

Appendix 1b. Kuala Lumpur 2018 Catalyst Project
Zero Touch Marketplace Integration
Extension to TMForum Assets

Addition of Engaged Party Specification

Zero Touch Marketplace Integration (ZTM) solution starts by registering a new SUPPLIER/PARTNER in BUYER system. All information of PARTNER required during design-time and run-time journey (such as Endpoints and authentication credentials/certificates) are captured in this registration process. In our solution, we introduced the concept of a partner (using Configuration type of Specification) within the BUYER system where the register entities are specification instances.

Following is the Partner Registry Specification. It has *partnerName* and *endpoints* as specCharacteristics. *endpoints* is a Collection type specCharacteristics, with child specCharacteristics as follow:

- *endpointName*,
- *endpointURL*,
- *interfaceSpecification*, Enumerated specCharacteristic with options of interface specifications to be used during integration, such as TMF API v3.0,
- *configuration*, String KeyValue specCharacteristic where we can put any configuration information required during integration.

```
{
  "id": "ZTM-SPC-CFG-0001",
  "name": "Partner Registry",
  "lifecycleStatus": "PUBLISHED",
  "version": "1.0",
  "@type": "CONFIGURATION",
  "specCharacteristic": [{
    "id": "partnerName",
    "readOnly": false,
    "displayFormat": "",
    "publicKey": "",
    "name": "Partner Name",
    "description": "",
    "@type": "KEY_VALUE",
    "minCardinality": 0,
    "maxCardinality": 1,
    "regex": "",
    "regexDescription": "",
    "specCharRelationship": [],
    "specCharacteristicValue": [{
      "tags": [],
      "value": "",
      "valueType": "STRING",
      "isDefault": false
    }],
    "tags": []
  }],
  {
    "id": "endpoints",
    "readOnly": false,
    "displayFormat": "",
    "publicKey": "",
    "name": "Endpoints",
    "description": "",
    "@type": "COLLECTION",
    "minCardinality": 1,
    "maxCardinality": 5,
    "regex": "",
    "regexDescription": "",
    "specCharRelationship": [{
      "id": "endpointName",
```

```

    "readOnly": false,
    "displayFormat": "",
    "publicKey": "",
    "name": "Endpoint Name",
    "description": "",
    "@type": "ENUMERATED",
    "minCardinality": 0,
    "maxCardinality": 1,
    "regex": "",
    "regexDescription": "",
    "specCharRelationship": [],
    "specCharacteristicValue": [{
      "tags": [],
      "value": "ProductQualification",
      "valueType": "STRING",
      "isDefault": false
    },
    {
      "tags": [],
      "value": "ProductOrder",
      "valueType": "STRING",
      "isDefault": false
    }
  ],
  "tags": []
},
{
  "id": "endpointURL",
  "readOnly": false,
  "displayFormat": "",
  "publicKey": "",
  "name": "Endpoint URL",
  "description": "",
  "@type": "KEY_VALUE",
  "minCardinality": 0,
  "maxCardinality": 1,
  "regex": "",
  "regexDescription": "",
  "specCharRelationship": [],
  "specCharacteristicValue": [{
    "tags": [],
    "value": "",
    "valueType": "STRING",
    "isDefault": false
  }],
  "tags": []
},
{
  "id": "interfaceSpecification",
  "readOnly": false,
  "displayFormat": "",
  "publicKey": "",
  "name": "Interface Specification",
  "description": "",
  "@type": "ENUMERATED",
  "minCardinality": 0,
  "maxCardinality": 1,
  "regex": "",
  "regexDescription": "",
  "specCharRelationship": [],
  "specCharacteristicValue": [{
    "tags": [],
    "value": "TMF API v3.0",
    "valueType": "STRING",
    "isDefault": false
  },
  {
    "tags": [],
    "value": "Telflow API v2.0",
    "valueType": "STRING",
    "isDefault": false
  }
  ],
  "tags": []
},
{
  "id": "configuration",
  "readOnly": false,
  "displayFormat": "TextArea",
  "publicKey": "",
  "name": "Configuration",
  "description": "",
  "@type": "KEY VALUE",

```

```

    "minCardinality": 0,
    "maxCardinality": 1,
    "regex": "",
    "regexDescription": "",
    "specCharRelationship": [],
    "specCharacteristicValue": [{
      "tags": [],
      "defaultValue": "",
      "value": "",
      "valueType": "STRING",
      "isDefault": false
    }],
    "tags": []
  }],
  "specCharacteristicValue": [],
  "tags": []
}
}
}

