ISSN: 2230-9926

Available online at http://www.journalijdr.com

IJDR

International Journal of Development Research
Vol. 14, Issue, 12, pp. 67293-67298, December, 2024
https://doi.org/10.37118/ijdr.29109.12.2024

RESEARCH ARTICLE OPEN ACCESS

DIFFTREE/DIFFITEMS REST API FOR GUIDEWIRE POLICY CENTER CLOUD IMPLEMENTATION

***Shanmugasundaram Senathipathi**

Manager, Capgemini, Financial Services, Woodland Hills, California, United States

ARTICLE INFO

Article History:
Received 18th September, 2024
Received in revised form
19th October, 2024
Accepted 11th November, 2024
Published online 30th December, 2024

Key Words:

Insurance Technology, Guidewire Policy Center, Guidewire PolicyCenter Diff Tree, Guidewire Integrations, Guidewire Insurance Suite Cloud API.

****Corresponding Author:***
Shanmugasundaram Senathipathi

ABSTRACT

Aims: This study provides a high-level design and technical solution for comparing two PolicyPeriods in Guidewire Policy Center application and exposing the differences as a REST API service, The differences provide payloads containing diffItems and that can be sent to downstream system via KAFKA. It also addresses the Guidewire application Out-Of-The-Box limitation which is currently not offering this solution. This new solutioning framework is flexible, reusable, and scalable for various integrations in Guidewire InsuranceSuite products. ***Study Design:*** This study outlines the technical solution to address the limitations of Guidewire PolicyCenter's OOTB Difftree functionality, and focusing on usability, Scalability, architectural design for exposing diffItems via REST APIs and KAFKA payloads. ***Place and Duration of Study:*** The analysis was performed between December 2023 and July 2024, based on real time project situations from an array of software development environments, project scenarios and feedback from customers. ***Methodology:*** The methodologylaid out in this paper includes systematic approach to addressing the limitations of Guidewire PolicyCenter's out-of-the-box Difftree functionality and outlines the technical solution designed to expose the Difftree functionality as a REST API Service and provide diffItems JSON payloads through KAFKA messaging for downstream/external vendor systems. ***Results:*** The end results provide a sturdy, scalable, and high-performing framework for comparing Policy Periods, utilizing a REST API to provide diffItems and KAFKA to distribute payloads, this enhances the accessibility, usability, and efficiency of the Difftree functionality in Guidewire Policy Center, enabling seamless integration with external systems and addressing essential business needs. ***Conclusion:*** The study concludes that the Out-of-The-Box Difftree/DiffItems functionality in Guidewire Policy Center comes with significant limitations, such as being restricted to UI users, only comparing immediate predecessor PolicyPeriods,and being incapable of making the data usable by external systems or exposing the capability as a service. To overcome these challenges, a custom framework was developed that is flexible, scalable, and reusable.The proposed solution successfully exposes the Difftree functionalityas a REST API service to external systems with filtering capabilities and a KAFKA payload delivery mechanism that uses events to provide unfiltered diffItems to downstream systems.

Citation*: *Shanmugasundaram Senathipathi. 2024. *"Difftree/Diffitems rest api for Guidewire Policy Center Cloud Implementation". International Journal of Development Research*, 14, (12), 67293-67298.

1.INTRODUCTION

The objective of this article is to present a high-level design for comparing the two PolicyPeriods and exposing their differences via REST API services. In the current implementation, we call this functionality as difftree or diffItems and used by upstream systems as diffItems API and with slight modification, the same functionality is also included in the KAFKA payload sent to downstream systems. The comparison is not just limited to entities, but also extended at the field level. The Intended functionality is designed in such a way that it can compares any two PolicyPeriods, regardless of whether they are part of the same Line of business (LOB) or the same Policy or the same Job (as job versions) - These functionalities has been validated as separate scenarios. The aim of the current implementation is to compare the PolicyPeriod with either any future-dated transaction or its immediate predecessor, known as the based-On period.The latest Guidewire cloud version – CORTINA, does not have this feature to publish the difftree as a REST API service. In the current implementation, the service addresses the following challenges and the solution approach is highly adaptable for any other implementation needs.

1. The list of entities or fields added/ modified / deleted from based On policy?
2. The list of entities or fields added/ modified / deleted from a future dated policy for a given PolicyPeriod and these two scenarios exposed at policy and job level

a) https://{baseUrl}/{client}/job/v1/jobs/{jobid}/diffitems for a given {{jobId}}
b) https://{baseUrl}/{client}/policy/v1/policies/{poicyid}/diffitems for any given {policyId}

2.METHODOLOGY

2.1.Use Cases

The diffItems functionality can be categorized under two broad headings as below.

