



POLICY ADMINISTRATION SYSTEM UPGRADE

– An Agile, Lean, and Flexible Approach

Abstract

The insurance industry has always thrived on changes, which intrinsically makes upgrades and changes to policy administration systems unavoidable. Periodic upgrades to newer versions are necessary, as well as advantageous. Upgrading not only brings new features to help insurers keep up with their competitors, but also provides an opportunity to renew and/or optimize existing business rules to make them more relevant. With most insurance policy systems being either off-the-shelf, or XML-based, we can utilize the strengths of XML to upgrade their business rules with an agile and adaptable approach. This article discusses an efficient approach to upgrade an XML-based policy administration system by creating a tree-based relationship structure of business rules. This structure will then be parsed, while filling the upgrade gaps for a successful business rule upgrade.



The history of insurance is similar to 'A Tale of Two Cities,' entailing the development of the modern business of insurance against known and unknown risks, including risks arising from fire, property, casualty, automobile accidents, and medical treatment. The insurance industry helps us to alleviate risks by adopting various risk-mitigation approaches and spreads these risks from the compounded liability of a single individual, to the larger community as a whole. Additionally, it also provides an important source of long-term finance for both the public and private sectors.

Evolution

The insurance sector has been evolving since its inception, fostering the

changes required to adapt to changing demographic profiles, customer needs, regulatory changes, micro- and macro-economic conditions, globalization, technology trends, and reduced product development time-to-market among a number of items in the list. But with external factors constantly changing the business environment and evolving customer needs, insurance companies are compelled to bring operational agility by transforming internal / external processes, systems, and customer experience.

In the recent past, insurance companies have started their transformation journey with Policy Administration Systems that deal with new businesses, policy servicing,

and other administration activities during the policy lifecycle. Having said that, an efficient Policy Administration System has always been a pressing need for efficient operations for the insurance industry. Over the last decade, there has been a radical shift from home-grown policy administration solutions to third party product-based solutions, as these products reduce implementation cost, streamline policy administration business processes, and provide next-generation functionalities that insurers have always strived for - efficient operations. Further, it helps insurers build sustainable competitive advantage in the long run.

Approach

Adopting third party product-based solutions makes it imperative for insurers to upgrade their systems (deployed in the product platform) on a periodic basis in order to effectively leverage all of the new features and functionalities and stay ahead in the competitive market place. Any two versions of Policy Administration Systems will have gap(s) between them that will have to be filled effectively and efficiently for a successful version upgrade. Version upgrades should also be used as a platform to optimize existing business rules by

leveraging the new features available in the newer version. The various gap(s) between the versions can be categorized as functional and structural gap(s).

- Functional gap(s) are the functionalities that are either present in the source system, or in the target system, but not in both.
- Structural gap(s) are differences in syntax, data structure incompatibility, program constructs, deprecations, etc. Structural gap(s) can further be

categorized into user merge gap(s) and system merge gap(s)

- User merge gaps are those in which business / SME intervention is required
- System merge gaps, on the other hand, can be automated

A unique and differentiated approach to perform a business rules upgrade using an agile, lean, and flexible rule-relationship inventory, prepared just-in-time, is highlighted below:

Step 1

Extract the business rules, categorize them based on rule types and their associated functionalities, and build a classified business rule inventory

Step 2

Identify the functional and structural gap(s) between the source and target versions of the identified systems

Step 3

Fill the functional and user merge gap(s). Leverage an automated adaptive gap filler algorithm to transform business rules inventory.

Step 4

Review each transformed business rule, with respect to functional and structural gap(s)

Step 5

Functional and comparative quality assurance between the source and target versions of the policy administration system

Step 6

Elevate the business rule inventory to an integrated environment for an integration testing of all inbound and outbound integration points

Step 7

Elevate the transformed and tested business rule inventory to the destination system

Step 8

The data migration approach and detailed plan will be based on business needs

Rules Relationship Tree

The Rules Relationship Tree is a flexible rule-relationship inventory prepared just-in-time. It can be created using

the available XSD or by using the following algorithm: RRT (Rule Relationship Tree) =

BRT (Backward Relationship Tree) + FRT (Forward Relation Tree).

