BPFragmentODRL: A Web Tool for Generating ODRL Policies for Business-Process Fragments

Abderrahmane Maaradji[1,*], Zakaria Maamar[1]

[1]University of Doha for Science and Technology, Doha, Qatar

Abstract

As organizations increasingly adopt distributed process execution models and cloud-based architectures, business processes are often decomposed into fragments that execute across different environments and systems. However, managing governance policies for these fragments remains a significant challenge, as existing approaches focus primarily on process-level policies without addressing the specific requirements of individual fragments. This paper presents BPFragmentODRL, an online tool for fragment policy generation in business processes that addresses this critical gap. The tool enables users to upload BPMN process models and automatically generate ODRL-compliant policies for process fragments using both template-based and LLM-based techniques. BPFragmentODRL supports multiple fragmentation strategies and provides policy consistency checking and metrics analysis. By enabling fine-grained governance at the fragment level, the tool facilitates compliance management, reduces policy conflicts, and supports the practical deployment of distributed business processes in modern enterprise environments.

Keywords

Business Process, BPMN, Fragmentation, ODRL, Policy Generation

1. Introduction

Business Process Management (BPM) has evolved significantly with the increasing need for process fragmentation across distributed environments [1]. While existing research has extensively covered BP modeling, execution, and optimization, the management of process fragments as governed assets remains largely unexplored. This gap becomes critical when BPs are decomposed into fragments that require coordinated governance to avoid conflicting situations and ensure compliance with organizational policies [2].

Our web tool, BPFragmentODRL, addresses this gap by generating necessary fragment policies in BPs. BPFragmentODRL leverages the Open Digital Rights Language (ODRL) [3] to express permissions, prohibitions, and obligations for process fragments, enabling fine-grained governance that goes beyond traditional process-level policy management. The significance of this tool to the BPM community lies in its ability to bridge BP fragmentation and policy governance, providing practitioners and researchers with a practical solution for exploring and implementing fragment governance strategies.

2. System Architecture

This section presents the technical architecture of BPFragmentODRL and its key components. The system integrates multiple policy generation techniques within a unified framework to support fragment governance.

Cooperative Information Systems - Early Research Achievement and Demos, 2025

The tool is available at: https://bpfragmentodrl-demo-v1.onrender.com/

*Corresponding author.

abderrahmane.maaradji@udst.edu.qa (A. Maaradji); zakaria.maamar@udst.edu.qa (Z. Maamar)

0000-0002-7969-2825 (A. Maaradji); 0000-0003-4462-8337 (Z. Maamar)

2.1. Core Architecture

Figure 1 illustrates BPFragmentODRL's architecture, showcasing the adoption of multiple policy generation techniques. The tool processes BPMN 2.0 process models through a fragmentation module that produces fragments with their dependencies, which are then fed into a centralized policy generation hub.