1. Provide required diffItems over the REST API service
   a) Compare the two PolicyPeriods based on Current transaction
   b) Compare the current PolicyPeriod with based-On PolicyPeriod.
2. Compare the current PolicyPeriod with future-dated / backdated PolicyPeriods for a given slice date.Filter the list of diffItems as per external system requirements
3. Provide all diffItems (without any filters) as a part of KAFKA payload to the downstream systems

1. **Vehicle exposure – Entity level:** A new vehicle AUDI Q5 2018 was added and the existing vehicle was modified from 2010 Honda Civic to 2020 Ford Explorer

Table 1. Vehicle Exposure Old Value vs New Value

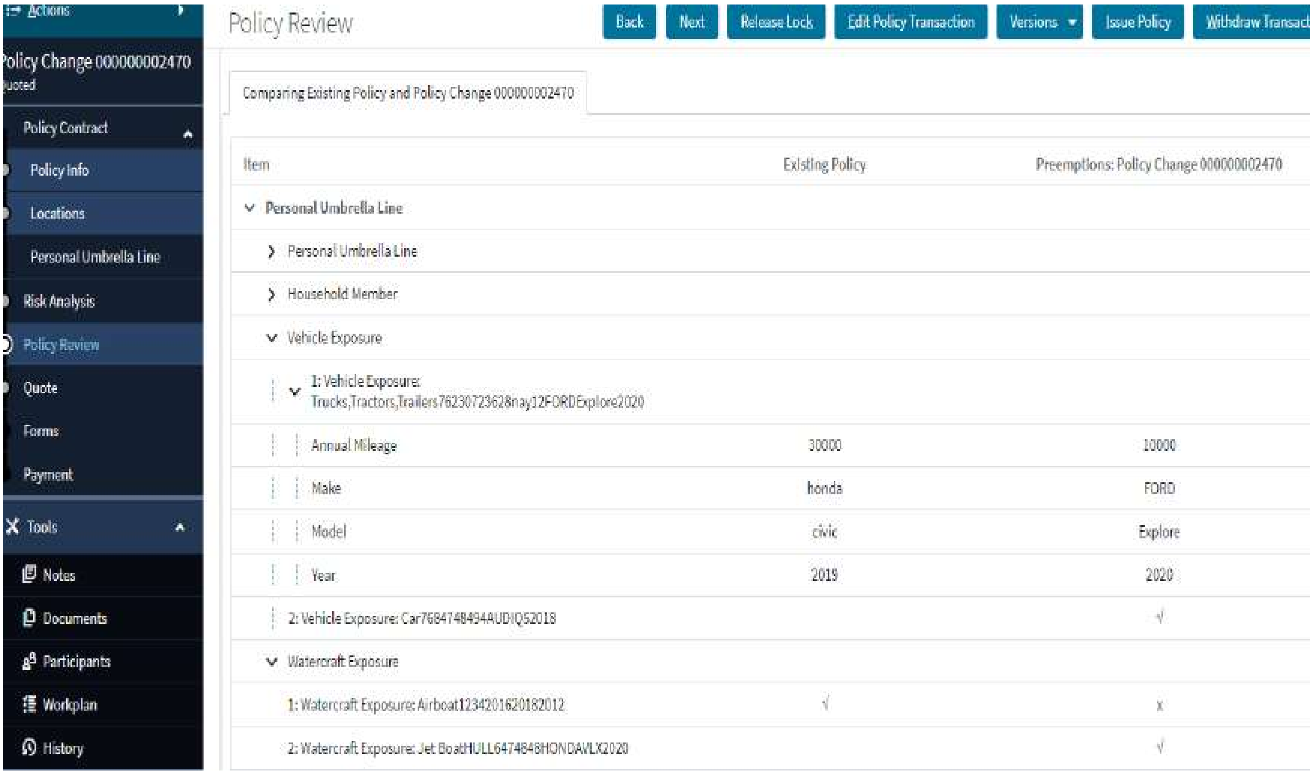
Field name	Old Value	New Value
Annual mileage	3000	1000
Make	Honda	Ford
Model	Civic	Explorer
Year	2019	2020

2. **Watercraft Exposure - Entity level**
   a) An old Watercraft (1) was deleted
   b) New Watercraft (2) was added for the current period

2. 4. Solution

2.4.1. Business Context

Consider a typical scenario where a policy is bound and issued on 1st Jan 2021 for a one-year term with policy expiring on 31st Dec'21. The policy undergoes a PolicyChange transaction as per figure:2.



