

Phase 1, System Management Delhi Release

Latest Requirements Scoping Document: [EdgeX Foundry Micro Service System Management-v5-awe-comments.pdf](#)

Current Design Doc:

[System Management Design.odt](#)

[System Management Design-v2.odt](#)

[System Management Design-v3.odt](#)

[System Management Design-v4.odt](#)

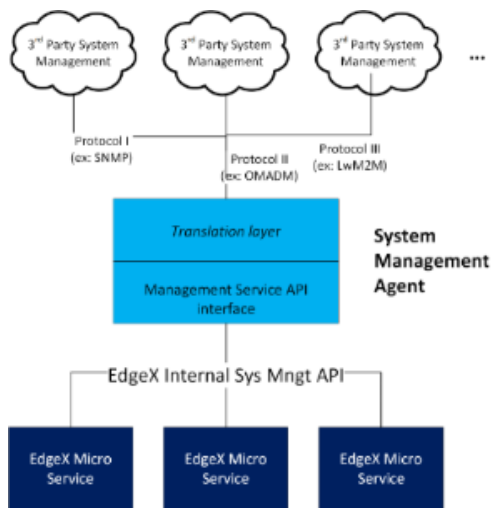
[System Management Design-v5.odt](#)

[System Management Design-v6.odt](#)

[System Management Design-v7.odt](#)

[System Management Design-v8-Delhi-final.pdf](#)

System Management (SM) Agent

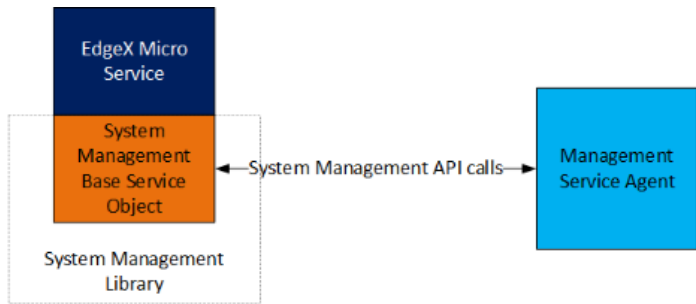


Functions and eventual public API Offered by the SM Agent

- Stop all EdgeX microservices
- Start all EdgeX microservices
- Restart all EdgeX microservices
- Stop an EdgeX microservice by name
- Start an EdgeX microservice by name
- Restart an EdgeX microservice by name
- Ping an EdgeX microservice (check that the service is still up)
- Get the configuration settings (aka properties) for an EdgeX microservice by name
- Get the configuration setting (aka property) for an EdgeX microservice (by name) by configuration setting name (aka key)
- Set the configuration setting (aka property) for a writable (versus read-only) property for an EdgeX microservice (by name). Example: the port of a service is read-only whereas the log level may be updated (writable).
 - Removed from scope due to shortened release time line and complexities of persisting info in Consul or file system.
- Get the current Admin status (locked or unlocked) of an EdgeX microservice (only valid for Device Services?)
 - Removed from scope due to shortened release time
- Register a client for change to a configuration setting to a named EdgeX microservice
- Register a client for change of the operating state between enabled and disabled states to a named EdgeX microservice
- Register a client for change of the admin state between locked and unlocked state to a named EdgeX microservice
 - Removed from scope due to shortened release time
- Register a client for (significant) change in memory usage to a named EdgeX microservice
- Deregister a client from any existing registration

Stretch goal: to facilitate native system processes call of above (like systemd) per [Delhi Release#SystemManagementTasksandNotes](#)

Microservice level support for SM



SM per microservice API

- Stop this microservice
- Get the current Admin status (locked or unlocked) ~~(not sure if this should only be valid for Device Services)~~
- Get the configuration settings (aka properties) for this microservice
- Get the configuration setting (aka property) for this microservice by configuration setting name (aka key)
- Set the configuration setting (aka property) for a writable (versus read only) property for this microservice.
 - Removed from scope due to shortened release time
- Get the memory usage for this microservice

Microservice SM Callbacks

(each microservice will send out a notification on these events)

- Notify registered clients on change to a configuration setting
- Notify registered clients on change of the operating state between enabled and disabled states. ~~(not sure if this should only be valid for Device Services)~~
- Notify registered clients on change of the admin state between locked and unlocked states. ~~(not sure if this should only be valid for Device Services)~~
 - Removed from scope due to shortened release time
- Notify registered clients on (significant) change of the memory usage