

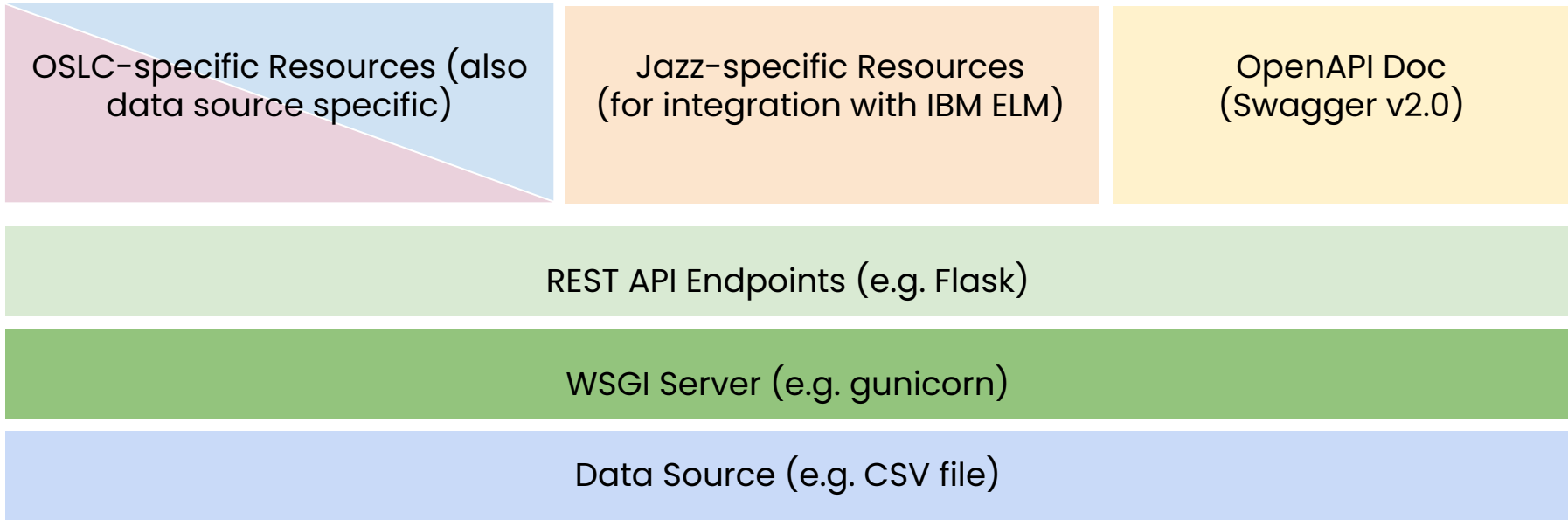
PyOSLC

Existing Architecture

Mario Jiménez Carrasco

May 12, 2021

PyOSLC Building Blocks



The diagram illustrates the PyOSLC Building Blocks architecture. It consists of three main components at the top, each in a colored box: 'OSLC-specific Resources (also data source specific)' in a blue and pink box, 'Jazz-specific Resources (for integration with IBM ELM)' in an orange box, and 'OpenAPI Doc (Swagger v2.0)' in a yellow box. Below these are three horizontal bars representing the infrastructure layers: 'REST API Endpoints (e.g. Flask)' in a light green bar, 'WSGI Server (e.g. gunicorn)' in a dark green bar, and 'Data Source (e.g. CSV file)' in a light blue bar.

OSLC-specific Resources (also
data source specific)

Jazz-specific Resources
(for integration with IBM ELM)

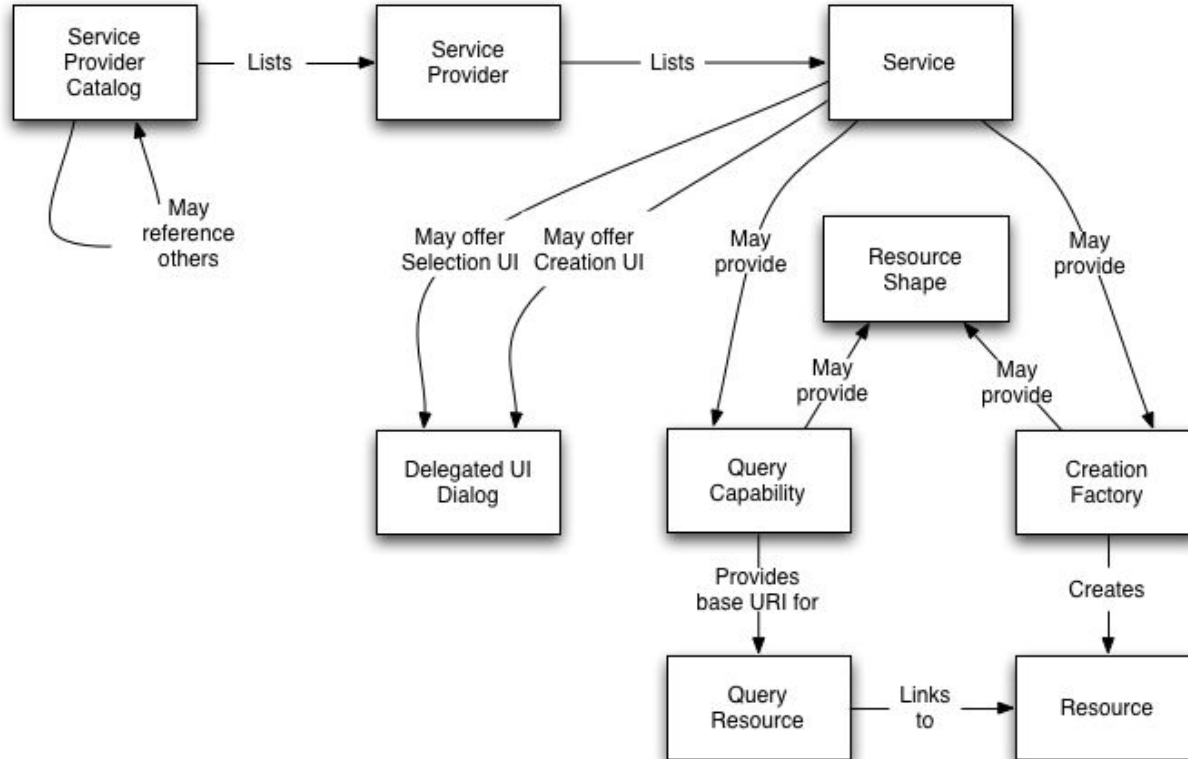
OpenAPI Doc
(Swagger v2.0)

REST API Endpoints (e.g. Flask)

WSGI Server (e.g. gunicorn)

Data Source (e.g. CSV file)

OSLC Resources for Resource Discovery



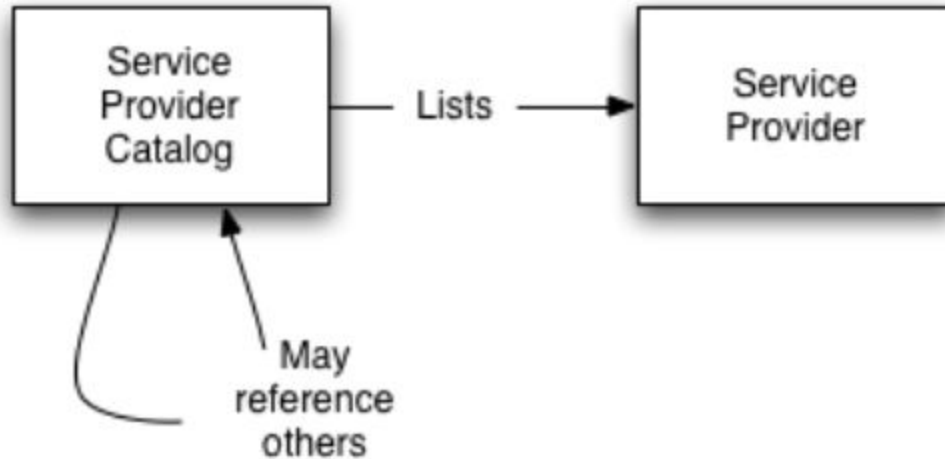
PyOSLC Review Approach

For each OSLC-specific resource (e.g. OSLC ServiceProviderCatalog)

- See an example RDF representation in Turtle
- See Python code of the endpoint and of the Python SDK method used
- See in the Python code how OSLC resources can be configured (e.g. relationships to other OSLC resources)
- See the Python code creating the RDF representation of OSLC resources

OSLC Service Provider Catalog

OSLC Service Provider Catalog



Service Provider Catalog

GET

<http://localhost:5000/oslc/services/catalog>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/catalog> a oslc:ServiceProviderCatalog ;
6   oslc:domain j.0: , <http://open-services.net/ns/rm#> ;
7   oslc:serviceProvider <http://localhost:5000/oslc/services/provider/Project-1> ;
8   dcterms:description "A Service Provider for the Contact Software Platform." ;
9   dcterms:title "Contact Software Platform Service Provider Catalog" .
10 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
11   j.0:oauthConfiguration aware yes ""xsd:string ;
12   j.0:supportContributionsToLinkIndexProvider true ;
13   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
14   j.0:supportOSLCSimpleQuery true ;
15   oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
16   oslc:service [ a oslc:Service ;
17     oslc:creationDialog [ a oslc:Dialog ;
18       oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
19       oslc:label "Creation Dialog Service" ;
20       oslc:resourceType <http://open-services.net/ns/am#Resource> ,
21         <http://open-services.net/ns/cm#ChangeRequest> ,
22         <http://open-services.net/ns/rm#Requirement> ;
23       oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
24       dcterms:hintHeight "325px" ;
25       dcterms:hintWidth "525px" ;
26       dcterms:title "Creation Dialog Service" ] ;
27     oslc:creationFactory [ a oslc:CreationFactory ;
28       oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
29       oslc:label "Creation Factory" ;
30       oslc:resourceShape <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
31       oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
32       dcterms:title "Creation Factory" ] ;
33     oslc:domain <http://open-services.net/ns/rm#> ;
34     oslc:queryCapability [ a oslc:QueryCapability ;
35       oslc:label "Query Capability"^^xsd:string ;
36       oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
37       oslc:resourceShape <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
38       oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
39       dcterms:title "Query Capability" ] ;
40     oslc:selectionDialog [ a oslc:Dialog ;
41       oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
42       oslc:label "Selection Dialog Service" ;
43       oslc:resourceType <http://open-services.net/ns/am#Resource> ,
44         <http://open-services.net/ns/cm#ChangeRequest> ,
45         <http://open-services.net/ns/rm#Requirement> ;
46       oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
47       dcterms:hintHeight "325px" ;
48       dcterms:hintWidth "525px" ;
49       dcterms:title "Selection Dialog Service" ] ] ;
50   dcterms:description "Service Provider for the Contact Software platform service (id: Project-1; kind: Specification)" ;
51   dcterms:identifier "Project-1"^^xsd:string ;
52   dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

