Elementum, Inc.

Maximize Your Infor WFM Investment

The Role of Groovy Scripting in Infor Workforce Management (WFM) Implementations

A Whitepaper Discussing Customization and Automation in Workforce Management

Date: April 26, 2025

Written by:

Konrad Wagner
(c) (647) 716-4701
(o) (877) 294-1524 ext 701
(e) konrad.wagner@elementumconsulting.com

Elementum, Inc.
Workforce Management Consulting

For more information, visit www.elementumconsulting.com

Contents

Abstract

Infor Workforce Management (WFM) is a comprehensive enterprise solution designed to optimize labor management, scheduling, time and attendance, and compliance with global labor regulations. The integration of Groovy scripting within Infor WFM's custom scripting framework empowers organizations to create tailored rules, conditions, and entitlements that align with their unique operational needs. This whitepaper explores the transformative role of Groovy scripting in WFM implementations, emphasizing its ability to enable rapid customization without requiring new deployments or server access, accelerate development through Groovy's concise syntax, and provide precise control over data manipulation to meet customer requirements. Through detailed analysis, practical examples, and best practices, this paper highlights how Groovy scripting enhances flexibility, efficiency, and compliance in workforce management.

1 Introduction

In today's fast-paced business landscape, organizations face increasing pressure to optimize workforce operations while ensuring compliance with complex labor laws and meeting employee expectations. Infor Workforce Management (WFM) is a leading software solution that addresses these challenges through its robust modules for time and attendance, demand-driven scheduling, absence management, and labor forecasting. However, the diverse needs of industries such as healthcare, retail, and manufacturing often require customizations beyond standard functionalities.

What sets Infor WFM apart from other workforce management solutions, such as UKG (Kronos), Workday, and ADP, is its unique custom scripting framework powered by the Groovy scripting language. Unlike competitors, which rely on configuration tools, APIs, or proprietary languages, Infor WFM's integration of Groovy enables organizations to create highly tailored rules, conditions, and entitlements directly within the application. This unparalleled flexibility empowers businesses to address complex and unique requirements with ease, providing a significant competitive edge in adapting to dynamic workforce demands.

Groovy, a dynamic and developer-friendly language running on the Java Virtual Machine (JVM), combines simplicity with robust functionality, making it an ideal tool for customizing Infor WFM. This whitepaper explores the transformative role of Groovy scripting in enhancing the flexibility, automation, and efficiency of Infor WFM implementations. Through detailed analysis, practical examples, and best practices, it highlights how Groovy scripting enables organizations to achieve operational excellence in workforce management, setting Infor WFM apart in a crowded market.

By exploring the technical capabilities, practical applications, and best practices of Groovy scripting, this paper aims to provide WFM administrators, developers, and business leaders with insights into maximizing the value of their Infor WFM investment

2 Overview of Infor WFM and Its Customization Needs

Infor WFM is a modular, cloud-based solution designed to streamline workforce management processes across various industries. Its key functionalities include:

- Time and Attendance: Automates time tracking and payroll calculations to ensure accuracy and compliance.
- Workforce Scheduling: Supports demand-driven and shift-based scheduling, adhering to labor regulations.
- Absence Management: Manages leave requests, accruals, and compliance with organizational policies.
- Labor Forecasting: Leverages machine learning and historical data to predict staffing needs.

While these features provide a strong foundation, organizations often require customizations to address unique challenges, such as industry-specific compliance requirements, complex pay structures, or specialized scheduling rules. For instance, a healthcare provider may need to enforce nurse-to-patient ratios, while a retail chain might require custom shift-bidding logic. Groovy scripting, integrated into Infor WFM's custom scripting framework, enables these tailored solutions, ensuring the system aligns with specific business needs.

3 Understanding Groovy Scripting in Infor WFM

3.1 What is Groovy?

Groovy is an object-oriented scripting language that operates on the JVM, combining the power of Java with a concise, developer-friendly syntax inspired by languages like Python and Ruby. Its key features include:

- Dynamic Typing: Simplifies coding by allowing flexible variable declarations.
- Java Interoperability: Seamlessly integrates with Java libraries and Infor WFM's public Java API.
- Concise Syntax: Reduces code complexity, enabling faster development.