Policy Review

Comparing Existing Policy and Policy Change 000000002470

Item	Existing Policy	Preemptions: Policy Change 000000002470
Personal Umbrella Line		
Personal Umbrella Line		
Household Member		
Vehicle Exposure		
1: Vehicle Exposure: Trucks,Tractors,Trailers76230723628nay12FORDExplore2020		
Annual Mileage	30000	10000
Make	honda	FORD
Model	civic	Explore
Year	2019	2020
2: Vehicle Exposure: Car7684748494AUDIQ52018		√
Watercraft Exposure		
1: Watercraft Exposure: Airboat1234201620182012	√	x
2: Watercraft Exposure: Jet BoatHULL6474848HONDAVLX2020		√

Figure 1. Guidewire PolicyCenter Policy Review Screen

2.2.What is Difftree?

Guidewire PolicyCenter UI provides Policy Review screen to display the difftree functionality and supports following transactions only [1].

a. PolicyChange, Cancellation, and
b. Renewal transactions.

2.3.Example of a Difftree

As shown in figure:1 below, Retrieve the policy change transaction and navigate to policy review screen on left hand menu on GW PolicyCenter. A Policy Change is quoted after the Policy is bound and issued; it compares the current period for PolicyChange#1 with the previous PolicyPeriod Issuance at every entity level and at every property level [1].

Based on the current date (as of Jun'21), there are 3 future-dated Policy Change transactions and the difftree service generates 3 diffItems (diffItems #1,2, and3). As of slice date, there are 2 future-dated PolicyChange transactions and generates 2 diffItems (diffItems #2, and3) [4].

2.4.2. Limitations on Guidewire PolicyCenter– OOTB

GW Policy Center has difftree functionality, but it does not fulfil requirements due to following limitations.

1. **Limited to UI users only:** Guidewire's OOTB Difftree functionality is limited to only PC-UI users, and external users cannot access the functionality.
2. **Comparison based on 'basedOn' period:** The GW PC UI compares the current PolicyPeriod transaction with its

immediate processor PolicyPeriod (called basedOn) Period only.

3. **Not able to be consumed by third party:** The field names shown on the GW UI screen are not same as the of the database field names. So, it is not readily consumable by the downstream system.
4. **No service enabled:** GW OOTB doesn't have a REST / SOAP based solution to expose the functionality.

2.5.Technical Solution

The objective of the technical solution is to overcome above limitations from Guidewire PolicyCenter-OOTB solution. Also, address two main use cases such as: a) to provide diffItems over REST API service and b) to provide payload to the downstream systems. To optimize the code reuse, we've built a single set of components that can be leveraged for both REST API needs and downstream payload needs, using following components [1] [5].