- Only one ServiceProvider
- No nested ServiceProviderCatalogs
- oslc:oauthConfiguration supported but not specified
- dcterms:publisher supported but not specified

SDK for ServiceProviderCatalog

Explanation

SPC the entry point of OSLC API

API

app.api.adapter.namespace.
core.py

API:
ServiceProviderCatalog /
get method

app.api.adapter.services.
providers.py

API:
ServiceProviderCatalogSingleton /
get_catalog method

Initialize_providers method

SDK

pyoslc.resources.
models.py

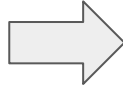
SDK:
ServiceProviderCatalog

class **ServiceProviderCatalog**(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None, uri=None,
publisher=None, domain=None,
service_provider_catalog=None,
auth_configuration=None)
```


Definition of ServiceProviders

Number of service
Providers configured in
CSVImplementation
class



```
service_providers = [{  
    'id': 'Project-1',  
    'name': Service Provider for Project 1,  
    'class': Specification  
}]
```

No nested SPCs
possible at the
moment



ServiceProviderCatalogSingleton



ContactServiceProviderFactory

RDF Generation of ServiceProviderCatalog

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
267 class ServiceProviderCatalog(BaseResource):
268
269     def __init__(self, about=None, types=None,
270                 identifier=None, short_title=None,
271                 creator=None, subject=None,
272                 discussed_by=None, instance_of=None,
273                 relation=None, uri=None, publisher=None,
274                 service_provider_catalog=None,
```

to_rdf() Python object
to RDF translation

```
326 def to_rdf(self, graph):
327     super(ServiceProviderCatalog, self).to_rdf(graph)
328
329     spc = Resource(graph, URIRef(self.about))
330     spc.add(RDF.type, OSLC.ServiceProviderCatalog)
331
332     if self.title:
333         spc.add(DCTERMS.title, Literal(self.title))
334
335     if self.description:
336         spc.add(DCTERMS.description, Literal(self.description))
337
338     if self.publisher:
339         spc.add(DCTERMS.publisher, URIRef(self.publisher.about))
```

Mapping from Python
object type to RDF

Mapping from Python
object attributes to RDF

OSLC Service Provider

OSLC Service Provider



Service-Provider

GET

<http://localhost:5000/oslc/services/provider/Project-1>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
6   j.0:globalConfigurationAware "yes"^^xsd:String ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
11  oslc:service [ a oslc:Service ;
12    oslc:creationDialog [ a oslc:Dialog ;
13      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14      oslc:label "Creation Dialog service" ;
15      oslc:resourceType <http://open-services.net/ns/am#Resource>,
16        <http://open-services.net/ns/cm#ChangeRequest>,
17        <http://open-services.net/ns/rm#Requirement> ;
18      oslc:usage <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
19      dcterms:hintHeight "325px" ;
20      dcterms:hintWidth "525px" ;
21      dcterms:title "Creation Dialog" ;
22    ] ;
23    oslc:creationFactory [ a oslc:Factory ;
24      oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
25      oslc:label "Creation Factory"^^xsd:string ;
26      oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
27      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
28      dcterms:title "Creation Factory" ] ;
29    oslc:domain <http://open-services.net/ns/rm#> ;
30    oslc:queryCapability [ a oslc:QueryCapability ;
31      oslc:label "Query Capability"^^xsd:string ;
32      oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
33      oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
34      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
35      dcterms:title "Query Capability" ] ;
36    oslc:selectionDialog [ a oslc:Dialog ;
37      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
38      oslc:label "Selection Dialog Service" ;
39      oslc:resourceType <http://open-services.net/ns/am#Resource>,
40        <http://open-services.net/ns/cm#ChangeRequest>,
41        <http://open-services.net/ns/rm#Requirement> ;
42      oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
43      dcterms:hintHeight "325px" ;
44      dcterms:hintWidth "525px" ;
45      dcterms:title "Selection Dialog" ] ] ;
46  dcterms:description "Service Provider for the Contact Software platform service (id: Project-1; kind: Specification)" ;
47  dcterms:identifier "Project-1"^^xsd:string ;
48  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

- Only one Service

SDK for ServiceProvider

Explanation

SP given the configuration taken from the **CSVImplementation**.

API

app.api.adapter.namespace.
core.py

API:
ServiceProvider /
get method

app.api.adapter.services.
providers.py

API:
ServiceProviderCatalogSingleton /
get_provider method

SDK

pyoslc.resources.
models.py

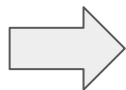
SDK:
ServiceProvider

```
class ServiceProvider(BaseResource):  
  
    def __init__(self, about=None, types=None,  
                  properties=None, description=None,  
                  identifier=None, short_title=None, title=None,  
                  contributor=None, creator=None, subject=None,  
                  created=None, modified=None, type=None,  
                  discussed_by=None, instance_shape=None,  
                  service_provider=None, relation=None,  
                  publisher=None, service=None, details=None,  
                  prefix_definition=None,  
                  auth_configuration=None)
```

Definition of Services

Currently hardcoded: One Service per Service Provider

Number of services
hardcoded in **core.py**
module



```
config_service_resource(  
    'specification', ServiceResource,  
    'app.api.adapter.services.specification', 'Specification',  
)
```



ContactServiceProviderFactory

get_service_resources()