These attributes make Groovy an ideal choice for creating custom scripts within Infor WFM, balancing ease of use with robust functionality.

3.2 What customizations are possible with Groovy?

Customizations are possible with Infor provided libraries to extend many parts of the core (and extended) application including, but not limited to:

- Pay Rules and Conditions (Rule Engine)
- Data Events
- Entitlements (Employee Benefits)
- Payroll Export
- Imports
- Attendance Control
- Alerts
- Schedule Compliance
- Time Off Planner
- MVS (Multi View Scheduler)
- … and more

Understanding the appropriate modules and the application's underlying design is essential for developing extension scripts that fulfill requirements while maintaining data integrity, system performance, and reliability.

3.3 The Custom Scripting Framework

Infor WFM's custom scripting framework leverages Groovy to enable developers to create, test, and deploy custom rules, conditions, and entitlements. Key components include:

- Coding Environment: Scripts can be written and debugged within the WFM application or in external IDEs with Groovy support, such as IntelliJ IDEA or Eclipse, offering features like syntax checking and auto-complete.
- Sandbox Testing: A sandbox environment allows developers to test scripts without impacting production data, ensuring reliability.
- API Access: The public Java API, documented in the WFM Javadoc, provides access to WFM services and data, such as employee records and schedules.
- Entitlement Endpoints: Supports customizations for actions like shift assignments, compliance checks, and ratio calculations.

This framework ensures that customizations are seamlessly integrated into the WFM ecosystem, enhancing system functionality without compromising stability.

4 Practical Applications of Groovy Scripting

Groovy scripting in Infor WFM enables a wide range of customizations. Below are key applications, illustrated with simplified examples.

4.1 No Need for New Deployments or Server Access

Traditional software customization often involves updating the application's codebase, requiring server access, compilation, and deployment of new versions—a process that can be time-consuming and disruptive. Infor WFM's custom scripting framework eliminates these hurdles by allowing developers to create and deploy scripts directly within the application. This approach offers several advantages:

- **Rapid Implementation**: Customizations can be developed, tested, and deployed in hours or days, compared to weeks for traditional methods, enabling organizations to respond swiftly to business changes.
- **Zero Downtime**: Scripts are deployed through the WFM interface, avoiding system downtime and ensuring continuous operation.
- **Cost Efficiency**: By reducing the need for server access and specialized IT resources, organizations can lower implementation costs.
- **Accessibility**: Non-administrative users with appropriate permissions can manage scripts, democratizing customization and reducing reliance on external vendors.

This streamlined process empowers project implementors to deliver tailored solutions without the complexities of traditional software updates.

4.2 Speed of Developing New Customs

Groovy's design prioritizes developer productivity, enabling rapid creation of custom scripts. Its concise syntax reduces the lines of code needed, while dynamic features allow for flexible and iterative development. The custom scripting framework further accelerates this process through:

- IDE Integration: Support for IDEs with Groovy plugins provides advanced features like code completion, syntax highlighting, and real-time error detection, speeding up development.
- In-App Development: For simpler scripts, the WFM application's built-in editor offers immediate feedback, enabling quick iterations.
- Reusable Components: Groovy's support for modular code allows developers to create libraries of reusable functions, minimizing redundant coding efforts.
- Rapid Testing: The sandbox environment facilitates quick testing and debugging, reducing the time from concept to deployment.

These features enable implementors to develop new customs in a fraction of the time required for traditional programming, ensuring that WFM remains agile and responsive to organizational needs.