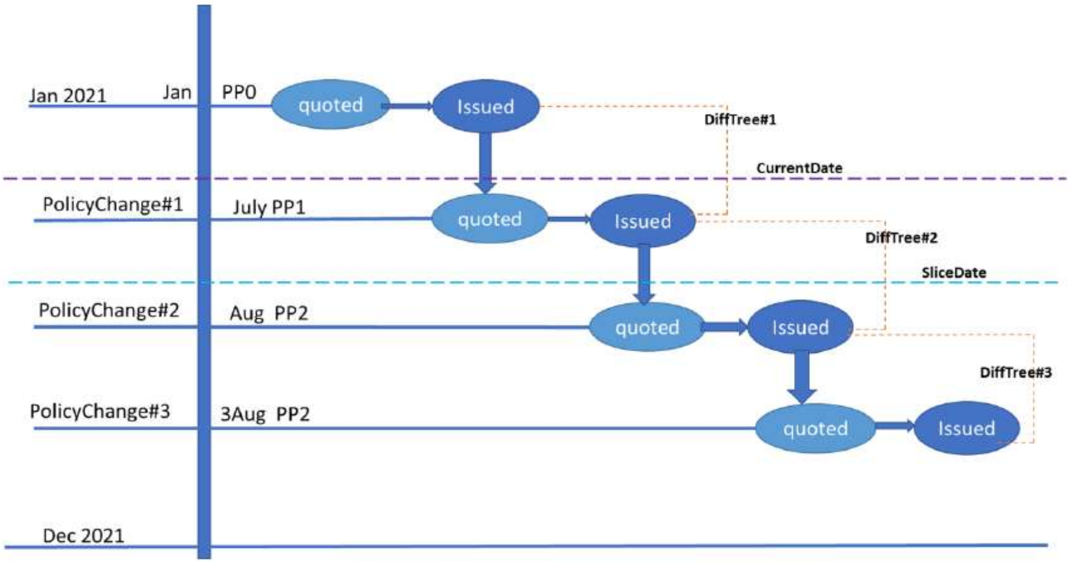


Figure 2. Policy Change Transactions

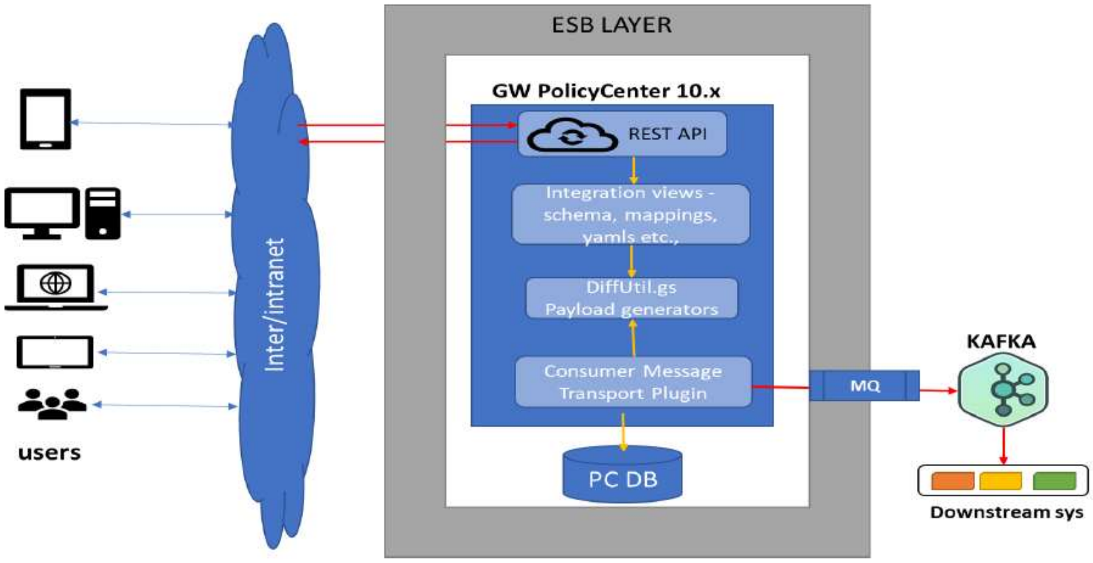

ESB LAYER
GW PolicyCenter 10.x
REST API
Integration views - schema, mappings, yamls etc.,
DiffUtil.gs Payload generators
Consumer Message Transport Plugin
PC DB
MQ
KAFKA
Downstream sys
Inter/intranet
users

Figure 3. Enterprise Service Bus for DiffItems Payload

Table 2. ESB Layer Components

Component Name	Description
GW PolicyCenter10.x	PolicyCenter is a transactional system supports policy life cycle - draft, quote, issuance for all policy transactions – cancel, PolicyChange, cancel, renewal, reinstatement, audit etc.
ESB – GW Gateway	ESB layer surrounds the PolicyCenter, any interaction with the external system must go thru ESB channels
GW IS REST API	A set of interfaces exposed by GW PolicyCenter for external interaction.A framework for defining RESTful web services- support, GET, put POST, PATCH, delete, based on swagger schema.
Integration views	An external view of the GW domain model Produces the payload in JSON format, significantly faster than XML in terms of serialization and data extraction with reduced payload size. Made up of schema, mapping, and domain model.
Message Transport	A GW proprietary component to support the asynchronous message processing.
DiffItem Utils	A core component to generate the payload using the integration views and filters.
KAFKA	External system to transform the GW raw payload as per consumer needs.
Downstream system	A set of consumers subscribed for KAFKA generated transformed payload (pub-sub model)

2.6. Sequence Diagrams

This section describes various GOSU components involved in supporting a) retrieval of diffItems using REST API and b) send diffItems as a part of KAFKA payload.