RDF Generation of ServiceProvider

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
362 class ServiceProvider(BaseResource)
363
364     def __init__(self, about=None,
365                 identifier=None, s
366                 creator=None, subj
367                 discussed_by=None,
368                 relation=None, pub
```

to_rdf() Python object
to RDF translation

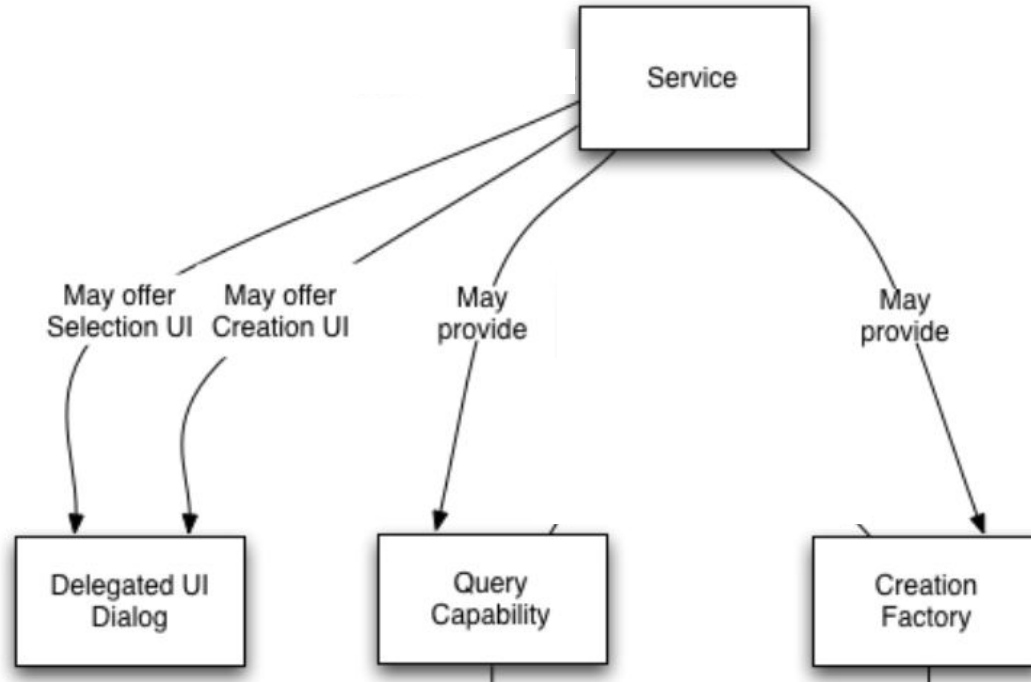
```
434 def to_rdf(self, graph):
435     super(ServiceProvider, self).to_rdf(graph)
436
437     uri = self.about if self.about.__contains__(self.identifier) \
438         else self.about + '/{}'.format(self.identifier) if self.iden
439
440     sp = Resource(graph, URIRef(uri))
441     sp.add(RDF.type, OSLC.ServiceProvider)
442
443     if self.identifier:
444         sp.add(DCTERMS.identifier, Literal(self.identifier, datatype=
445
446     if self.title:
447         sp.add(DCTERMS.title, Literal(self.title, datatype=XSD.Liter
```

Mapping from Python
object type to RDF

Mapping from Python
object attributes to RDF

OSLC Service

OSLC Service



Service

Service resource is described inside the ServiceProvider resource

GET

<http://localhost:5000/oslc/services/provider/Project-1>

- One QueryCapability
- One CreationFactory
- One CreationDialog
- One SelectionDialog

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/prod/...> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1/resources/creator>
6   j.0:globalConfigurationAware "yes"^^xsd:string ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
11  oslc:service [ a oslc:Service ;
12                oslc:creationDialog [ a oslc:Dialog ;
13                                      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14                                      oslc:label "Creation Dialog Service" ;
15                                      oslc:resourceType <http://open-services.net/ns/am#Resource>,
16                                      <http://open-services.net/ns/cm#ChangeRequest>,
17                                      <http://open-services.net/ns/rm#Requirement> ;
18                                      oslc:usage <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
19                                      dcterms:hintHeight "325px" ;
20                                      dcterms:hintWidth "525px" ;
21                                      dcterms:title "Creation Dialog" ] ;
22                oslc:creationFactory [ a oslc:CreationFactory ;
23                                       oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
24                                       oslc:label "Creation Factory"^^xsd:string ;
25                                       oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
26                                       oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
27                                       dcterms:title "Creation Factory" ] ;
28                oslc:domain <http://open-services.net/ns/rm#> ;
29                oslc:queryCapability [ a oslc:QueryCapability ;
30                                      oslc:label "Query Capability"^^xsd:string ;
31                                      oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
32                                      oslc:resourceShape <http://localhost:5000/oslc/services/resourceShapes/requirement> ;
33                                      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
34                                      dcterms:title "Query Capability" ] ;
35                oslc:selectionDialog [ a oslc:Dialog ;
36                                       oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
37                                       oslc:label "Selection Dialog Service" ;
38                                       oslc:resourceType <http://open-services.net/ns/am#Resource>,
39                                       <http://open-services.net/ns/cm#ChangeRequest>,
40                                       <http://open-services.net/ns/rm#Requirement> ;
41                                       oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
42                                       dcterms:hintHeight "325px" ;
43                                       dcterms:hintWidth "525px" ;
44                                       dcterms:title "Selection Dialog" ] ] ;
45  dcterms:description "Service Provider for the contact software platform service (id: Project-1; kind: Specification)" ;
46  dcterms:identifier "Project-1"^^xsd:string ;
47  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

SDK for Service

Explanation

Each ServiceProvider has a Service
(hardcoded)



API

API:
NA

Representation of Service described in
the representation of the ServiceProvider



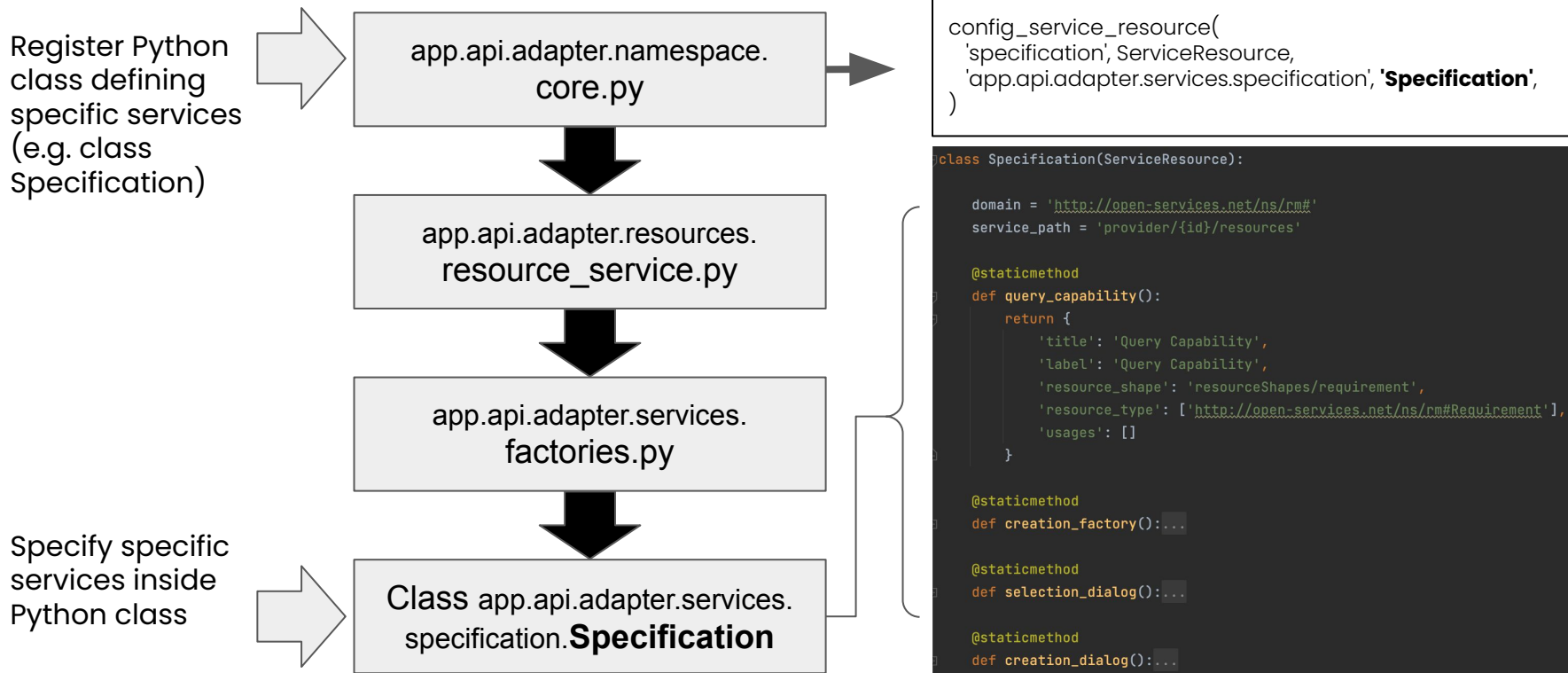
SDK

pyoslc.resources.
models.py

SDK:
Service

```
class Service(BaseResource):  
  
    def __init__(self, about=None, types=None,  
                  properties=None, description=None,  
                  identifier=None, short_title=None, title=None,  
                  contributor=None, creator=None, subject=None,  
                  created=None, modified=None, type=None,  
                  discussed_by=None, instance_shape=None,  
                  service_provider=None, relation=None,  
                  domain=None, creation_factory=None,  
                  query_capability=None, selection_dialog=None,  
                  creation_dialog=None, usage=None)
```

Definition of Specific Services



RDF Generation of Service

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
479 class Service(BaseResource):
480
481     def __init__(self, about=None, title=None, identifier=None, subject=None, creator=None, subject=None, discussed_by=None, relation=None, domain=None,
```

to_rdf() Python
object to RDF
translation

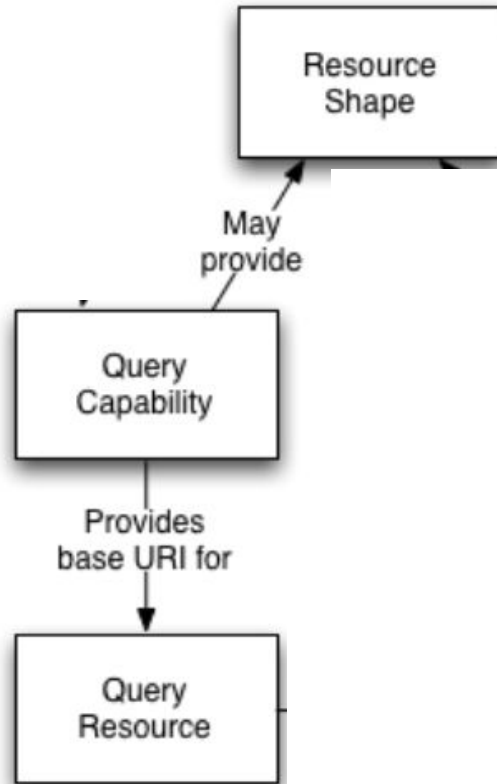
```
555 def to_rdf(self, graph):
556     super(Service, self).to_rdf(graph)
557
558     # uri = self.about if self.about.__contains__(self.identifier) else self.about
559     # self.identifier) if self.identifier else ''
560
561     s = Resource(graph, BNode())
562     s.add(RDF.type, OSLC.Service)
563
564     if self.title:
565         s.add(DCTERMS.title, Literal(self.title, datatype=XSD.Literal))
566
567     if self.description:
568         s.add(DCTERMS.description, Literal(self.description, datatype=XSD.Literal))
```

Mapping from
Python object type to
RDF

Mapping from
Python object
attributes to RDF

OSLC Query Capability

OSLC Query Capability



Query Capability

QueryCapability resource is described inside the ServiceProvider resource

GET

<http://localhost:5000/oslc/services/provider/Project-1>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
6   j.0:globalConfigurationAware "yes"^^xsd:String ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
11  oslc:service [ a oslc:Service ;
12    oslc:creationDialog [ a oslc:Dialog ;
13      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14      oslc:label "Creation Dialog Service" ;
15      oslc:resourceType <http://open-services.net/ns/am#Resource>,
16        <http://open-services.net/ns/cm#ChangeRequest>,
17        <http://open-services.net/ns/rm#Requirement> ;
18      oslc:usage <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
19      dcterms:hintHeight "325px" ;
20      dcterms:hintWidth "525px" ;
21      dcterms:title "Creation Dialog" ] ;
22    oslc:creationFactory [ a oslc:CreationFactory ;
23      oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
24      oslc:label "Creation Factory"^^xsd:string ;
25      oslc:resourceShape <http://localhost:5000/oslc/services/resourceShapes/requirement> ;
26      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
27      dcterms:title "Creation Factory" ] ;
28    oslc:domain <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
29    oslc:queryCapability [ a oslc:QueryCapability ;
30      oslc:label "Query Capability"^^xsd:string ;
31      oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
32      oslc:resourceShape <http://localhost:5000/oslc/services/resourceShapes/requirement> ;
33      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
34      dcterms:title "Query Capability" ] ;
35    oslc:selectionDialog [ a oslc:Dialog ;
36      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
37      oslc:label "Selection Dialog Service" ;
38      oslc:resourceType <http://open-services.net/ns/am#Resource>,
39        <http://open-services.net/ns/cm#ChangeRequest>,
40        <http://open-services.net/ns/rm#Requirement> ;
41      oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
42      dcterms:hintHeight "325px" ;
43      dcterms:hintWidth "525px" ;
44      dcterms:title "Selection Dialog" ] ] ;
45  dcterms:description "Service Provider for the Contact Software platform service (id: Project-1; kind: Specification)" ;
46  dcterms:identifier "Project-1"^^xsd:string ;
47  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