4.3 Ultimate Control Over Data Manipulation

Groovy scripts leverage Infor WFM's public Java API to provide comprehensive access to system data and services, enabling precise manipulation to meet customer requirements. This control manifests in several ways:

- **Data Access**: Scripts can retrieve and update employee records, time entries, schedules, and other data, allowing for highly customized logic.
- **Complex Logic Implementation**: Developers can create intricate rules for pay calculations, scheduling algorithms, and compliance checks tailored to specific business scenarios.
- **Integration Capabilities**: Scripts can interface with external systems, enabling data exchange and enhancing WFM's interoperability.
- **Flexibility**: The ability to manipulate data at a granular level ensures that organizations can address unique challenges, such as regional labor laws or industry-specific policies.

This level of control empowers organizations to align WFM processes with their strategic goals, enhancing efficiency, compliance, and employee satisfaction.

5 Custom Script Examples

To demonstrate the versatility of Groovy scripting in Infor WFM, the following examples illustrate its application in real-world scenarios.

5.1 Custom Payrule Logic

Organizations often need to send additional details to their payroll system within the Work Detail information exported.. For instance, a company may need to store the start of the pay period to which the detail belongs in one of the Work Detail UDF values, for later transmission during the payroll export process. The example below determines a Sunday's date for ea

```
// determine the current day of the week
def workDate = workSummaryPubl.getWorkDate()
def currentDay = workDate[Calendar.DAY_OF_WEEK]    // Groovy property access
def daysToAdd = (1 - currentDay + 7) % 7
def sundayDate = workDate + daysToAdd
Udf10Day = sundayDate.format("YYYYMMdd")
for (int i = 0; i < wdList.size(); i++) {
        workDetailPubl = (WorkDetailPubl) wdList.get(i);
        workDetailPubl.setUdf10(Udf10Day)
}
```

The script above calculates the date of Sunday in the week that the detail occurred. Then, this date is added to UDF 10 in the Work Detail. The Payroll Export scripts would need to be updated to take advantage of the date stored and export it appropriately.

5.2 Entitlement Customizations

Groovy supports the creation of custom Unit Dates. This is useful when Entitlement logic needs to be applied on dates other than January 1st or a specific number of days into the year. Some clients require entitlements to be applied every July 1st of each year. Here is a snipped of code that could accomplish this task.

```
if (applyOnValue == JULY_FIRST) {
        workDateCal.set(Calendar.MONTH, 6); //set Month to July (months 0..11)
        workDateCal.set(Calendar.DAY_OF_MONTH, 1);

        return workDateCal.getTime();
} else {
        return sdf.parse("01013000");
}
```

This script compares the current date to a fixed date (July 1), otherwise returns the default of a "forever" date (1/1/3000).

5.3 Custom Data Events

Groovy scripts can automate actions on the WFM timesheet. For instance, some clients wish to prevent overrides being made by employees to their own timesheets. A simple check could prevent the user from being able to make edits.

```
if (ovr.getEmployeeId()== ovrCtxPublService.getOverrideCreatorEmployeeId(context, ovr)){
        ovr.setStatus(OverrideStatus.ERROR);
}
```

This script checks to see if the user creating the edit is the same as the timesheet's employee. If so, it sets the Override to status ERROR.

6 Best Practices for Groovy Scripting

To ensure effective and sustainable use of Groovy scripting in Infor Workforce Management (WFM), developers must adopt a disciplined approach that balances functionality, performance, and maintainability. Groovy's flexibility enables powerful customizations, but without proper practices, scripts can lead to performance issues, security risks, or maintenance challenges. Below is an expanded set of best practices designed to guide developers in creating robust, efficient, and secure Groovy scripts for Infor WFM implementations.

- **Leverage Documentation**: Consult the WFM Javadoc for detailed API information to ensure accurate script development.
- **Write Efficient Code**: Optimize scripts to handle large datasets, avoiding excessive API calls or complex loops.
- **Test Extensively**: Use the sandbox environment to simulate real-world scenarios and validate script functionality.
- **Document Thoroughly**: Include comments and maintain version control to facilitate maintenance and collaboration.
- **Secure Scripts**: Restrict access to authorized users and audit scripts to prevent data exposure.
- **Plan for Upgrades**: Test scripts against WFM updates to ensure compatibility.
- **Collaborate with Stakeholders**: Engage business analysts, WFM administrators, and end-users during script development to ensure alignment with requirements.