2.6.2. Send diffItemsas a part of KAFKA Payload

The following sequence diagram indicates list of events while sending payload to the downstream system, supports use case#1

2.6.1. Retrieve diffItems using REST API

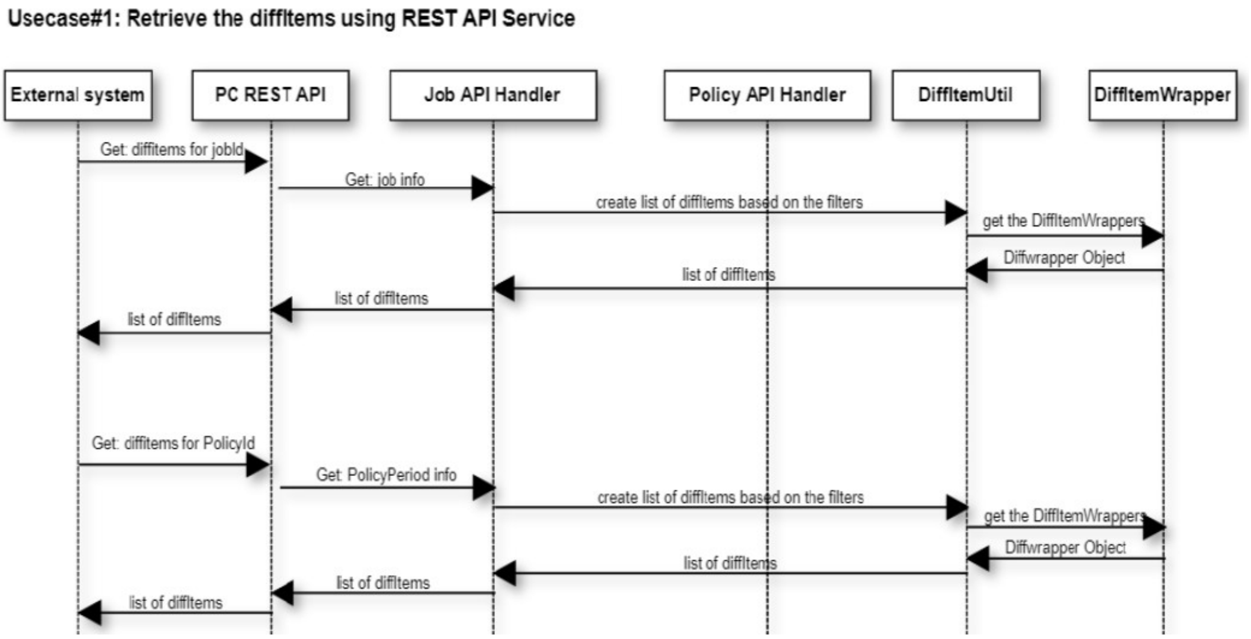


Figure 4. Rest API Service for DiffItems

Table 3. DiffItems/Rest API Service Components

Component Name	Description
External system	The caller / consumer of this service
GW Gateway	Gateway to the PolicyCenter REST APIs
JobHandler	Handles all requests for a given {{jobid}}
PolicyHandler	Handles all requests for a given {{policyId}}
DiffUtil	Generates the required payload 'basedon' the integration view components
DiffItemWrapper	Encapsulates the various attributes of a diffItems such as status, current, previous etc.