- One Query Resource
- One Resource Shape

SDK for QueryCapability

Explanation

Each Service has a QueryCapability
(hardcoded)



API

API:
NA

Representation of Service described in
the representation of the ServiceProvider



SDK

pyoslc.resources.
models.py

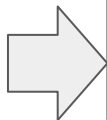
SDK:
QueryCapability

class **QueryCapability**(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
label=None, query_base=None, usage=None,
resource_type=None, resource_shape=None)
```

Definition of Resource Shape

Specify
ResourceShape
in URL in class
Specification



app.api.adapter.services.specification.
specification.py



```
def query_capability():  
    return {  
        'title': 'Query Capability',  
        'label': 'Query Capability',  
        'resource_shape': 'resourceShapes/requirement',  
        'resource_type': ['http://open-services.net/ns/rm#Requirement'],  
        'usages': []  
    }
```



Specify last path of
QueryBase URL as
hardcoded

app.api.adapter.resources.factories.py
ServiceProviderFactory



```
base_path = base_uri + '/'  
class_path = 'provider/{id}/resources'  
method_path = 'requirement'
```

RDF Generation of QueryCapability

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
596 class QueryCapability(BaseResource):
597
598     def __init__(self, about=None, types=None,
599                 identifier=None, short_title=None,
600                 creator=None, subject=None,
601                 discussed_by=None, instance_of=None,
602                 relation=None, label=None,
603                 resource_type=None, resource_shape=None,
```

to_rdf() Python object
to RDF translation

```
666 def to_rdf(self, graph):
667     super(QueryCapability, self).to_rdf(graph)
668
669     qc = Resource(graph, BNode())
670     qc.add(RDF.type, OSLC.QueryCapability)
671
672     if self.title:
673         qc.add(DCTERMS.title, Literal(self.title))
674
```

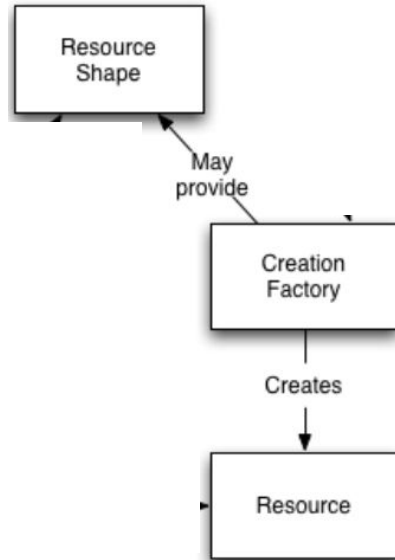
Mapping from Python
object type to RDF

Mapping from Python
object attributes to RDF

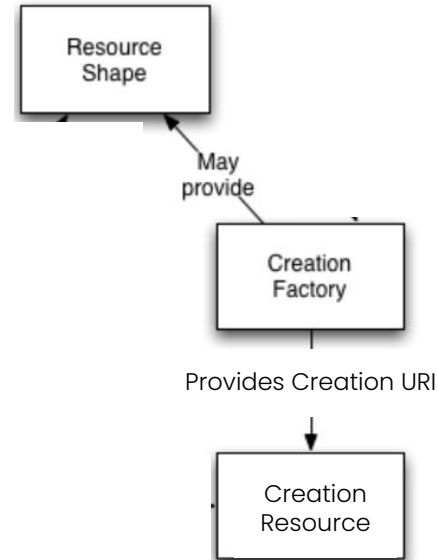
OSLC Creation Factory

OSLC Creation Factory

Extract from **Original OSLC** Spec diagram



Corrected OSLC diagram



[Reference to spec](#)

Creation Factory

CreationFactory
resource is described
inside the
ServiceProvider resource

GET

<http://localhost:5000/oslc/services/provider/Project-1>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
6   j.0:globalConfigurationAware "yes"^^xsd:String ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
11  oslc:service [ a oslc:Service ;
12                oslc:creationDialog [ a oslc:Dialog ;
13                                      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14                                      oslc:label "Creation Dialog Service" ;
15                                      oslc:resourceType <http://open-services.net/ns/am#Resource> ;
16                                      <http://open-services.net/ns/cm#ChangeRequest> ,
17                                      <http://open-services.net/ns/rm#Requirement> ;
18                                      oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
19                                      dcterms:hintHeight "325px" ;
20                                      dcterms:hintWidth "525px" ;
21                                      dcterms:title "Creation Dialog" ] ;
22                oslc:creationFactory [ a oslc:CreationFactory ;
23                                       oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
24                                       oslc:label "Creation Factory"^^xsd:string ;
25                                       oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
26                                       oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
27                                       dcterms:title "Creation Factory" ] ;
28                oslc:domain <http://open-services.net/ns/rm#> ;
29                oslc:queryCapability [ a oslc:QueryCapability ;
30                                       oslc:label "Query Capability"^^xsd:string ;
31                                       oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
32                                       oslc:resourceShape <http://localhost:5000/oslc/services/resourceShapes/requirement> ;
33                                       oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
34                                       dcterms:title "Query Capability" ] ;
35                oslc:selectionDialog [ a oslc:Dialog ;
36                                       oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
37                                       oslc:label "Selection Dialog Service" ;
38                                       oslc:resourceType <http://open-services.net/ns/am#Resource> ,
39                                       <http://open-services.net/ns/cm#ChangeRequest> ,
40                                       <http://open-services.net/ns/rm#Requirement> ;
41                                       oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
42                                       dcterms:hintHeight "325px" ;
43                                       dcterms:hintWidth "525px" ;
44                                       dcterms:title "Selection Dialog" ] ] ;
45  dcterms:description "Service Provider for the Contact Software platform service (id: Project-1; kind: Specification)" ;
46  dcterms:identifier "Project-1"^^xsd:string ;
47  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

- One Creation Resource
- One Resource Shape

SDK for CreationFactory

Explanation

Each Service has a CreationFactory
(hardcoded)

API

API:
NA

Representation of Service described in
the representation of the ServiceProvider

SDK

pyoslc.resources.
models.py

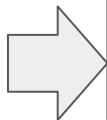
SDK:
CreationFactory

class **CreationFactory**(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
label=None, creation=None, usage=None,
resource_type=None, resource_shape=None)
```


Definition of Resource Shape

Specify
ResourceShape
in URL in class
Specification



app.api.adapter.services.specification.
specification.py



```
def creation_factory():  
    return {  
        'title': 'Creation Factory',  
        'label': 'Creation Factory',  
        'resource_shape': ['resourceShapes/requirement'],  
        'resource_type': ['http://open-services.net/ns/rm#Requirement'],  
        'usages': []  
    }
```



Specify last path of
QueryBase URL as
hardcoded

app.api.adapter.resources.factories.py

ServiceProviderFactory



```
class_path = 'provider/{id}/resources'  
method_path = 'requirement'
```

RDF Generation of CreationFactory

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterm', DCTERMS)
25         self.graph.bind('j', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
697 class CreationFactory(BaseResource):
698
699     def __init__(self, about=None, types=None,
700                 identifier=None, short_title=None,
701                 creator=None, subject=None,
702                 discussed_by=None, instance_of=None,
703                 relation=None, label=None,
704                 resource_type=None, resource_shape=None,
```

to_rdf() Python object
to RDF translation

```
770 def to_rdf(self, graph):
771     super(CreationFactory, self).to_rdf(graph)
772
773     cf = Resource(graph, BNode())
774     cf.add(RDF.type, OSLC.CreationFactory)
775
776     if self.title:
777         cf.add(DCTERMS.title, Literal(self.title))
```

Mapping from Python
object type to RDF

Mapping from Python
object attributes to RDF

OSLC Delegated UI Dialog

Creation Factory

CreationFactory
resource is described
inside the
ServiceProvider resource

GET

<http://localhost:5000/oslc/services/provider/Project-1>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
6   j.0:globalConfigurationAware "yes"^^xsd:String ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
11  oslc:service [ a oslc:Service ;
12    oslc:creationDialog [ a oslc:Dialog ;
13      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14      oslc:label "Creation Dialog Service" ;
15      oslc:resourceType <http://open-services.net/ns/am#Resource>,
16        <http://open-services.net/ns/cm#ChangeRequest>,
17        <http://open-services.net/ns/rm#Requirement> ;
18      oslc:usage <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
19      dcterms:hintHeight "325px" ;
20      dcterms:hintWidth "525px" ;
21      dcterms:title "Creation Dialog" ] ;
22    oslc:creationFactory [ a oslc:CreationFactory ;
23      oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
24      oslc:label "Creation Factory"^^xsd:string ;
25      oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
26      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
27      dcterms:title "Creation Factory" ] ;
28    oslc:domain <http://open-services.net/ns/rm#> ;
29    oslc:queryCapability [ a oslc:QueryCapability ;
30      oslc:label "Query Capability"^^xsd:string ;
31      oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
32      oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/selector> ;
33      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
34      dcterms:title "Query Capability" ] ;
35    oslc:selectionDialog [ a oslc:Dialog ;
36      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
37      oslc:label "Selection Dialog Service" ;
38      oslc:resourceType <http://open-services.net/ns/am#Resource>,
39        <http://open-services.net/ns/cm#ChangeRequest>,
40        <http://open-services.net/ns/rm#Requirement> ;
41      oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
42      dcterms:hintHeight "325px" ;
43      dcterms:hintWidth "525px" ;
44      dcterms:title "Selection Dialog" ] ] ;
45  dcterms:description "Service Provider for the contact software platform service (id: Project-1; kind: Specification)" ;
46  dcterms:identifier "Project-1"^^xsd:string ;
47  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:literal .
```

SelectionDialog Resource

Creation Factory

CreationFactory resource is described inside the ServiceProvider resource

GET

<http://localhost:5000/oslc/services/provider/Project-1>

