

MODAPTO [101091996]: Modular Manufacturing and Distributed Control via Interoperable Digital Twins



12.2.1 Registering Smart Service Manifest in the service Catalogue

2025-08-11



The process of registering a Smart Service Manifest in the MODAPTO Service Catalogue follows a structured workflow designed to ensure consistency, quality, and completeness. This registration process transforms a service definition into an active Catalogue entry that can be discovered and utilized by other components within the manufacturing ecosystem.

Pre-Registration Preparation begins with ensuring that the service manifest is complete and properly formatted. Service providers must gather all required information including service identification details, comprehensive descriptions, technical specifications, and metadata. For Internal services, Docker images must be built and pushed to accessible repositories. For External services, endpoints must be tested and verified as accessible from the MODAPTO environment. FMU services require packaging according to FMI standards with all dependencies included. This preparation phase often involves using validation tools to check manifest syntax and completeness before attempting registration.

Accessing the Registration Interface requires appropriate authentication and authorization within the MODAPTO system. Users navigate to the Service Catalogue UI at the designated endpoint (typically `http://[host]:8080/Catalogue/`) and authenticate using their MODAPTO credentials. The system verifies that the user has the necessary permissions to register services, which may vary based on organizational roles and policies. The registration interface presents a comprehensive form divided into common fields applicable to all services and type-specific fields that appear based on the selected service type.

Completing Common Registration Fields involves entering fundamental service information that applies regardless of service type. The service name must be unique within the Catalogue and follow naming conventions (alphanumeric with hyphens or underscores, no spaces). The logo upload accepts standard image formats (PNG, JPG) with recommended dimensions of 200x200 pixels for optimal display. Short descriptions are limited to 280 characters and should convey the service's primary value proposition. Long descriptions support rich text formatting, enabling comprehensive documentation with sections, lists, code examples, and embedded images.

Defining Input and Output Specifications using AAS SubmodelElement format requires careful attention to data structure and typing. Each input parameter is defined as a separate JSON object with required fields: `modelType` (typically "Property" for simple values), `idShort` (unique parameter name), and `valueType` (`xs:string`, etc.). Optional fields include constraints, units, and semantic references. Outputs are defined as JSON arrays of SubmodelElements, accommodating services that return multiple values. The interface provides templates and validation to ensure correct JSON syntax.

Type-Specific Registration Requirements vary significantly based on the selected service type. For FMU services, users upload ZIP files containing the complete FMU package, which the system validates for FMI compliance. External services require endpoint URL (must be HTTPS for production), HTTP method selection, custom headers (often for authentication), and optional request payload templates. Internal services need Docker repository information (registry URL, image name, tag), exposed port configuration, and relative endpoint paths. Orchestrated services require Microservice ID and Operation ID obtained from the MODAPTO Orchestrator's test interface.



Validation and Error Handling occurs automatically as users complete the registration form. Client-side validation provides immediate feedback for format errors, missing required fields, or invalid values. Upon submission, server-side validation performs deeper checks including manifest schema compliance, service name uniqueness, endpoint accessibility (for External services), Docker image availability (for Internal services), and FMU package integrity. Validation errors are displayed with specific messages indicating the nature of the problem and suggestions for resolution.

Successful Registration Outcomes result in the service being immediately available in the Catalogue. The system assigns a unique identifier to the service, stores the manifest in the Catalogue database, indexes service metadata for search functionality, and generates a service detail page. For Internal services, the registration triggers deployment processes that provision containers and configure networking. Confirmation messages provide the service URL and any additional steps required for service activation.

References

1. MODAPTO Consortium. (2024). "Service Catalogue User Manual." MODAPTO Project Documentation.
2. Docker Inc. (2023). "Docker Registry HTTP API V2." Available at: <https://docs.docker.com/registry/spec/api/>
3. FMI Development Group. (2023). "FMI 3.0 Implementation Guide." Modelica Association.
4. RESTful API Design. (2023). "Best Practices for API Design." Available at: <https://restfulapi.net/>
5. JSON Schema Validation. (2023). "Understanding JSON Schema." Available at: <https://json-schema.org/understanding-json-schema/>