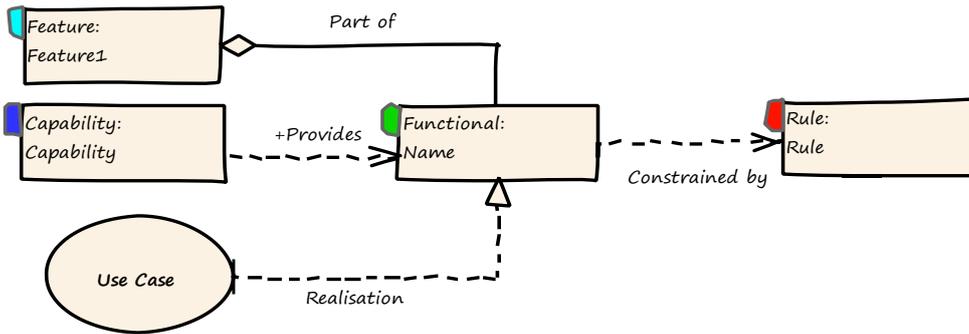
Controls can be used together

Details:  
ID:  
Status: Proposed  
Source: Ross Park

Notes:  
The notes Section

## BSIF Relationships

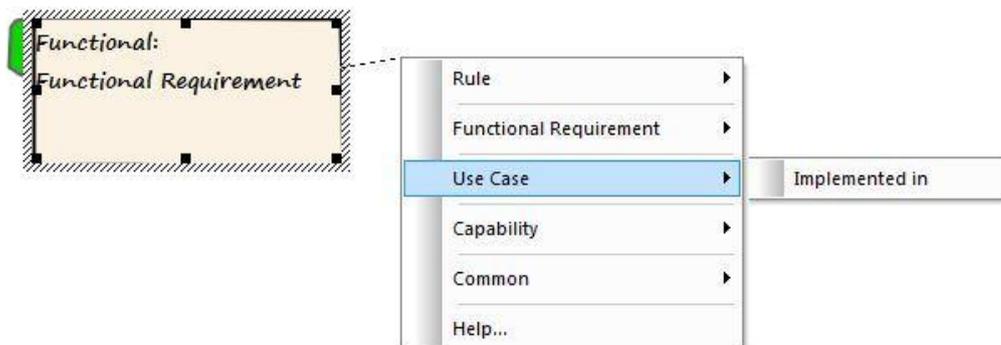
BSIF requirement types are based on the relationships that the elements can have with other BSIF elements as well as elements in the supported notations. These relationships come from a Meta model that has been developed in which all relationships within and between, the supported notations elements have been carefully considered and then mapped to the BSIF elements.



The mapping for a Functional Requirement

### Quick Linking

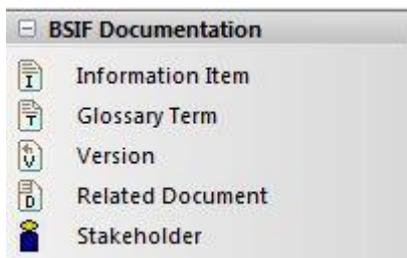
As a result of the Meta model we have been able to make all elements in BSIF "Quick Link" aware of supported notations. This means that if you drag a quick link connector to or from a BSIF element you will be prompted to select the correct relationship thus making adoption of the framework more intuitive when working between notations.



Quick linking in BSIF

### BSIF Extensions

There are also some Enterprise Architect extensions that have bundled with BSIF that I have repeatedly found have improved adoption of the tool. The key one is a documentation toolbox that supports the complete creation of complex documents from the models.



# 4 Using BSIF

BSIF was developed to support effective Business Solution delivery using modelling notations and so is ideal for organisations that are looking at using EA as the tool of choice for analysis and design.

Because BSIF evolved from years of experience discovering how to make Enterprise Architect useful for practitioners involved in Business Solution delivery it is a very pragmatic framework that bridges the gap between theory and practical use of the tool to produce deliverables.

## Processes

BSIF has been carefully designed to support agile work practices based on collaborative modelling. At the centre of this is modelling approaches that have deliverables that integrate phases of solution delivery and make use of shared reference information and framework.

## People

We also have available a complete and integrated range of training modules designed to support the framework and the processes designed around BSIF.

Implementing Enterprise Architect as the standard for Business Solution delivery is a complex undertaking that includes matching and customising the framework, the supporting processes and training, to the organisations needs. It is a journey that has been different for each of the clients I have worked with. Starting with BSIF gets you 80% of the way - the other 20% you have to work out as you go along.

This article focused on the platform perspective of BSIF which is the MDG technology and you can have this part for free.

The rest (our experience) is available via our consulting and training services.

## Downloading BSIF

The BSIF MDG can be downloaded Free from our website.

[www.parkconsulting.co.nz](http://www.parkconsulting.co.nz)

BSIF as well as supporting documentation and guides in the use of the framework that will get you started is provided as free to anyone who wants to use it.

If you have any comments for questions I can be contacted at:

Ross@ParkConsulting.co.nz