```
1 @prefix dcterms: <http://purl.org/dc/terms/> .
2 @prefix j.0: <http://jazz.net/xmlns/prod/jazz/process/1.0/> .
3 @prefix oslc: <http://open-services.net/ns/core#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 <http://localhost:5000/oslc/services/provider/Project-1> a oslc:ServiceProvider ;
6   j.0:globalConfigurationAware "yes"^^xsd:String ;
7   j.0:supportContributionsToLinkIndexProvider true ;
8   j.0:supportLinkDiscoveryViaLinkIndexProvider true ;
9   j.0:supportOSLCSimpleQuery true ;
10  oslc:details <http://localhost:5000/oslc/services/provider/Project-1> ;
11  oslc:service [ a oslc:Service ;
12    oslc:creationDialog [ a oslc:Dialog ;
13      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/creator> ;
14      oslc:label "Creation Dialog service" ;
15      oslc:resourceType <http://open-services.net/ns/am#Resource>,
16        <http://open-services.net/ns/cm#ChangeRequest>,
17        <http://open-services.net/ns/rm#Requirement> ;
18      oslc:usage <http://open-services.net/ns/am#PyOSLCCreationDialog> ;
19      dcterms:hintHeight "325px" ;
20      dcterms:hintWidth "525px" ;
21      dcterms:title "Creation Dialog" ] ;
22    oslc:creationFactory [ a oslc:CreationFactory ;
23      oslc:creation <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
24      oslc:label "Creation Factory"^^xsd:string ;
25      oslc:resourceShape <http://localhost:5000/oslc/services/catalog/resourceShapes/requirement> ;
26      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
27      dcterms:title "Creation Factory" ] ;
28    oslc:domain <http://open-services.net/ns/rm#> ;
29    oslc:queryCapability [ a oslc:QueryCapability ;
30      oslc:label "Query Capability"^^xsd:string ;
31      oslc:queryBase <http://localhost:5000/oslc/services/provider/Project-1/resources/requirement> ;
32      oslc:resourceShape <http://localhost:5000/oslc/services/resourceShapes/requirement> ;
33      oslc:resourceType <http://open-services.net/ns/rm#Requirement> ;
34      dcterms:title "Query Capability" ] ;
35    oslc:selectionDialog [ a oslc:Dialog ;
36      oslc:dialog <http://localhost:5000/oslc/services/provider/Project-1/resources/selector> ;
37      oslc:label "Selection Dialog Service" ;
38      oslc:resourceType <http://open-services.net/ns/am#Resource>,
39        <http://open-services.net/ns/cm#ChangeRequest>,
40        <http://open-services.net/ns/rm#Requirement> ;
41      oslc:usage <http://open-services.net/ns/am#PyOSLCSelectionDialog> ;
42      dcterms:hintHeight "325px" ;
43      dcterms:hintWidth "525px" ;
44      dcterms:title "Selection Dialog" ] ] ;
45  dcterms:description "Service Provider for the Contact Software platform service (id: Project-1; kind: Specification)" ;
46  dcterms:identifier "Project-1"^^xsd:string ;
47  dcterms:title "PyOSLC Service Provider for Project 1"^^xsd:Literal .
```

CreationDialog Resource

SDK for DelegatedUI

Explanation

Each Service has a pair of Dialogs (hardcoded):

CreationDialog
and
SelectionDialog

API

API:
NA

Representation of Service described in the representation of the ServiceProvider

SDK

pyoslc.resources.
models.py

SDK:
Dialog

class **Dialog**(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
dialog=None, hint_height=None,
hint_width=None, label=None, usage=None,
resource_type=None)
```

Definition of Dialogs / UI

Dialogs configuration are hardcoded in class **Specification**

app.api.adapter.services.specification.
specification.py

```
def selection_dialog():  
    return {  
        'title': 'Selection Dialog',  
        'label': 'Selection Dialog Service',  
        'uri': 'provider/{id}/resources/selector',  
        'hint_width': '525px',  
        'hint_height': '325px',  
        'resource_type': [  
            ...  
            'http://open-services.net/ns/rm#Requirement'  
        ],  
        'usages': [  
            'http://open-services.net/ns/am#PyOSLCSelectionDialog'  
        ]  
    }
```

Both, selection and creation dialog use the same configuration



Specify the path of Dialog URL as hardcoded

app.api.adapter.resources.factories.py
ServiceProviderFactory

```
'label': 'Selection Dialog Service',  
'uri': 'provider/{id}/resources/selector',
```


RDF Generation of DelegatedDialog

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
800 class Dialog(BaseResource):
801
802     def __init__(self, about=None, types=None,
803                 identifier=None, short_title=None,
804                 creator=None, subject=None,
805                 discussed_by=None, instance=None,
806                 relation=None, dialog=None,
807                 label=None, usage=None, reso
```

to_rdf() Python object
to RDF translation

```
876 def to_rdf(self, graph):
877     super(Dialog, self).to_rdf(graph)
878
879     d = Resource(graph, BNode())
880     d.add(RDF.type, OSLC.Dialog)
881
882     if self.label:
883         d.add(OSLC.label, Literal(self.label))
884
885     if self.title:
886         d.add(DCTERMS.title, Literal(self.title))
```

Mapping from Python
object type to RDF

Mapping from Python
object attributes to RDF

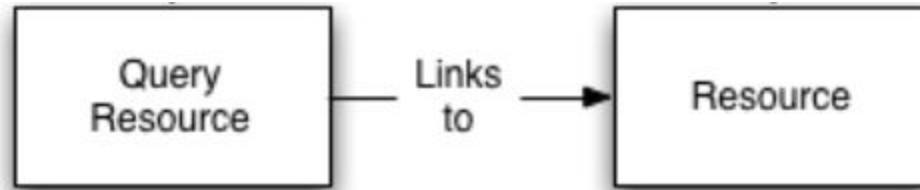
OSLC Resource Shape

OSLC Resource Shape

- Not supported yet
- GET on ResourceShape returns 404

OSLC Query Resource

OSLC Query Resource



Query Resource

GET

<http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement>

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix oslc: <http://open-services.net/ns/core#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement> a oslc:ResponseInfo ;
    oslc:totalCount 5 ;
    dcterms:title "Query Results for Requirements"^^xsd:Literal ;
    rdfs:member <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement/X1C2V3B1>,
        <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement/X1C2V3B2>,
        <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement/X1C2V3B3>,
        <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement/X1C2V3B4>,
        <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirements/requirement/X1C2V3B5>
```

SDK for Query Resource

Explanation

Each QueryBase will generate the list of resources available in the datasource for the resource type

API

app.api.adapter.namespace.
core.py

API:
ResourceOperation /
get method

app.api.adapter.namespaces.
business.py

API:
get_requirement_list method

SDK

pyoslc.resources.
models.py

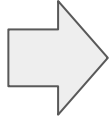
SDK:
ResponseInfo

class **ResponseInfo**(FilteredResource):

```
def __init__(self, about=None, title=None,  
types=None, properties=None, resource=None,  
total_count=None, next_page=None,  
container=None)
```

Definition of ResponseInfo (Requirements)

API defining
specific
endpoint

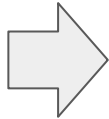


app.api.adapter.namespace.
core.py



```
def get(self, service_provider_id):  
    ...  
    select = request.args.get('oslc.select', '')  
    where = request.args.get('oslc.where', '')  
    ...  
    data = get_requirement_list(base_url, select, where)
```

Get the
requirements
from CSV file

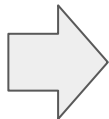


app.api.adapter.namespaces.
business.py



```
def get_requirement_list(base_url, select, where):  
    requirements = list()  
    path = 'examples/specifications.csv'  
    if os.path.isfile(path):  
        with open(path, 'r') as f:  
            reader = csv.DictReader(f, delimiter=';')  
            for row in reader:  
                requirement = Requirement()  
                requirement.update(row, attributes=attributes)  
                requirements.append(requirement)  
    return requirements
```

Returning the
ResponseInfo



app.api.adapter.namespace.
core.py



```
def get(self, service_provider_id):  
    ...  
    response_info = ResponseInfo(base_url)  
    response_info.total_count = len(data)  
    response_info.title = 'Query Results for Requirements'  
    response_info.members = data  
    response_info.to_rdf(self.graph)
```

RDF Generation of QueryResource

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

Models.py / OSLC Resource class

```
1104 class ResponseInfo(FilteredResource):
1105
1106     def __init__(self, about=None, title=None,
1107                 types=None, properties=None,
1108                 resource=None, total_count=None, next_page=None,
1109                 container=None):
1110         super(ResponseInfo, self).__init__(about, types, prop
1111         self._total_count = total_count if total_count is not
1112         self._next_page = next_page if next_page is not None
1113         self._container = container if container is not None
1114         self._members = list()
```

to_rdf() Python
object to RDF
translation

```
48 def to_rdf(self, graph):
49     super(ResponseInfo, self).to_rdf(graph)
50
51     uri = self.about
52     ri = Resource(graph, URIRef(uri))
53     ri.add(RDF.type, OSLC.ResponseInfo)
54
55     if self.title:
56         ri.add(DCTERMS.title, Literal(self.title, datatype=XSD.Literal))
```

Mapping from
Python object
type to RDF

Mapping from
Python object
attributes to
RDF

OSLC Resource

Resource

GET <http://127.0.0.1:5000/oslc/services/provider/Project-1/resources/requirement/X1C2V3B1>

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix oslc_rm: <http://open-services.net/ns/rm#> .