Step 1

Select a business rule

Step 2

Parse the business rule to find all other legitimate business entities referenced in it and augment it to FRT, recursively. At the same time, update the BRT to relate all the referenced entities with the selected business rule.

Step 3

Parse all the business entities outside the purview of RRT to find additional references of the selected business rule. Append it to existing in memory FRT and BRT accordingly.

Step 4

Consolidate BRT and FRT to form a complete RRT

Adaptive gapfiller algorithm

Step 1

Prepare an inventory list of user and system gap(s)

Step 2

Arrange the gap(s) in a sequential order based on program construct

Step 3

Perform step 2 for user and system gap(s)

Step 4

Select the business rule for the business rule inventory

Step 5

Parse every line of selected business rule

Step 6

Perform textual search on each line for user and system gap(s)

Step 7

Prepare Rule Relationship Tree (RRT) using forward and backward references

Step 8

Prepare the list of gap(s) identified and spotted in RRT

Step 9

Fill the gap(s) dynamically based on gap types

Step 10

Store the transformed rule in staging area

Step 11

Perform steps 5 to 10 for the entire business rule inventory

Step 12

Prepare transformed business rule inventory with user and system gap(s) filled in

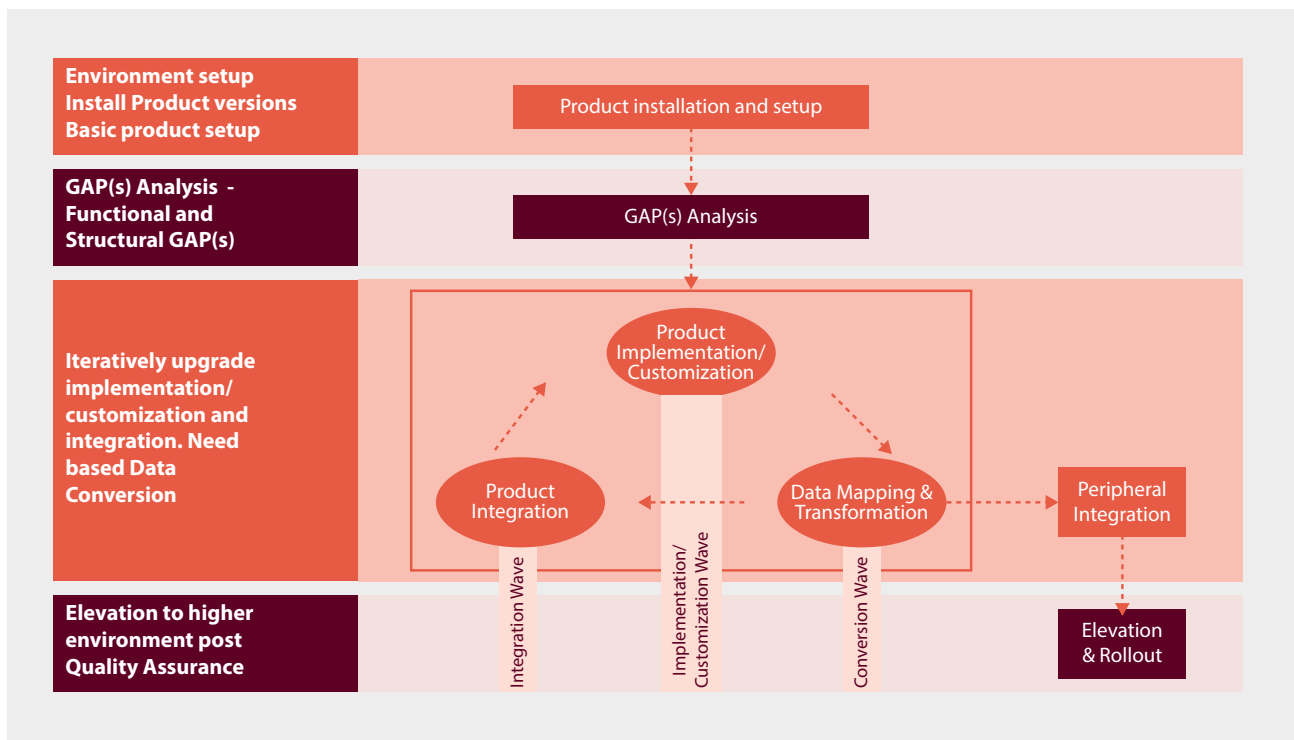


Product version upgrade

The product version upgrade task is split across a series of activities, out of which a few of them are performed in sequential and others in parallel. At conceptual level, product version upgrade begins with establishing product environment and setting up vanilla version. Post installation and basic configuration in the