7 Challenges and Considerations

While Groovy scripting offers significant benefits, it also presents challenges:

- Learning Curve: Developers even familiar with Groovy may require training to master WFM's API, as scripts are typically simple in syntax, yet complex in design around the application.

- Performance Risks: Inefficient scripts can impact system performance, necessitating optimization.
- Maintenance Overhead: Custom scripts require ongoing updates to remain compatible with WFM upgrades, especially with monthly maintenance releases in Multi-tenant Cloud (MTC)
- Security Concerns: While the scripting libraries are controlled by Infor's WFM team, Scripts should be carefully audited to prevent unauthorized data access.

Mitigating these challenges involves investing in training, rigorous testing, and collaboration with experienced WFM consultants.

8 Case Studies

8.1 Healthcare: Custom Bonus Incentives

A large hospital network implemented some custom Groovy scripts to generate a bonus structure based on attendance and average hours worked. The scripts were able to help reduce bonus payout errors by over 90%.

8.2 Retail: Custom Unit Date

A national retail chain used Groovy to create a new Unit Date to allow for entitlements to fire annually on the first day of the hire month of the employee. This aligned with their long standing HR policies and helped with the transition to the WFM system.

8.3 Industrial: Attendance Control Extensions

A large industrial client used Groovy scripting to extend logic around when attendance events would fire, including the addition of point reductions for perfect attendance and the accommodation of medical documentation. The solution helped automate much of the work previously done by the payroll department, saving hundreds of hours per year in corrections.

8.4 Manufacturing: Overtime Logic Extensions

A growing manufacturing client had complex requirements around overtime and application of overtime codes to specific jobs and departments. The custom rule implemented had corrected a process which previously required dozens of hours of manual corrections per pay period, and reduced payroll errors by 15%.

9 Conclusion

Groovy scripting is a cornerstone of Infor WFM's customization capabilities, offering unmatched flexibility and control over workforce management processes. Unlike other leading WFM solutions, such as UKG (Kronos), Workday, and ADP, which utilize configuration-based tools, APIs, or proprietary languages like Workday's XO/YO, Infor WFM's use of Groovy scripting allows for rapid development and deployment of custom rules and conditions directly within the application. This unique feature enables organizations to efficiently meet specific business requirements, ensuring agility in responding to evolving workforce needs.

By leveraging Groovy's accessible syntax, seamless Java integration, and Infor's robust custom scripting framework, businesses can automate complex rules, enhance compliance, and improve operational efficiency. While challenges such as performance optimization and maintenance require careful consideration, adherence to best practices and strategic partnerships with Infor consultants can ensure successful implementations. As workforce management continues to evolve, Groovy scripting positions Infor WFM as a leader, empowering organizations to stay agile and competitive in a dynamic labor landscape.

10 References

[1] Infor, Custom scripting framework, https://docs.infor.com

[4] Infor, Workforce Management | Enterprise WFM software, https://www.infor.com, published June 21, 2016,

[5] Surety Systems, Everything You Need to Know About Infor Workforce Management (WFM), https://www.suretysystems.com, published September 18, 2022

[6] Infor, Workforce Scheduling | Employee scheduling software, https://www.infor.com

[7] Infor, Workforce Management | Enterprise WFM software, https://www.infor.com

[8] Axsium Group, Infor WFM Implementation Partner, https://axsiumgroup.com, published September 25, 2020

[9] Inovium, Human Capital and Workforce Management Consulting, https://www.inovium.com, published April 25, 2023

11 Disclosures

This whitepaper was developed with the assistance of Grok, an artificial intelligence tool created by xAI. Grok was used to generate some initial drafts, structure content, and provide technical insights related to Groovy scripting and Infor Workforce Management. The content has been thoroughly reviewed, edited, and enhanced by the author to ensure accuracy, relevance, and alignment with industry standards. All examples, case studies, and additional sections reflect the author's contributions and expertise. The use of AI was intended to streamline the drafting process while maintaining the integrity and originality of the final work