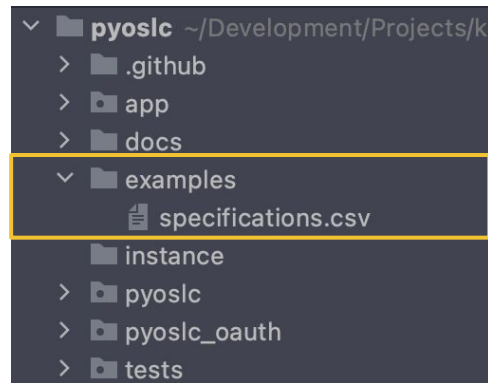
<http://localhost:5000/oslc/services/provider/Project-1/resources/requirement/X1C2V3B1> a oslc_rm:Requirement ;
    oslc_rm:affectedBy "0" ;
    oslc_rm:constrainedBy "Customer Requirement" ;
    oslc_rm:decomposedBy "Draft" ;
    oslc_rm:elaboratedBy "Ian Altman" ;
    oslc_rm:satisfiedBy "Software Development" ;
    oslc_rm:trackedBy "0" ;
    oslc_rm:validatedBy "1" ;
    dcterms:creator "Mario" ;
    dcterms:description "The OSLC RM Specification needs to be awesome 1" ;
    dcterms:identifier "X1C2V3B1" ;
    dcterms:shortTitle "SDK-Dev" ;
    dcterms:subject "Project-1" ;
    dcterms:title "The ACRV shall provide medical life-support accommodations for one crew member" .
```

Resource Datasource

The example application of the API uses a hardcoded datasource located in a CSV file with some resources.

The operation to create/read/update/delete use the hardcoded location of the file.

Some examples:



```
def get_requirement(base_url, specification_id):
    path = 'examples/specifications.csv'
    if os.path.isfile(path):

def get_requirement_list(base_url, select, where):
    requirements = list()
    path = 'examples/specifications.csv'
    if os.path.isfile(path):

    if specification:
        path = 'examples/specifications.csv'
        tempfile = NamedTemporaryFile(mode='w', delete=False)
```

SDK for Resource

Explanation

The Resource endpoint will manage the information over the requested artifact.

API

app.api.adapter.namespace.
core.py

API:
ResourceOperation /
post method

ResourcePreview /
get, put, delete method

app.api.adapter.namespaces.
business.py

API:
create_requirement method
update_requirement method
delete_requirement method
get_requirement method

SDK

pyoslc.resources.
models.py

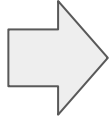
SDK:
Requirement

class ***Requirement***(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
elaborated_by=None, elaborates=None,
specified_by=None, specifies=None,
affected_by=None, tracked_by=None,
implemented_by=None, validated_by=None,
satisfied_by=None, satisfies=None,
decomposed_by=None, decomposes=None,
constrained_by=None, constrains=None)
```

Definition of Resource (GET Requirement)

API defining
specific
endpoint

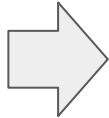


app.api.adapter.namespace.
core.py



```
def get(self, service_provider_id, requirement_id):  
    ...  
    requirement = get_requirement(base_url, requirement_id)
```

There are
methods to
create, get,
update and delete
resources
from CSV file

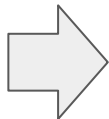


app.api.adapter.namespaces.
business.py



```
def get_requirement(base_url, specification_id):  
    path = 'examples/specifications.csv'  
    if os.path.isfile(path):  
        with open(path, 'r') as f:  
            reader = csv.DictReader(f, delimiter=';')  
            for row in reader:  
                if row['Specification_id'] == specification_id:  
                    about = base_url.replace('selector', 'requirement')  
                    requirement = Requirement(about=about)  
                    requirement.update(row, attributes)  
            return requirement
```

Returning the
Requirement on
RDF format



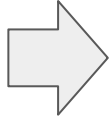
app.api.adapter.namespace.
core.py



```
def get(self, service_provider_id):  
    ...  
    requirement = get_requirement(base_url, requirement_id)  
    if requirement:  
        requirement.about = base_url  
        requirement.to_rdf(self.graph, base_url, attributes)
```

Definition of Resource (PUT Requirement)

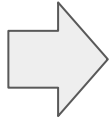
API defining
specific
endpoint



app.api.adapter.namespace.
core.py



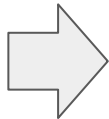
Updating
Requirement



app.api.adapter.namespaces.
business.py



Returning the
Requirement on
RDF format



app.api.adapter.namespace.
core.py

```
def put(self, service_provider_id, requirement_id):
    ...
    etag = request.headers.get(key='If-Match', default=None, type=str)
    elif not etag:
        raise BadRequest()
    else:
        dig = r.digestion()
        if dig != etag.strip("\n"):
            raise PreconditionFailed()
    ...
    req = update_requirement(requirement_id, data)
```

```
if specification:
    path = 'examples/specifications.csv'
    tempfile = NamedTemporaryFile(mode='w', delete=False)

    with open(path, 'r') as f:
        ...

    modified = False
    with open(path, 'r') as csvfile, tempfile:
        reader = csv.DictReader(csvfile, fieldnames=field_names, delimiter=';')
        writer = csv.DictWriter(tempfile, fieldnames=field_names, delimiter=';')
        for row in reader:
            if row['Specification_id'] == str(requirement_id):
                rq = Requirement()
                rq.from_json(specification, attributes=attributes)
                row = rq.to_mapped_object(attributes=attributes)
                row['Specification_id'] = requirement_id
                modified = True
            writer.writerow(row)
```

```
def put(self, service_provider_id):
    ...
    req = update_requirement(requirement_id, data)
    if isinstance(req, Requirement):
        req.to_rdf(self.graph, base_url, attributes)
    ...
```

RDF Generation of Resource (e.g. Requirement)

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationError", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

domains/rm.py / OSLC Resource class

```
14 class Requirement(BaseResource):
15
16     def __init__(self, about=None, types=None,
17                 title=None, contributor=None,
18                 discussed_by=None, instance_shape=None,
19                 elaborates=None, specified_by=None,
20                 implemented_by=None, validated_by=None,
21                 decomposes=None, constrained_by=None,
```

specification.py / Mapping of CSV to RDF attributes

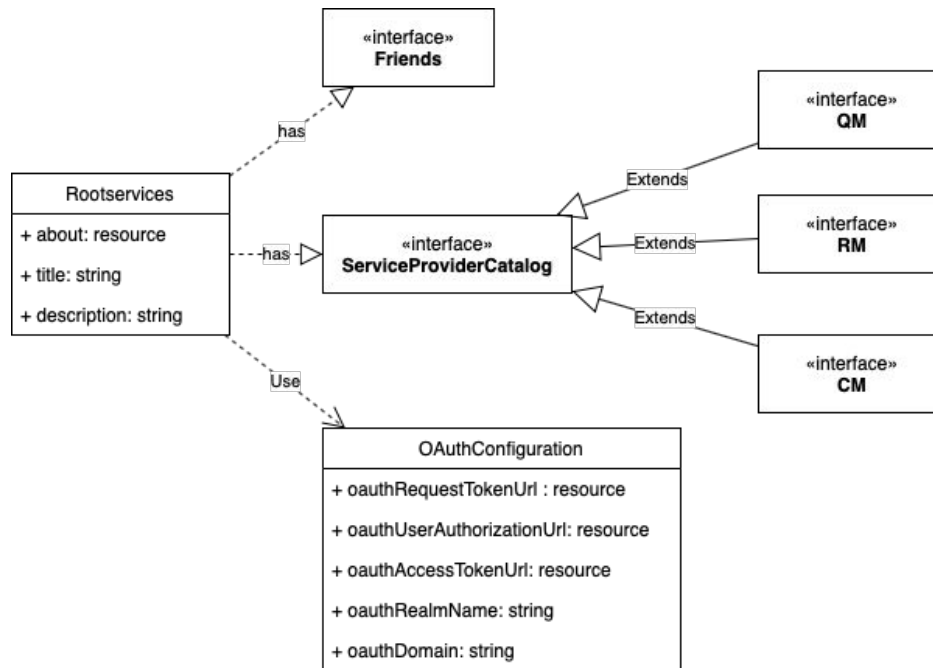
```
1
2 specification_map = {
3     # RDF and OSLC attributes
4     'Specification_id': {'attribute': '_BaseResource_id'},
5     'Title': {'attribute': '_BaseResource_title'},
6     'Description': {'attribute': '_BaseResource_description'},
7     'Author': {'attribute': '_BaseResource_creator'},
```

```
59 def to_rdf(self, graph, base_url=None, attributes=None):
60     assert attributes is not None, 'The mapping for attributes is required'
61
62     graph.bind('oslc_rm', OSLC_RM)
63
64     d = Describer(graph, base=base_url)
65     identifier = getattr(self, '_BaseResource__identifier')
66     if isinstance(identifier, Literal):
67         identifier = identifier.value
68     if identifier not in base_url.split('/'):
69         base_url = self.get_absolute_url(base_url, identifier)
```

Mapping from Python object type to RDF using the attributes dictionary taken from the mapping file.

JAZZ Integration / Interop

OSLC Rootservices Resource



[Friends Spec](#)

[Rootservices Spec](#)

Configuration Enabled

Any application that wants to interact with JAZZ should have the Configuration Enabled to be open to the interaction.

PyOSLC hardcoded the attributes that makes the application Configuration Enabled.



```
sp = Resource(graph, URIRef(uri))
sp.add(RDF.type, OSLC.ServiceProvider)

...

sp.add(JAZZ_PROCESS.supportContributionsToLinkIndexProvider, Literal(True, datatype=XSD.boolean))
sp.add(JAZZ_PROCESS.supportLinkDiscoveryViaLinkIndexProvider, Literal(True, datatype=XSD.boolean))
sp.add(JAZZ_PROCESS.supportOSLCSimpleQuery, Literal(True, datatype=XSD.boolean))
sp.add(JAZZ_PROCESS.globalConfigurationAware, Literal('yes', datatype=XSD.String))
```

The ServiceProvider is configured to be discoverable for any JAZZ application using **Scenario 1** of the [follow document from IBM](#).