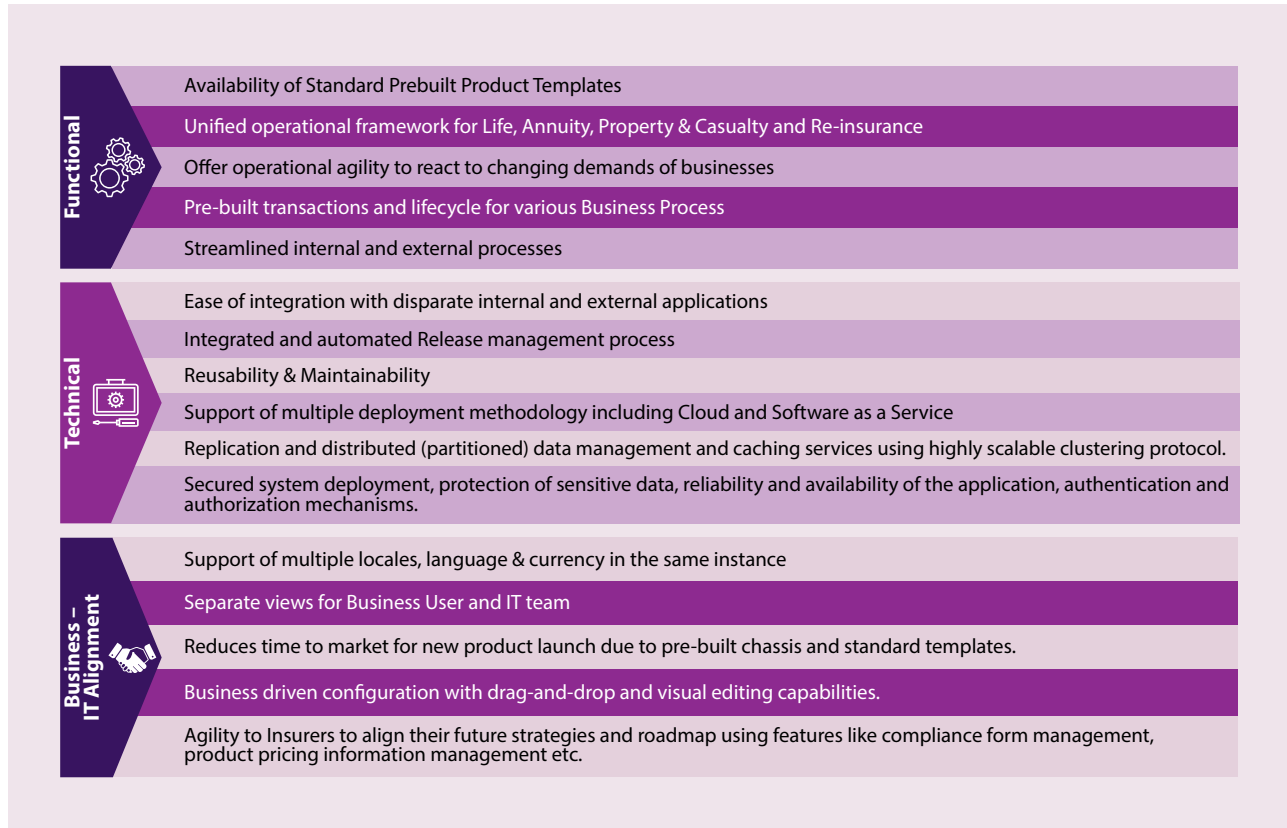
product, perform FIT-GAP analysis from functional (features, functionalities, etc.) and structural (syntax, program constructs, etc.) perspectives. After that, commence three waves starting with implementation and customization, integration and data conversion in iterative or agile or hybrid development methodology to upgrade

entire set of business rules, data (look-up and transactional) and integration points (including internal & peripheral interfaces). Finally, validate (quality assurance) upgraded business rules, data set, and integration points, and make it go-live.