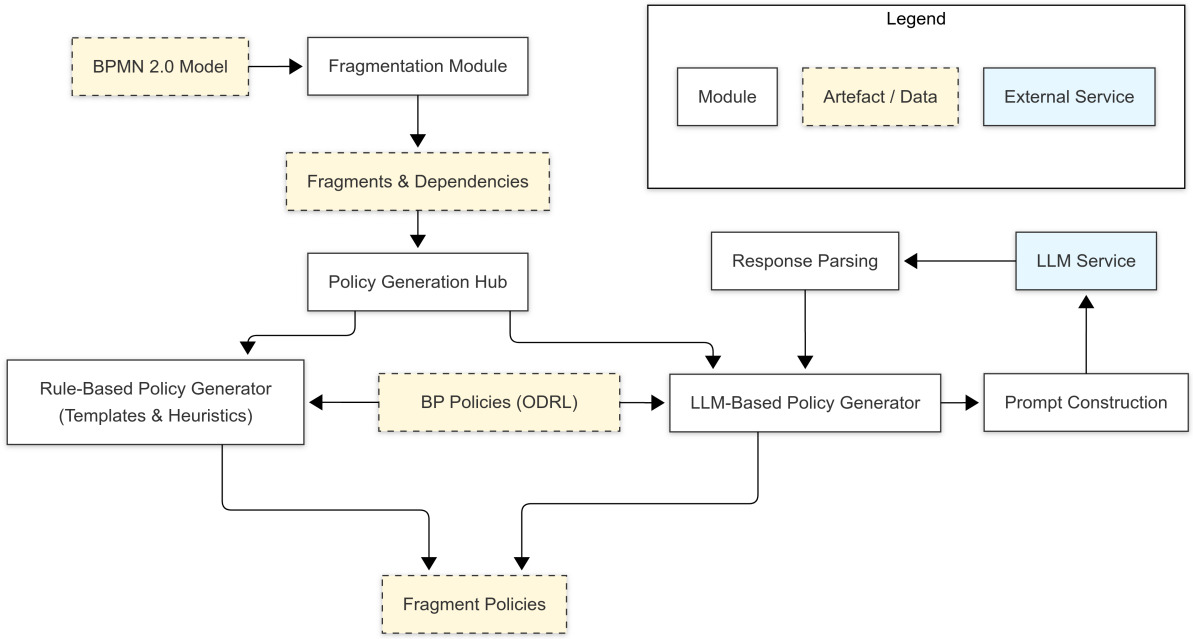

Figure 1: BPFragmentODRL architecture showing dual policy generation techniques

The architecture demonstrates two distinct policy generation techniques. The template-based policy generator employs predefined rules and heuristics to generate consistent, reliable policies based on established BPM best practices and regulatory frameworks, ensuring deterministic policy generation suitable for compliance-critical environments. The LLM-based policy generator uses Large Language Models through prompt engineering to analyze BP semantics and generate contextually appropriate policies, demonstrating the potential for AI-powered policy generation that adapts to unique process characteristics.

2.2. Web Interface and Multi-strategy Support

Figure 2 shows the user interface that makes advanced policy generation capabilities accessible to both researchers and practitioners. The interface provides clear options for selecting policy generation techniques, fragmentation strategies, and business process policy management while maintaining professional usability standards.

BPFragmentODRL supports three distinct fragmentation strategies: activity-based fragmentation creates fine-grained fragments corresponding to individual activities, gateway-based fragmentation uses BPMN gateway structures as natural fragment boundaries, and hybrid fragmentation combines elements of both techniques to identify optimal fragment boundaries that balance granularity with manageability.

3. Tool Demonstration and Capabilities

This section demonstrates the capabilities of BPFragmentODRL through its policy generation workflow and analysis features. The tool provides end-to-end support for fragment policy management from BPMN model upload to policy export and integration.