Rootservices Document

Rootservices resource describes the ServiceProvider as containers to interact with JAZZ

GET <http://localhost:5000/oslc/services/rootservices>

```
@prefix jfs: <http://jazz.net/xmlns/prod/jazz/jfs/1.0/> .
@prefix oslc: <http://open-services.net/ns/core#> .
@prefix oslc_am: <http://open-services.net/ns/am#> .
@prefix oslc_config: <http://open-services.net/ns/config#> .
@prefix oslc_rm: <http://open-services.net/xmlns/rm/1.0/> .
@prefix pyoslc: <http://example.com/ns/pyoslc#> .
@prefix trs: <http://open-services.net/ns/core/trs#> .

<http://localhost:5000/oslc/services/rootservices> pyoslc:TrackedResourceSetProvider [ a pyoslc:TrackedResourceSetProvider ;
    trs:TrackedResourceSet [ a trs:TrackedResourceSet ;
        oslc:domain oslc_am,
            <http://open-services.net/ns/rm#> ;
        trs:trackedResourceSet <http://localhost:5000/oslc/> ;
        dc:description "Description" ;
        dc:title "Title" ;
        dc:type <http://open-services.net/ns/cm#> ] ] ;
jfs:oauthAccessTokenUrl <http://localhost:5000/token> ;
jfs:oauthApprovalModuleUrl <http://localhost:5000/approve> ;
jfs:oauthDomain "http://localhost:5000/oslc/" ;
jfs:oauthRealmName "PyOSLC" ;
jfs:oauthRequestConsumerKeyUrl <http://localhost:5000/register> ;
jfs:oauthRequestTokenUrl <http://localhost:5000/token> ;
jfs:oauthUserAuthorizationUrl <http://localhost:5000/token> ;
oslc_config:cmServiceProviders <http://localhost:5000/oslc/services/config> ;
oslc:publisher <http://localhost:5000/oslc/services/config/publisher> ;
oslc_rm:rmServiceProviders <http://localhost:5000/oslc/services/catalog> ;
dc:description "Services available on the PyOSLC." ;
dc:title "Root services for connecting with Jazz" .
```

SDK for Rootservices

Explanation

The Rootservice endpoint will generate all the elements required to start the integration with any JAZZ application.

API

app.api.adapter.namespace.
core.py

API:
**RootServices /
get** method

pyoslc.resources.
jazz.py

API:
to_rdf method

SDK

pyoslc.resources.
jazz.py

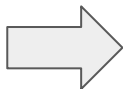
SDK:
RootService

class **RootService**(BaseResource):

```
def __init__(self, about=None, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
friends=None, publisher=None, service=None,
details=None, prefix_definition=None,
oauth_configuration=None)
```

Definition of Authentication

The OAuthConfiguration endpoints are hardcoded on the rootservices



```
rs.add(JFS.oauthRealmName, Literal("PyOSLC"))
rs.add(JFS.oauthDomain, Literal(url_for('oslc.doc', _external=True)))
rs.add(JFS.oauthRequestConsumerKeyUrl, URIRef(url_for('consumer.register', _external=True)))
rs.add(JFS.oauthApprovalModuleUrl, URIRef(url_for('consumer.approve', _external=True)))
rs.add(JFS.oauthRequestTokenUrl, URIRef(url_for('oauth.issue_token', _external=True)))
rs.add(JFS.oauthUserAuthorizationUrl, URIRef(url_for('oauth.issue_token', _external=True)))
rs.add(JFS.oauthAccessTokenUrl, URIRef(url_for('oauth.issue_token', _external=True)))
```

```
jfs:oauthAccessTokenUrl <http://localhost:5000/token> ;
jfs:oauthApprovalModuleUrl <http://localhost:5000/approve> ;
jfs:oauthDomain "http://localhost:5000/oslc/" ;
jfs:oauthRealmName "PyOSLC" ;
jfs:oauthRequestConsumerKeyUrl <http://localhost:5000/register> ;
jfs:oauthRequestTokenUrl <http://localhost:5000/token> ;
jfs:oauthUserAuthorizationUrl <http://localhost:5000/token> ;
```



pyoslc_oauth / consumers

RDF Generation of Rootservices

jazz.py / JAZZ Process Vocab

```
from rdflib import URIRef
from rdflib.namespace import ClosedNamespace

JFS = ClosedNamespace(
    uri=URIRef("http://jazz.net/xmlns/prod/jazz/ifs/1.0/"),
    terms=[
        # RDFS Classes in this namespace
        # RDF Properties in this namespace
        # for OAuth
        "oauthRealmName", "oauthDomain",
        "oauthRequestConsumerKeyUrl",
```

Resource.py / API Resource / Namespace Prefix Initialization

```
15 class OslcResource(Resource):
16
17     def __init__(self, *args, **kwargs):
18         super(OslcResource, self).__init__(*args, **kwargs)
19
20         self.graph = kwargs.get('graph', Graph())
21         self.graph.bind('oslc', OSLC)
22         self.graph.bind('rdf', RDF)
23         self.graph.bind('rdfs', RDFS)
24         self.graph.bind('dcterms', DCTERMS)
25         self.graph.bind('j.0', JAZZ_PROCESS)
```

resources/jazz.py / Rootservices class

```
class RootService(BaseResource):
    def __init__(self, about=None, types=None, properties=None, description=None, identifier=None, title=None, contributor=None, creator=None, subject=None, created=None, discussed_by=None, instance_shape=None, service_provider=None, relation=None, publisher=None, service=None, details=None, prefix_definition=None, out):
        """
        Initialize Root Service
        """
        super(RootService, self).__init__(about, types, properties, description, identifier, contributor, creator, subject, created, modified, instance_shape, service_provider, relation)
```

to_rdf() Python object
to RDF translation

```
def to_rdf(self, graph):
    graph.bind('dc', DCTERMS)
    graph.bind('jd', JAZZ_DISCOVERY)
    uri = self.about if self.about.__contains__(self.identifier) else self.about + '/{}'.format(
        self.identifier) if self.identifier else ''
    rs = Resource(graph, URIRef(uri))
    if self.title:
        rs.add(DCTERMS.title, Literal(self.title))
```

Binding JAZZ vocabulary
with specific prefixes

Mapping from Python
object attributes to RDF

OAuth Configuration

OSLC OAuthConfiguration Resource

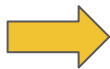
OAuthConfiguration
+ oauthRequestTokenUrl : resource
+ oauthUserAuthorizationUrl: resource
+ oauthAccessTokenUrl: resource
+ oauthRealmName: string
+ oauthDomain: string

OAuthConfiguration Endpoints

OAuth endpoints are available through the Rootservices response to create consumer keys.

GET <http://localhost:5000/oslc/services/rootservices>

Endpoints used when
adding the PyOSLC API
to JAZZ as a friend



```
@prefix jfs: <http://jazz.net/xmlns/prod/jazz/jfs/1.0/> .
@prefix oslc: <http://open-services.net/ns/core#> .
@prefix oslc_am: <http://open-services.net/ns/am#> .
@prefix oslc_config: <http://open-services.net/ns/config#> .
@prefix oslc_rm: <http://open-services.net/xmlns/rm/1.0/> .
@prefix pyoslc: <http://example.com/ns/pyoslc#> .
@prefix trs: <http://open-services.net/ns/core/trs#> .

<http://localhost:5000/oslc/services/rootservices> pyoslc:TrackedResourceSetProvider [ a pyoslc:TrackedResourceSetProvider ;
    trs:TrackedResourceSet [ a trs:TrackedResourceSet ;
        oslc:domain oslc_am;
        <http://open-services.net/ns/rm#> ;
        trs:trackedResourceSet <http://localhost:5000/oslc/> ;
        dc:description "Description" ;
        dc:title "Title" ;
        dc:type <http://open-services.net/ns/rm#> " " .
        jfs:oauthAccessTokenUrl <http://localhost:5000/token> ;
        jfs:oauthApprovalModuleUrl <http://localhost:5000/approve> ;
        jfs:oauthDomain "http://localhost:5000/oslc/" ;
        jfs:oauthRealmName "PyOSLC" ;
        jfs:oauthRequestConsumerKeyUrl <http://localhost:5000/register> ;
        jfs:oauthRequestTokenUrl <http://localhost:5000/token> ;
        jfs:oauthUserAuthorizationUrl <http://localhost:5000/token> ;
    ] ;
    oslc_config:cmServiceProviders <http://localhost:5000/oslc/services/config> ;
    oslc:publisher <http://localhost:5000/oslc/services/config/publisher> ;
    oslc_rm:rmServiceProviders <http://localhost:5000/oslc/services/catalog> ;
    dc:description "Services available on the PyOSLC." ;
    dc:title "Root services for connecting with Jazz" .
```

SDK for OAuthConfiguration

Explanation

The current implementation of OAuthConfiguration is only used on the Rootservices as a hardcoded implementation.



API

API:
NA

Representation of OAuthConfiguration described in the representation of Rootservices



SDK

pyoslc.resources.
jazz.py

SDK:
OAuthConfiguration

class **OAuthConfiguration**(BaseResource):

```
def __init__(self, about, types=None,
properties=None, description=None,
identifier=None, short_title=None, title=None,
contributor=None, creator=None, subject=None,
created=None, modified=None, type=None,
discussed_by=None, instance_shape=None,
service_provider=None, relation=None,
authorization_uri=None,
oauth_access_token_uri=None,
oauth_request_token_uri=None)
```

RDF Generation of OAuthConfiguration

Core.py / OSLC Core Vocab

```
9 from rdflib import URIRef
10 from rdflib.namespace import ClosedNamespace
11
12 OSLC = ClosedNamespace(
13     uri=URIRef("http://open-services.net/ns/core#"),
14     terms=[
15         # RDFS Classes in this namespace
16         "AllowedValues", "AttachmentContainer", "AttachmentDescriptor",
17         "Comment", "Compact", "CreationFactory", "Dialog",
18         "Discussion", "Error", "ExtendedError", "OAuthConfiguration",
19         "PrefixDefinition", "Preview", "Property", "Publisher",
20         "QueryCapability", "ResourceShape", "ResponseInfo",
21         "Service", "ServiceProvider", "ServiceProviderCatalog",
```



```
graph.bind('oslc_am', OSLC_AM)
graph.bind('oslc_rm', OSLC_RM_JAZZ, override=True)
graph.bind('oslc_cm', OSLC_CM_JAZZ, override=True)
graph.bind('oslc_config', OSLC_CONFIG)
graph.bind('jfs', JFS)
graph.bind('trs', OSLC_TRS)
graph.bind('pyoslc', PYOSLC)
```

jazz.py / JAZZ Process Vocab

```
from rdflib import URIRef
from rdflib.namespace import ClosedNamespace

JFS = ClosedNamespace(
    uri=URIRef("http://jazz.net/xmlns/prod/jazz/jfs/1.0/"),
    terms=[
        # RDFS Classes in this namespace

        # RDF Properties in this namespace
        # for OAuth
        "oauthRealmName", "oauthDomain",
        "oauthRequestConsumerKeyUrl",
```

