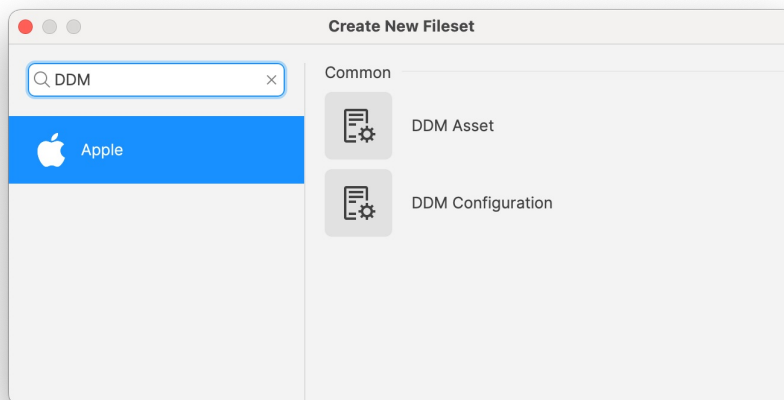


Apple Device Management - DDM Configurations

What

Device Declarative Management (DDM) is Apple's modern approach to device management, introduced to enhance and eventually replace traditional Mobile Device Management (MDM) protocols. With FileWave 15.5, support for DDM configurations is now available, allowing administrators to manage Apple devices more efficiently. DDM shifts some management logic to the device, enabling it to proactively apply configurations and report status updates, reducing server load and improving scalability.



When/Why

Implement DDM Configurations in FileWave 15.5 when you aim to:

- Modernize Device Management: Adopt Apple's latest device management methodology to stay current with industry advancements.
- Increase Efficiency: Allow devices to autonomously manage configurations, reducing reliance on constant server communication.
- Enhance Scalability: Improve performance when managing large fleets, as devices handle more processing locally.
- Improve Responsiveness: Devices can apply configurations and respond to changes more quickly without waiting for server commands.

This is particularly useful for organizations managing numerous devices, seeking to optimize performance and reduce overhead.

Note: Assets and DDM configurations are supported on devices running: iOS 15 and later, iPadOS 15 and later, macOS 12 Monterey and later

How

To create and use Configurations in FileWave 15.5:

1. Create a Configuration:
 - Create a new Fileset and pick DDM Configuration from the Apple section.
2. Configure the Configuration:
 - Pick the Configuration you want to use. In FileWave 15.5.0 we started with Account: CalDAV, Passcode Settings, Screen Sharing Connections, Screen Sharing Host Settings, and Software Update Settings as the first configurations.
3. Reference a DDM Asset in DDM Configurations:
 - Some Configurations can reference [Apple Device Management - DDM Assets](#). An example is the Account: CalDAV Configuration which can be fed credentials from a DDM Asset.
4. Automatic Dependency Handling:
 - When a configuration references an Asset, FileWave automatically manages the Asset as a dependency.
 - Deploying the configuration will also deploy the associated Asset to the target devices.
5. Deploy to Devices:
 - Assign the configurations to your devices or device groups.
 - Monitor the deployment to ensure that devices receive both the configuration and the associated Assets.

Related Content

- [Apple's Declarative Device Management Documentation](#)

Digging Deeper

Declarative Device Management (DDM) represents a significant evolution in Apple's device management strategy:

- **Device-Centric Management:** Devices receive declarations of desired states and autonomously ensure compliance, reducing the need for continuous server commands.
- **Enhanced Performance:** Offloading processing to devices improves performance and scalability, especially in large environments.
- **Improved Reliability:** Devices can enforce configurations even when temporarily disconnected from the management server.

Key Benefits:

- **Reduced Server Load:** Servers are less burdened with managing individual device states, as devices handle more tasks independently.
- **Faster Configuration Application:** Devices can apply changes immediately upon receiving declarations, without waiting for additional instructions.
- **Proactive Compliance:** Devices continuously ensure they meet the declared state, self-correcting if configurations are altered or removed.

By embracing DDM configurations in FileWave 15.5, organizations can achieve a more efficient, scalable, and responsive device management system that meets the demands of modern IT environments.

Conflicting Payloads:

Apple have not provided any alternate information, from MDM Payloads, regarding the experience if two DDM configurations are applied to control the same feature, but with differing settings. Please consider the following:

- Where MDM Payloads are concerned, Apple suggest the experience is undefined.
 - [Plan your configuration profiles for Apple devices](#)
- Apple used to have an additional clause, suggesting that where restrictions payloads conflicted, the more restrictive setting would win (but this detail was removed from their documentation).
- It would be reasonable to assume that the same conditions apply to DDM.

For what it is worth, testing the conflict between MDM and DDM for a restriction provided the following result:

- Set differing macOS Software Update defer durations in both MDM and DDM payloads.
- Associate both.
- In each test, the most restrictive (greatest duration of days) appeared to always be applied.
- It did not matter if DDM or MDM was the more restrictive.

i In would be sensible to avoid conflicts where possible, rather than rely on a tested experience.

🔄Revision #11

★Created 7 October 2024 13:36:32 by Josh Levitsky

✍Updated 6 January 2025 13:33:31 by Sean Holden