serviceOrchestrationLockManagement

Service Description

Abstract
This document provides service description for the **serviceOrchestrationLockManagement** service.

Contents

1 Overview

This document describes the **serviceOrchestrationLockManagement** service, which enables systems (with operator role or proper permissions) to create, remove and query active orchestration locks. An example of this interaction is that a higher entity (a dedicated system directly or a human operator indirectly via some tool) with management access creates orchestration locks in order to prevent the specified services from being orchestrated for a certain period of time. Also, an orchestration lock can be made automatically during the orchestration process in order to reserve an actual service when requested by a consumer system.

The **serviceOrchestrationLockManagement** service contains the following operations:

- *create* creates orchestration lock records in bulk;
- *query* lists the existing orchestration lock records that match the filtering requirements;
- *remove* deletes the specified orchestration lock records;

The rest of this document is organized as follows. In Section 2, we describe the abstract message operations provided by the service. In Section 3, we end the document by presenting the data types used by the mentioned operations.

1.1 How This Service Is Meant to Be Used

The purpose of the service is to manage the orchestration locks that allow to exclude services from the orchestration process.

It may happen that a certain service instance should not be orchestrated to the consumer systems for any reason, therefore a higher entity creates an ocrhestration lock with the service instance identifier by consuming the *create* service operation.

It also can happen that a certain service instance is reserved (locked) upon a consumer system request, but for any reason it needs to be released. In this case a higher entity can search for the associated orchestration lock record by consuming the *query* service operation in order to check the expiration time and the owner of the lock. If the service instance is need to be released before the expiration time of the lock, then the *remove* service operation can be consumed.

1.2 Important Delimitations

The requester has to identify itself to use any of the operations.

1.3 Access policy

The service is only available for operators, dedicated Core/Support systems and those who have the proper authorization rights to consume it.

2 Service Operations

This section describes the abstract signatures of each operations of the service. The **serviceOrchestrationLockManagement** service is used to *create*, *remove* and *query* orchestration lock records. In particular, each subsection names an operation, an input type and one or two output types (unsuccessful operations can return different structure), in that order. The input type is named inside parentheses, while the output type is preceded by a colon. If the operation has two output types, they are separated by a slash. Input and output types are only denoted when accepted or returned, respectively, by the operation in question. All abstract data types named in this section are defined in Section 3.

2.1 operation create (OrchestrationLockListRequest) : OrchestrationLockListResponse / ErrorResponse

Operation *create* creates orchestration lock records in bulk. Each lock requires a service instance identifier, the owner system name and an expiration time.

2.2 operation query (OrchestrationLockQueryRequest) : OrchestrationLockListResponse / ErrorResponse

Operation *query* lists the orchestration lock records that match the filtering requirements. The query data must meet the following criteria:

- The operation returns results in pages. There are default page data settings, but the requester can provide a custom specification.
- If page number is specified, the page size must be specified as well and vice versa.
- In some Local Clouds there is a maximum page size.
- If a filter expects a list, there is an OR relation between the elements of the filter.
- There is an AND relation between different kind of filters.

2.3 operation remove (OrchestrationLockRemoveRequest) : OperationStatus / ErrorResponse

Operation *remove* deletes the orchestration lock records by owner system name and service instance identifiers.

3 Information Model

Here, all data objects that can be part of the **serviceOrchestrationLockManagement** service are listed and must be respected by the hosting System. Note that each subsection, which describes one type of object, begins with the *struct* keyword, which is used to denote a collection of named fields, each with its own data type. As a complement to the explicitly defined types in this section, there is also a list of implicit primitive types in Section 3.9, which are used to represent things like hashes and identifiers.

3.1 struct OrchestrationLockListRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
locks	List<OrchestrationLockRequest>	yes	List of lock request details.

3.2 struct Identity

An Object which describes the identity of a system. It also contains whether the identified system has higher level administrative rights.

3.3 struct OrchestrationLockRequest

Field	Type	Mandatory	Description
serviceInstanceId	ServiceInstanceID	yes	Service instance to be locked.
owner	SystemName	yes	The system the lock belongs to.
expiresAt	DateTime	yes	The lock will be active until this timestamp.

3.4 struct OrchestrationLockListResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
entries	List<OrchestrationLockResponse>	List of lock records.
count	Number	Total number of lock records.

3.5 struct OrchestrationLockResponse

Field	Type	Description
id	Number	Lock record unique ID.
orchestrationJobId	OrchestrationJobId	ID of the associated orchestration job if any.
serviceInstanceId	ServiceInstanceID	ID of the locked service instance.
owner	SystemName	The system the lock belongs to.
expiresAt	DateTime	The lock is active until this timestamp.
temporary	Boolean	If true, the lock was made during an orchestration process and possibly will be released automatically within a short time (if service instance is not chosen to be reserved).

3.6 struct ErrorResponse

Field	Type	Description
status	OperationStatus	Status of the operation.
errorMessage	String	Description of the error.
errorCode	Number	Numerical code of the error.
type	ErrorType	Type of the error.
origin	String	Origin of the error.

3.7 struct OrchestrationLockQueryRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
pageNumber	Number	no (yes)	The number of the requested page. It is mandatory, if page size is specified.
pageSize	Number	no (yes)	The number of entries on the requested page. It is mandatory, if page number is specified.
pageSortField	String	no	The identifier of the field which must be used to sort the entries.
pageDirection	Direction	no	The direction of the sorting.
ids	List<Number>	no	Requester is looking for lock records with any of the specified record identifiers.
orchestrationJobIds	List<OrchestrationJobId>	no	Requester is looking for lock records with any of the specified orchestration job identifiers.
serviceInstanceIds	List<ServiceInstanceID>	no	Requester is looking for lock records with any of the specified service instance identifiers.
owners	List<SystemName>	no	Requester is looking for lock records with any of the specified owner system names.
expiresBefore	DateTime	no	Requester is looking for lock records that expires before this timestamp.
expiresAfter	DateTime	no	Requester is looking for lock records that expires after this timestamp.

3.8 struct OrchestrationLockRemoveRequest

Field	Type	Mandatory	Description
authentication	Identity	yes	The requester of the operation.
owner	SystemName	yes	System name the locks belongs.
serviceInstanceIds	List<ServiceInstanceID>	yes	List of service identifiers.

3.9 Primitives

Types and structures mentioned throughout this document that are assumed to be available to implementations of this service. The concrete interpretations of each of these types and structures must be provided by any IDD document claiming to implement this service.

Type	Description
Boolean	One out of true or false.
DateTime	Pinpoints a specific moment in time.
Direction	The direction of a sorting operation. Possible values are the representation of ascending or descending order.
ErrorType	Any suitable type chosen by the implementor of service.
List<A>	An *array* of a known number of items, each having type A.
Number	Decimal number.
Object	Set of primitives and possible further objects.
OperationStatus	Logical, textual or numerical value that indicates whether an operation is a success or a failure. Multiple values can be used for success and error cases to give additional information about the nature of the result.
OrchestrationJobId	Unique string identifier.
ServiceInstanceID	A composite string identifier that is intended to be both human and machine-readable. It consists of the instance's provider name, service definition and version, each separated by a special delimiter character. Each part must follow its related naming convention.
String	A chain of characters.
SystemName	A string identifier that is intended to be both human and machine-readable. Must follow PascalCase naming convention.

4 References

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	YYYY-MM-DD	5.0.0		Xxx Yyy

5.2 Quality Assurance

No.	Date	Version	Approved by
1	YYYY-MM-DD	5.0.0	