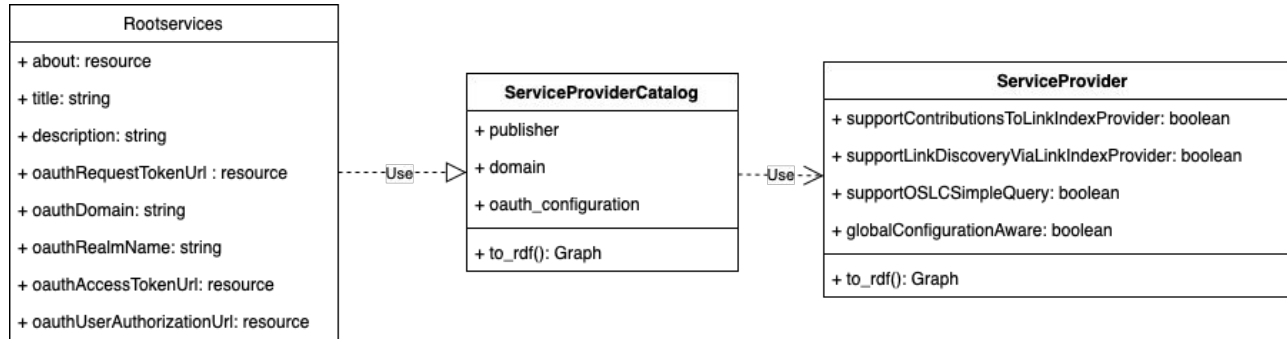


```
rs.add(JFS.oauthRealmName, Literal("PyOSLC"))
rs.add(JFS.oauthDomain, Literal(url_for('oslc.doc', _external=True)))
rs.add(JFS.oauthRequestConsumerKeyUrl, URIRef(url_for('consumer.register', _external=True)))
rs.add(JFS.oauthApprovalModuleUrl, URIRef(url_for('consumer.approve', _external=True)))
rs.add(JFS.oauthRequestTokenUrl, URIRef(url_for('oauth.issue_token', _external=True)))
rs.add(JFS.oauthUserAuthorizationUrl, URIRef(url_for('oauth.issue_token', _external=True)))
rs.add(JFS.oauthAccessTokenUrl, URIRef(url_for('oauth.issue_token', _external=True)))
```

The RDF generation occurs on the Rootservices **to_rdf** method since this is only used on the Rootservices document until now.

Friending / Associating / Linking With IBM Jazz

OSLC Resources for Friending/Associating



Discoverable endpoints

Rootservices exposes the Providers that could be used for Friending.

GET <http://localhost:5000/oslc/services/rootservices>

Authentication endpoints to register a Friend or Consumer



```
@prefix jfs: <http://jazz.net/xmlns/prod/jazz/jfs/1.0/> .
@prefix oslc: <http://open-services.net/ns/core#> .
@prefix oslc_am: <http://open-services.net/ns/am#> .
@prefix oslc_config: <http://open-services.net/ns/config#> .
@prefix oslc_rm: <http://open-services.net/xmlns/rm/1.0/> .
@prefix pyoslc: <http://example.com/ns/pyoslc#> .
@prefix trs: <http://open-services.net/ns/core/trs#> .

<http://localhost:5000/oslc/services/rootservices> pyoslc:TrackedResourceSetProvider [ a pyoslc:TrackedResourceSetProvider ;
  trs:TrackedResourceSet [ a trs:TrackedResourceSet ;
    oslc:domain oslc_am;
    <http://open-services.net/ns/rm#> ;
    trs:trackedResourceSet <http://localhost:5000/oslc/> ;
    dc:description "Description" ;
    dc:title "Title" ;
    dc:type <http://open-services.net/ns/rm#> " " ;
    jfs:oauthAccessTokenUrl <http://localhost:5000/token> ;
    jfs:oauthApprovalModuleUrl <http://localhost:5000/approve> ;
    jfs:oauthDomain "http://localhost:5000/oslc/" ;
    jfs:oauthRealmName "PyOSLC" ;
    jfs:oauthRequestConsumerKeyUrl <http://localhost:5000/register> ;
    jfs:oauthRequestTokenUrl <http://localhost:5000/token> ;
    jfs:oauthUserAuthorizationUrl <http://localhost:5000/token> ;
    oslc_config:cmServiceProviders <http://localhost:5000/oslc/services/config> ;
    oslc:publishes <http://localhost:5000/oslc/services/config/publishes> ;
    oslc_rm:rmServiceProviders <http://localhost:5000/oslc/services/catalog> ;
    dc:description "Services available on the PyOSLC." ;
    dc:title "Root services for connecting with" ;
```



Provider to use for Friending

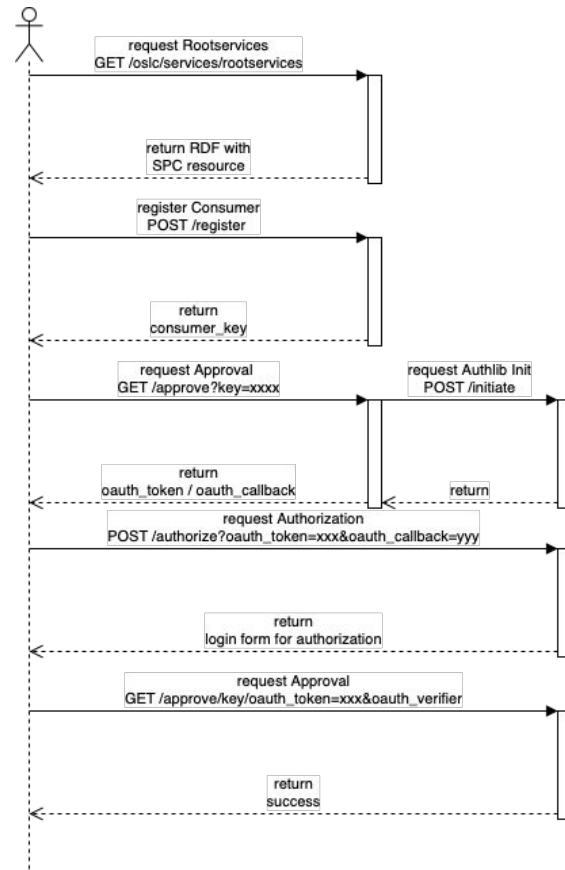
Friending

IBM / JAZZ requests Rootservices

Consumer is registered on a
FileSystemConsumerStore service using
SQLite as a database.

PyOSLC uses Authlib to run a AuthServer to
authenticate and validate the Consumer key.

After the approval the PyOSLC is registered as
a Friend on JAZZ



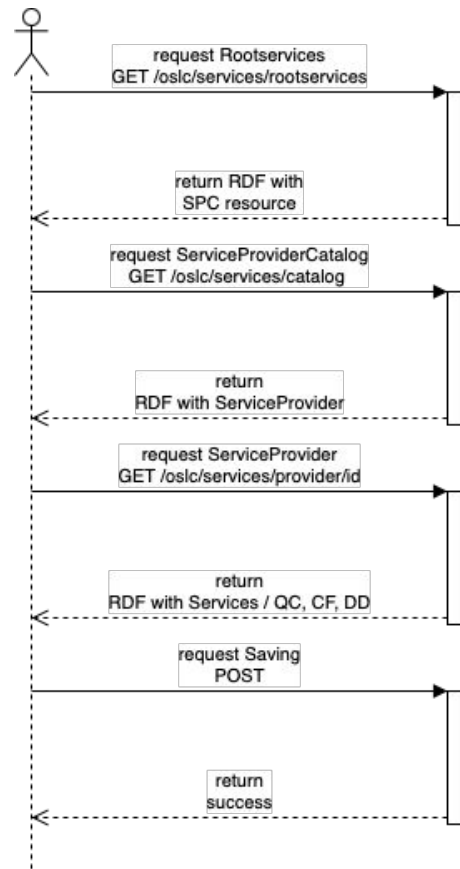
Associating

IBM / JAZZ requests Rootservices

IBM / JAZZ requests ServiceProviderCatalog
which define the Containers

IBM / JAZZ requests ServiceProvider that should
have ConfigurationEnabled feature

Association should work only for RM application
since the domain is hardcoded

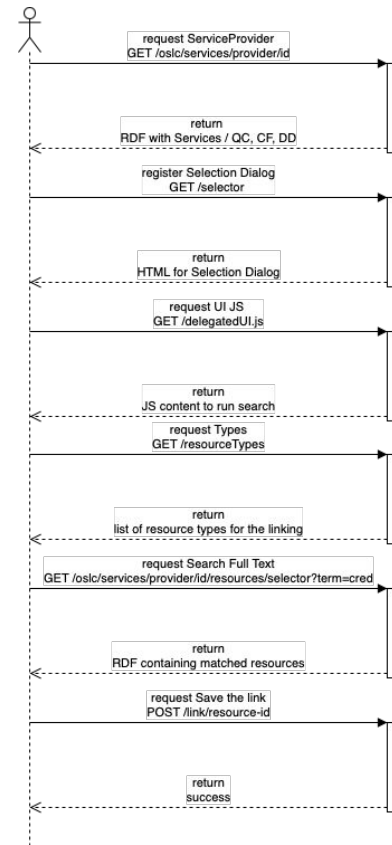


Linking

IBM / JAZZ requests ServiceProvider

Base on the Dialog url and the Javascript code, shows the Selection Dialog

Provide the text to run the search over the API using the QC endpoint and getting the resources on RDF



Previews

IBM / JAZZ requests resource information using configuration context

Icon could be used to identify resources for the OSLC API

IBM / JAZZ requests resource details for small preview with minimum list of data returned on HTML.

Large Preview could contain more or all the attributes of the resource.