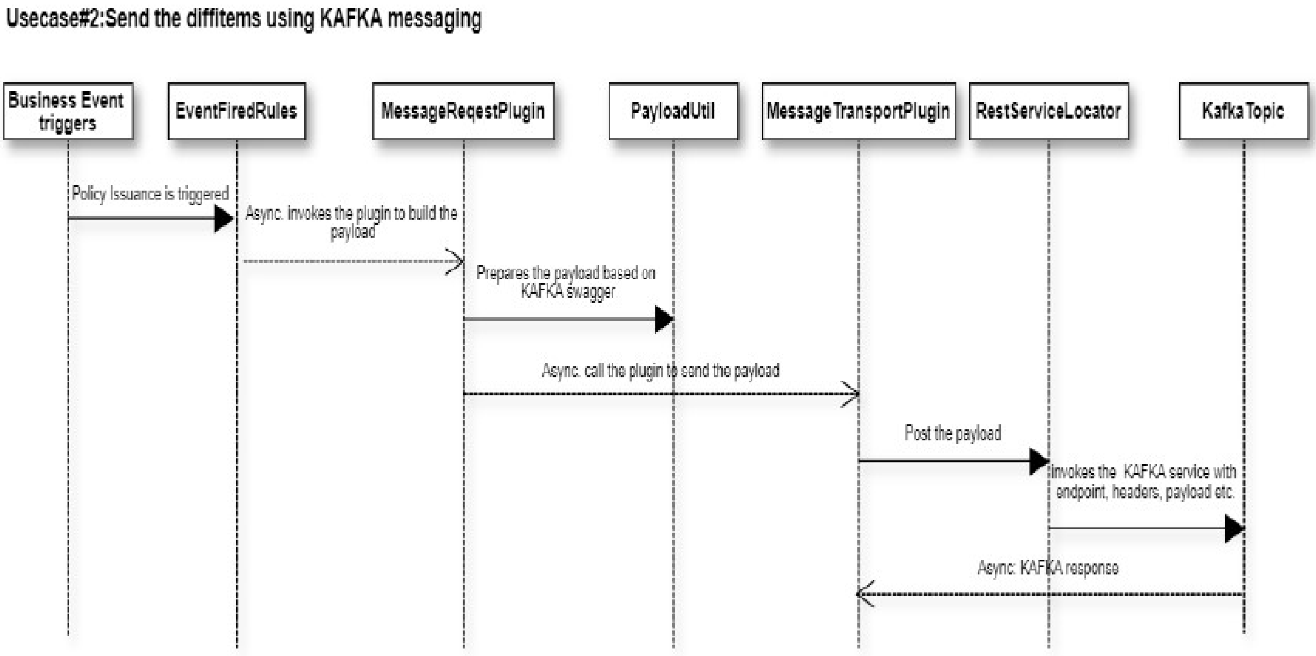

Usecase#2:Send the diffitems using KAFKA messaging
Business Event triggers
EventFiredRules
MessageReqestPlugin
PayloadUtil
MessageTransportPlugin
RestServiceLocator
KafkaTopic
Policy Issuance is triggered
Async: invokes the plugin to build the payload
Prepares the payload based on KAFKA swagger
Async: call the plugin to send the payload
Post the payload
invokes the KAFKA service with endpoint, headers, payload etc.
Async: KAFKA response

Figure 5. Processing DiffItems Using KAFKA Messaging

Table 4. DiffItems and KAFKA Components

Component Name	Description
Business events	Business events are fired due to external / internal interaction with workflows defined in PolicyCenter.
Event fired rules	Rules are triggered to generate the payload or call the external systems.
Message Request Plugin	Prepares the payload, Invoked by the EFR asynchronously.
Payload Util	Generates the payload for a given schema using the integration views.
Message Transport Plugin	Sends the payload to the endpoint.
REST service locator	Identifies the service based on the endpoint URL, client credentials, header params.
KAFKA Topic	A logical categorization of events/ messages store them in order.

