

# SO - L1 Service Orchestration

 **SO-2432** - Multi-domain Optical Network Service Orchestration Support in SO

CLOSED

- [SO APIs](#)
- [Service Creation Request](#)
- [SO-SDNC Request Format](#)
  - [Create Service](#)
  - [Delete Service](#)
  - [Update Async Status](#)

## SO APIs

Following are APIs to be used in this use-case

Path	Operation - Type	Description	Existing /New
/onap/so/infra/e2eServiceInstances/{version}	POST	Create an E2E Service Instance on a version provided	Existing
/business/customers/customer/{global-customer-id}/service-subscriptions/service-subscription/{service-type}/service-instances/service-instance/{service-instance-id}	PUT	Create a Service Instance in A&AI	Existing
/business/customers/customer/{global-customer-id}/service-subscriptions/service-subscription/{service-type}/service-instances/service-instance/{service-instance-id}	DELETE	Delete a Service Instance in AAI	Existing
/async/services/updateStatus	POST	Update Async Response from SDNC	New

## Service Creation Request

## ServiceCreationRequest

```
{
  "service":{
    "name":"svc-Customer1-orange-001",    // <== service instance name
    "description":"svc-Customer1-orange-001",
    "serviceInvariantUuid":"440484be-7b6e-4ca3-841a-a9d4661ec6fa",    // <== specify Invariant UUID of
service model (you can check on SDC service design screen)
    "serviceUuid":"bf82273f-f7d1-4edb-b291-e2f5bb3a4674",    // <== specify service UUID of the
model (this id will change if you check-in a new model on SDC)
    "globalSubscriberId":"Customer1",    // <== Customer name added to A&AI
    "serviceType":"???",    // <== service type, added by UsecaseUI
    "parameters":{
      "locationConstraints":[    // <== unknown; pass blank list

    ],
    "resources":[    // <== unknown; pass blank list

    ],
    "requestInputs":{    // <== pass parameters for inputs of service template
      "name": "svc-Customer1-orange-001",
      "customer": "Customer1",
      "service_provider": "Orange",
      "due_date": "2020-09-06T18:00:00+09:00",
      "end_date": "2020-09-05T18:00:00+09:00",
      "uni_id": "5847f9cc-8425-4841-9abe-2e8aeba6dd39",
      "uni_client-proto": "Ethernet",
      "uni_coding-func": "10GBASE-R",
      "uni_optical-interface": "LR",
      "enni_id": "b55a1942-fcb7-4511-bb76-02bbe52c7c85",
      /* the following attributes are optional (southbound will provide these info at discovery time ?) */
      "enni_multiplexing-capability": "ODU2e-ODU4",
      "enni_poh-odu-gcc1": true,
      "enni_poh-odu-gcc2": true,
      "enni_poh-odu-aps": true,
      "enni_poh-odu-tti": true,
      "enni_coding-func-k": "4",
      "enni_coding-func-oh-ho-oduk-tti": true,
      "enni_coding-func-oh-ho-oduk-aps": true,
      "enni_coding-func-oh-otuk-gcc0": true,
      "enni_coding-func-oh-ho-oduk-gcc2": true,
      "enni_coding-func-oh-ho-oduk-gcc1": true,
      "enni_coding-func-oh-otuk-tti": true,
      "enni_coding-func-oh-otuk-osmc": true,
      "enni_optical-interface": "P16S1-2B2"
    }
  }
}
```

## SO-SDNC Request Format

### Create Service

### SO-SDNC Request 1

```
{
  "request-ID": "<requestId>",
  "source": "UUI",
  "action": "CreateService",
  "service-Uid": "<TBD>",
  "payload": {
    "domain-type": "ROADM",
    "name": "Name from Service request",
    "rate": "100",
    "protocol": "Ethernet",
    "coding-func": "100GBASE-R",
    "service-a-end": {
      "node-id": "<generic-node-uuid>",
      "port": "1/0/0/C1"
    },
    "service-z-end": {
      "node-id": "<generic-node-uuid>",
      "port": "1/0/0/C3"
    },
    "service-layer": "OTN",
    "due-date": "yyyy-mm-ddT00:00:01Z"
  }
}
```

### SO-SDNC Request 2

```
{
  "request-ID": <requestId>,
  "source": "UUI",
  "action": "CreateService", "service-Uid": "<TBD>",
  "payload": {
    "domain-type": "TAPI",
    "name": "Name from Service request",
    "rate": "100",
    "protocol": "Ethernet",
    "Coding-func": "100GBASE-R",
    "service-a-end": {
      "node-id": "<generic-node-uuid>",
      "port": "1/0/0/C2"
    },
    "service-z-end": {
      "node-id": "<generic-node-uuid>",
      "port": "1/0/0/C4" },
    "service-layer": "OTN",
    "due-date": "yyyy-mm-ddT00:00:01Z"
  }
}
```

### Delete Service

#### DeleteServiceRequest

```
{
  "requestID":<requestId>,
  "source":"UUI",
  "action":"DeleteService",
  "serviceUuid": "<TBD>",
  "globalSubscriberId": <globalSubscriberId from BSS request>,
  "serviceType": <serviceType from BSS request>,
  "payload":{ "domain-type" : "<Lookedup by 'generic-node-uuid' from A&AI">,"name":"Name from Service request"
  }
}
```

#### Update Async Status

#### SDNCAsyncResponse

```
{
  "output": {
    "response-code": "400",
    "response-message": "Deletion in progress."
  }
}
```