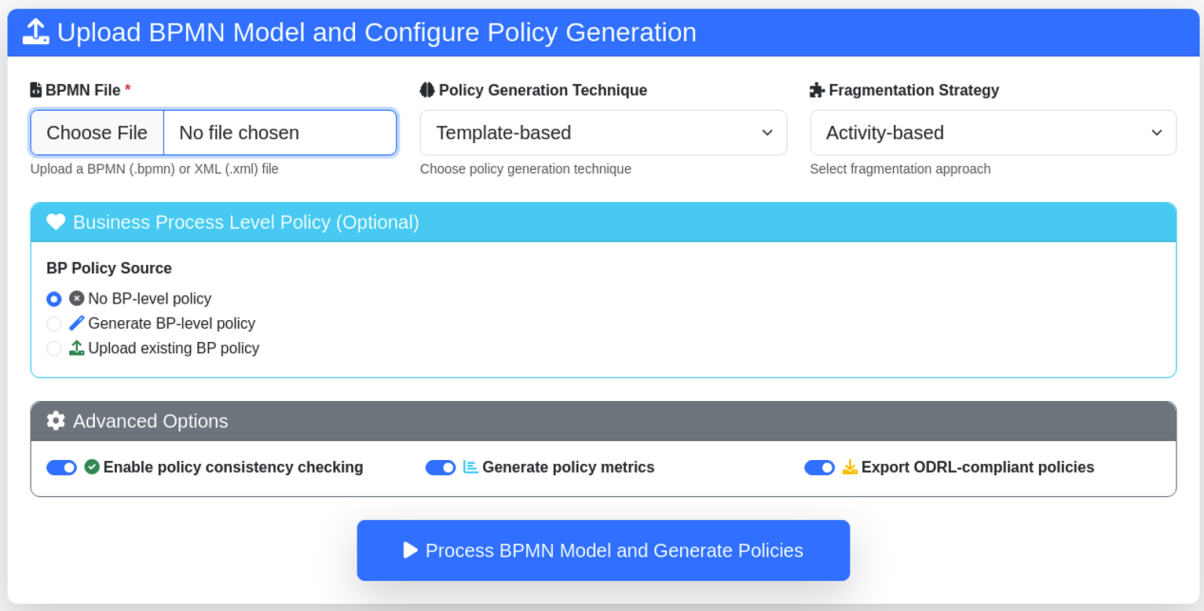

Figure 2: BPFragmentODRL user interface showing configuration options

3.1. Policy Generation Workflow

The demonstration workflow showcases the tool's capabilities through a systematic process: users upload BP process models in BPMN or XML formats through an intuitive drag-and-drop interface with real-time validation, choose between template-based and LLM-based policy generation techniques, select the appropriate fragmentation strategy, configure business process policies through three options (no BP policy, generate BP policy using domain-specific templates, or upload existing BP policies), process models to generate ODRL-compliant policies with hierarchical consistency, and review generated policies through interactive visualizations and metrics analysis.

3.2. LLM Prompt Engineering Innovation

A key technical innovation in BPFragmentODRL is the prompt engineering used for LLM-based policy generation. Listing 1 shows the core prompt structure that enables contextually appropriate policy generation while maintaining consistency with business process policies:

Listing 1: LLM prompt structure for fragment policy generation with BP policy integration

```
Generate ODRL policies for business process fragment {fragment_id}
containing the following activities: {activity_names}

Business Process Policy Context:
The fragment policies must be consistent with the following
business process policy: {bp_policy_json}

Fragment Context:
- Fragment Type: {fragmentation_strategy}
- Dependencies: {fragment_dependencies}
- Business Domain: {domain_context}

Generate policies that specify:
1. Permissions: What actions are allowed on fragment activities
2. Prohibitions: What actions are forbidden
3. Obligations: What actions must be performed

Requirements:
- Ensure ODRL compliance and proper JSON structure
```

```
- Maintain consistency with BP-level constraints
- Consider fragment dependencies and relationships
- Adapt policies to business domain context

Output format: Valid ODRL JSON with clear rule categorization
```

This prompt structure demonstrates how business process policies are integrated into fragment policy generation, ensuring hierarchical consistency while enabling context-specific adaptations. The prompt includes domain context, fragment relationships, and explicit requirements for ODRL compliance and hierarchical consistency with BP-level constraints.

3.3. Case Study: Credit Application Process

We demonstrate the tool's capabilities using a credit application business process that includes multiple decision points, parallel processing paths, and exception handling mechanisms. The process involves application completeness verification, initial checks, amount-based routing, and approval workflows.

The case study results demonstrate the effectiveness of both policy generation techniques. Template-based technique achieves processing times of 1-2 seconds for typical processes, generates 11 rules (6 permissions, 2 prohibitions, 3 obligations), maintains 100% conflict-free policy sets with deterministic results across multiple runs, and provides automatic inheritance of BP-level constraints. LLM-based technique requires 3-5 seconds with external API calls , generates contextually adapted rule sets with natural language descriptions, adapts to unique process characteristics and business domains, includes graceful degradation to template-based generation when LLM unavailable, and provides integration of BP policy context through prompt engineering.