Key considerations

Before upgrading to a newer version of a Policy Administration System, several key considerations that help achieve a smooth and successful upgrade should be taken into account. These consideration aspects will form a vision and plan for the upgrade.



Critical success factors

- Leadership support and sponsorship
- Product vendor and business SME participation
- Dedicated project and scope management
- Well established change management, communication plan, and training plan
- A defined data and business rule upgrade strategy
- Dedicated support from a cross-functional team within the organization and client SME
- A core project team composed of full-time employees, including a project manager, technical architects, and other representatives from core areas of the insurance business
- Adequate budget allocation
- Streamlined implementation, conversion, and integration methodology
- Proper documentation for all ongoing activities
- Systematic framework for identifying gaps and resolving them
- Well-defined upgrade approach - Big Bang / Staggered / Line-of-Business specific, during the planning phase
- Committed derisking approach and plan

Looking ahead

The dramatic changes in technology as well as the insurance business landscape, over the last decade, further augmented by economic, demographic, consumer behavior, and socio-cultural factors, had made the modernization of the insurance business inevitable. Commercial-off-the-

shelf (COTS) packages emerged as effective tools that significantly helped insurance carriers in modernizing their policy administration functions. This helped them cope with critical challenges of an inadequately skilled workforce, frequent regulatory changes, budget cuts, and the increasing operational cost of maintaining legacy IT systems. These third party package-based solutions / COTS enable insurance carriers to stay focused on their core business, while the operational work is seamlessly performed by the COTS packages. Most insurance carriers hold the myth that transforming their current policy or claim systems to a COTS package

will provide them the needed competitive advantage and that this completes their transformation journey. However, in reality, unless these COTS packages are properly configured to meet the business requirements and upgraded on a timely basis, they will not provide the necessary operational agility, competitive advantage, or cost competitiveness.

Therefore, it is of utmost importance to upgrade COTS packages, not only to reap the benefits of all the new functionalities – including technical and business features – but also to stay ahead and relevant in marketplace. Additionally, while upgrading

a COTS package, it is imperative to follow a defined and robust framework and strategy. This ensures that all components, including business rules and data, are elevated to the newer version and working as expected. According to the reports from leading analysts, most North America-based insurance carriers are upgrading their existing COTS packages or moving away from their legacy systems to COTS packages. This upgrade approach can be used seamlessly with any XML-based COTS package, such as Oracle's Insurance Policy Administration System(OIPA), and can also be leveraged for other XML-based Policy Administration System upgrades.





Ravi Shankar Jha

Lead Consultant, Insurance Practice, Infosys

Ravi has been working as a techno-functional consultant in the insurance domain including Life & Annuity, Property and Casualty for over 11 years. He specializes in niche insurance products including Oracle Insurance Policy Administration System, Oracle-InsBridge, and IBM Cognos ICM. He has a good focus on insurance package solutions, and he is also involved in conceptualizing and designing IT solution for leading insurance carriers.

Ravi is a Certified ScrumMaster (CSM) and also earned certifications from the Industry Association: Associate in General Insurance (AINS) from AICPCU and FLMI Level I from LOMA.



Deepu George Philip

Senior Consultant, Insurance Practice, Infosys

Deepu has covered significant ground in the insurance domain, across health, life, P&C insurance, and healthcare providers, for over 10 years. He specializes in niche products for both insurance and provider domains. The techno-functional aspects of the insurance industry account for his primary interests, and he has helped conceptualize and create effective solutions using commercial-off-the-shelf products and customizing them to suit individual businesses.

Deepu is a Certified ScrumMaster (CSM) and a Project Management Professional (PMP). He also holds an Associate (ALMI) designation from Life Office Management Association (LOMA) and various certifications from America's Health Insurance Plans (AHIP).

For more information, contact askus@infosys.com



© 2018 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.