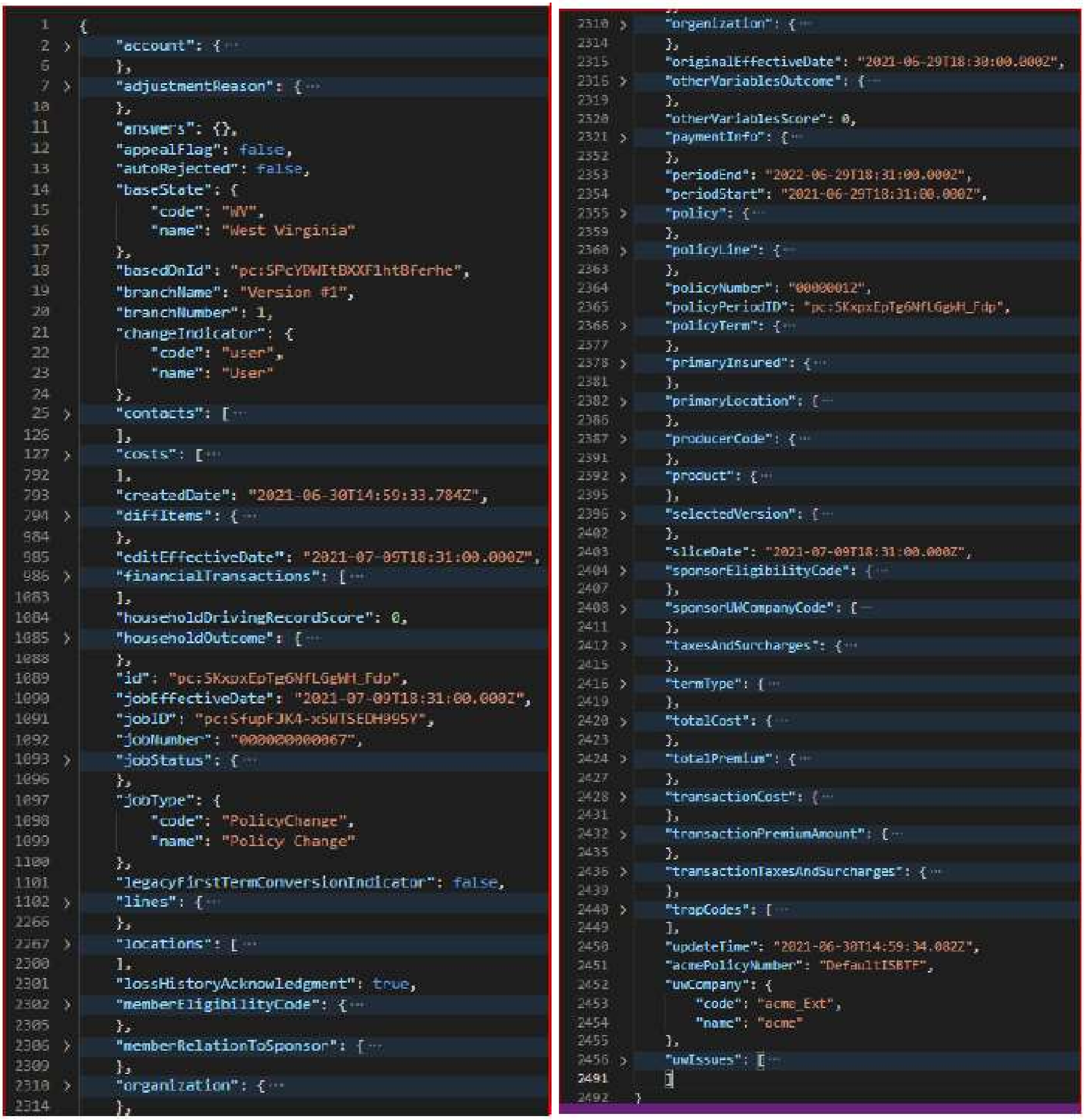
Current implementation supports following events, additional or custom events can be configured,

Table 5. Policy Transactions Events

Events	Description
CreatePeriod	Triggered whenever a policy is issued / bound
QuoteCloned	Triggered whenever quote is created / revised
DeclinePeriod	Triggered whenever a submission job is declined for new business
RenewPeriod	Triggered for Renewed policy
CancelPeriod	Triggered for cancellation events
ChangePeriod	Triggered for policy change transactions

2.7. Sample Payloads

2.7.1. Sample Difftree/DiffItems payload to downstream system



```
{
    "account": {⋯
    },
    "adjustmentReason": {⋯
    },
    "answers": {},
    "appealFlag": false,
    "autoRejected": false,
    "baseState": {
        "code": "WV",
        "name": "West Virginia"
    },
    "basedOnId": "pc:5PcYDWItBXXF1htBFerhe",
    "branchName": "Version #1",
    "branchNumber": 1,
    "changeIndicator": {
        "code": "user",
        "name": "User"
    },
    "contacts": [⋯
    ],
    "costs": [⋯
    ],
    "createdDate": "2021-06-30T14:59:33.784Z",
    "diffItems": {⋯
    },
    "editEffectiveDate": "2021-07-09T18:31:00.000Z",
    "financialTransactions": [⋯
    ],
    "householdDrivingRecordScore": 0,
    "householdOutcome": {⋯
    },
    "id": "pc:SKxpxEpTg6NfLGgWH_Fdp",
    "jobEffectiveDate": "2021-07-09T18:31:00.000Z",
    "jobID": "pc:SfupFJK4-xSWTSEDH995Y",
    "jobNumber": "000000000067",
    "jobStatus": {⋯
    },
    "jobType": {
        "code": "PolicyChange",
        "name": "Policy Change"
    },
    "legacyFirstTermConversionIndicator": false,
    "lines": {⋯
    },
    "locations": [⋯
    ],
    "lossHistoryAcknowledgment": true,
    "memberEligibilityCode": {⋯
    },
    "memberRelationToSponsor": {⋯
    },
    "organization": {⋯
    },
```

```
    "organization": {⋯
    },
    "originalEffectiveDate": "2021-06-29T18:30:00.000Z",
    "otherVariablesOutcome": {⋯
    },
    "otherVariablesScore": 0,
    "paymentInfo": {⋯
    },
    "periodEnd": "2022-06-29T18:31:00.000Z",
    "periodStart": "2021-06-29T18:31:00.000Z",
    "policy": {⋯
    },
    "policyLine": {⋯
    },
    "policyNumber": "00000012",
    "policyPeriodID": "pc:SKxpxEpTg6NfLGgWH_Fdp",
    "policyTerm": {⋯
    },
    "primaryInsured": {⋯
    },
    "primaryLocation": {⋯
    },
    "producerCode": {⋯
    },
    "product": {⋯
    },
    "selectedVersion": {⋯
    },
    "sliceDate": "2021-07-09T18:31:00.000Z",
    "sponsorEligibilityCode": {⋯
    },
    "sponsorUWCompanyCode": {⋯
    },
    "taxesAndSurcharges": {⋯
    },
    "termType": {⋯
    },
    "totalCost": {⋯
    },
    "totalPremium": {⋯
    },
    "transactionCost": {⋯
    },
    "transactionPremiumAmount": {⋯
    },
    "transactionTaxesAndSurcharges": {⋯
    },
    "trapCodes": [⋯
    ],
    "updateTime": "2021-06-30T14:59:34.082Z",
    "acmePolicyNumber": "DefaultISBTF",
    "uwCompany": {
        "code": "acme_Ext",
        "name": "acme"
    },
    "uwIssues": [⋯
    ]
}
```