3.4. Policy Analysis and Validation Features

The tool provides analysis capabilities that distinguish it from existing BPM tools. Consistency checking validates policies within and across fragments to identify potential conflicts, inconsistencies, or gaps in policy coverage, including BP-to-fragment policy alignment. Metrics analysis generates quantitative assessments including rule distribution, complexity measures, coverage analysis, and hierarchical policy relationships. ODRL compliance ensures that all generated policies conform to ODRL standards for interoperability and standardization with automated validation. Interactive visualization provides detailed exploration of fragment structures, policy relationships, dependency analysis, and BP policy inheritance patterns.

4. Tool Maturity and Access

This section presents the implementation details, performance characteristics, and accessibility of BPFragmentODRL. The tool is designed for production deployment with features for research and educational use.

4.1. Implementation and Performance

BPFragmentODRL is implemented as a Web application using Flask (Python) for robust backend processing and Bootstrap-based responsive frontend for cross-platform accessibility. The tool is publicly deployed at https://bpfragmentodrl-demo-v1.onrender.com/, providing immediate access for evaluation without installation requirements. The source code is available at https://github.com/amaaradji/bpfragmentodrl_demo. The modular architecture supports extensibility and customization while maintaining performance standards suitable for research and educational deployment.

Performance evaluation demonstrates response times of 1-3 seconds for template-based generation and 5-15 seconds for LLM-based generation, memory efficiency of <512MB for complex BPs with multiple fragments, support for multiple simultaneous policy generation sessions with load balancing,

cloud deployment with automatic scaling based on demand, and accuracy of 95%+ consistency for template-based policies and 85-92% contextual appropriateness for LLM-based policies.

4.2. Accessibility and Demonstration

A video demonstration is available at https://tinyurl.com/mrynnzkv, showcasing the complete policy generation workflow through the credit application case study(3), explaining each step and highlighting key innovative features. The live tool provides immediate hands-on access for reviewers and practitioners with sample BPMN. No registration is required, enabling immediate evaluation of all features, including both policy generation techniques, multiple fragmentation strategies, and business process policy integration.

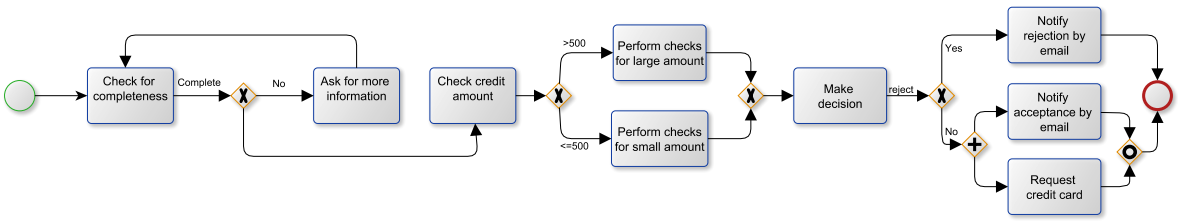

Figure 3: Credit application business process used for case study demonstration

5. Conclusion and Future Directions

BPFragmentODRL represents a significant advancement in the governance of fragmented BPs, providing the BPM community with a practical solution for implementing fragment policy management. The combination of template-based reliability and LLM-based flexibility offers researchers and practitioners multiple techniques for addressing diverse organizational requirements while maintaining hierarchical policy consistency. The tool's public availability, feature set, and open architecture facilitate community contributions and collaborative enhancement. Future development directions include enhanced LLM integration with domain-specific models, real-time policy adaptation capabilities, and cross-organizational policy coordination frameworks.

Declaration on Generative AI

During the preparation of this work, the authors used standard proofreading services (e.g., Overleaf/-Grammarly) solely for grammar and spelling checks. After using these services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] M. Weske, Business Process Management: Concepts, Languages, Architectures, 3rd ed., Springer, 2019.

[2] S. Kallel, Z. Maamar, M. Sellami, N. Faci, A. Ben Arab, W. Gaaloul, T. Baker, Restriction-based Fragmentation of Business Processes over the Cloud, Concurrency and Computation: Practice and Experience 37 (2021).

[3] W3C, ODRL information model 2.2, W3C Recommendation (2018). URL: https://www.w3.org/TR/odrl-model/.