Figure 6. Difftree/DiffItems payload to downstream system

3.RESULTS AND DISCUSSION

Performance is paramount important for REST APIs call in cloud deployments. The difftree and diffItems functionality is computationally expensive operation as it compares every field and every entity of given two PolicyPeriod graphs. It is always recommended to apply the appropriate filters to minimize the data elements to be compared, retrieved, and transported.

4.CONCLUSION

The Guidewire PolicyCenter (GW PC) Out-of-the-Box (OOTB) Difftree/DiffItems functionality has several limitations. It compares only the immediate predecessor of the current transaction's PolicyPeriod, making it less flexible for broader comparisons. Additionally, the functionality is not consumable by third-party systems, is limited to PolicyCenter users only, and is not exposed as a service for external integration. To overcome these limitations, a custom framework was developed to address these gaps and meet specific use cases. The solution serves two primary purposes. First, it provides required diffItems over a REST API service, enabling the comparison of two PolicyPeriods. This includes comparing the current PolicyPeriod with its based-on PolicyPeriod or with future-dated/backdated PolicyPeriods for a given slice date. The REST API also supports filtering the list of diffItems based on external system requirements. Second, theframework delivers all diffItems (without filters) as part of a KAFKA payload to downstream systems, ensuring seamless event-driven integration. The solution is highly flexible and can compare any two PolicyPeriods, whether within the same policy, job, or different policies under the same line of business (LOB), ensuring adaptability to various business scenarios with robust comparison capabilities.

REFERENCES

1. Comparing two jobs on the same policy. Guidewire documentation, Accessed12 Dec2024.Available:https://docs.guidewire.com/cloud/pc/202402/cloudapibf/cloudAPI/topics/124-PCsupport/00-comparing-jobs/c_comparing_two_jobs_on_the_same_policy.html
2. Shekar Madishetty. Integration Gateway: Re-imagining P&C Integrations for the Cloud. Available:https://developer.guidewire.com/integration-gateway-reimagining-p-and-c-integrations-for-the-cloud/
3. Simon Reading, and Mark Bolger. Simplifying Outbound Integrations in Insurance Suite Cloud - Part 2. Available: https://developer.guidewire.com/simplifying-outbound-integrations-in-insurancesuite-cloud-part-2/
4. Business flows: Framework APIs. Guidewire documentation, Accessed 12 December 2024. Available:https://docs.guidewire.com/cloud/pc/202402/cloudapibf/cloudAPI/topics/141-Framework/p_framework-business-flows.html
5. Cloud API Documentation. Guidewire documentation, Accessed 12 December 2024. Available: https://docs.guidewire.com/cloud/pc/202402/apiref/
6. Ryan Smith. Public API Reference for InsuranceSuite Available Now. Accessed 12 December 2024. Available : https://developer.guidewire.com/public-api-reference-for-insurancesuite/
7. Chris Vavra, Cloud Integration Framework: The Right Tools for the Job. Available:https://www.guidewire.com/resources/blog/technology/cloud-integration-framework-the-right-tools-for-the-job
8. Mark Bolger (Architect). Guidewire Application Events Service. Available:https://medium.com/guidewire-engineering-blog/guidewire-application-events-service-d1b0bee685b3

****