```

Following is an example of the Specification instance for Sinefa as one of the partner.

```

{
  "id": "ZTM-CON-PAR-0001",
  "name": "Sinefa",
  "version": "5",
  "describedBy": {
    "characteristicValue": [{
      "value": "Sinefa",
      "characteristic": {
        "id": "partnerName",
        "name": "Partner Name",
        "description": "",
        "type": "KeyValue",
        "readOnly": false,
        "minCardinality": 0,
        "maxCardinality": 1,
        "displayFormat": "",
        "regex": "",
        "regexDescription": "",
        "publicKey": ""
      }
    }],
    "specCharValueInstantiatedAs": {
      "valueType": "String"
    }
  },
  {
    "value": "",
    "characteristic": {
      "id": "endpoints",
      "name": "Endpoints",
      "description": "",
      "type": "Collection",
      "readOnly": false,
      "minCardinality": 1,
      "maxCardinality": 5,
      "displayFormat": "",
      "regex": "",
      "regexDescription": "",
      "publicKey": ""
    }
  },
  "characteristicValueReferences": [{
    "value": "ProductQualification",
    "characteristic": {
      "id": "endpointName",
      "name": "Endpoint Name",
      "description": "",
      "type": "Enumerated",
      "readOnly": false,
      "minCardinality": 0,
      "maxCardinality": 1,
      "displayFormat": "",
      "regex": "",
      "regexDescription": "",
      "publicKey": ""
    }
  }],
  "specCharValueInstantiatedAs": {
    "valueType": "String"
  }
},
},
}

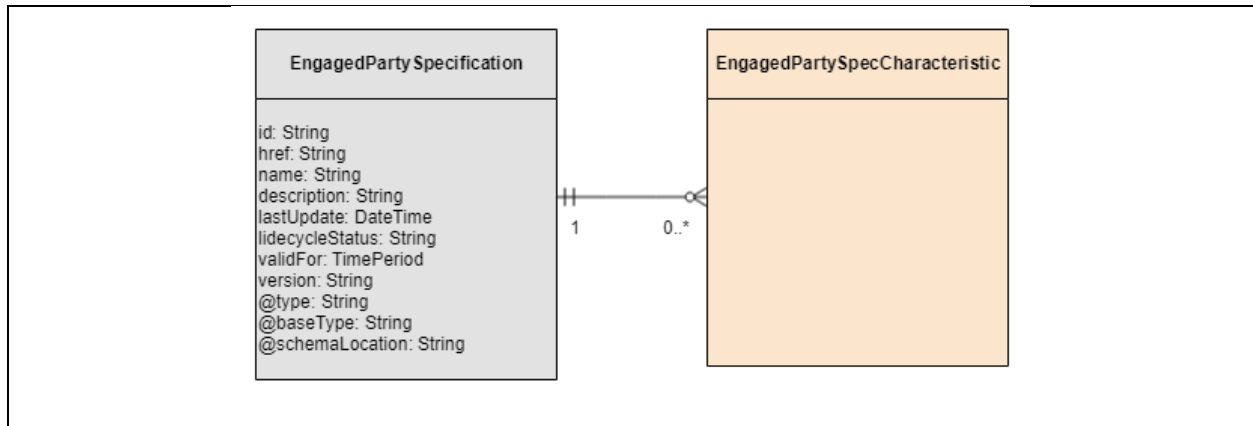
```

```

    {
      "value": "https://sinefa.demo.telflow.com/api/v2/productOfferingQualification",
      "characteristic": {
        "id": "endpointURL",
        "name": "Endpoint URL",
        "description": "",
        "type": "KeyValue",
        "readOnly": false,
        "minCardinality": 0,
        "maxCardinality": 1,
        "displayFormat": "",
        "regex": "",
        "regexDescription": "",
        "publicKey": ""
      },
      "specCharValueInstantiatedAs": {
        "valueType": "String"
      }
    },
    {
      "value": "TMF API v3.0",
      "characteristic": {
        "id": "interfaceSpecification",
        "name": "Interface Specification",
        "description": "",
        "type": "Enumerated",
        "readOnly": false,
        "minCardinality": 0,
        "maxCardinality": 1,
        "displayFormat": "",
        "regex": "",
        "regexDescription": "",
        "publicKey": ""
      },
      "specCharValueInstantiatedAs": {
        "valueType": "String"
      }
    },
    {
      "value": "{\n  \"token\": {\n    \"client_id\": \"telflow-portal\",\n    \"client_secret\": \"change-me\",\n    \"username\": \"globex_system\",\n    \"password\": \"globex_system\",\n    \"grant_type\": \"password\"\n  },\n  \"accountId\": \"0\"\n}",
      "characteristic": {
        "id": "configuration",
        "name": "Configuration",
        "description": "",
        "type": "KeyValue",
        "readOnly": false,
        "minCardinality": 0,
        "maxCardinality": 1,
        "displayFormat": "TextArea",
        "regex": "",
        "regexDescription": "",
        "publicKey": ""
      },
      "specCharValueInstantiatedAs": {
        "valueType": "String"
      }
    }
  ]
},
"specifiedBy": {
  "id": "ZTM-SPC-CFG-0001",
  "name": "Partner Registry",
  "version": "1.0"
},
"tags": [{
  "name": "sinefa"
}],
"class": "Config",
"status": "Active",
"creationDate": "2018-10-24T12:24:01.135Z",
"lastModifiedDate": "2018-11-07T06:12:22.271Z",
"_links": {
  "rel": "self",
  "href": "\/v1\/configurationentities\/ZTM-CON-PAR-0001"
}
}

```

To cover the needs of capturing the partner information mentioned above, our suggestion is to use “Engaged Party” type of Specification. “Engaged Party” Specification will have 1 to many relationship with SpecCharacteristic as other type of Specifications, as follow:

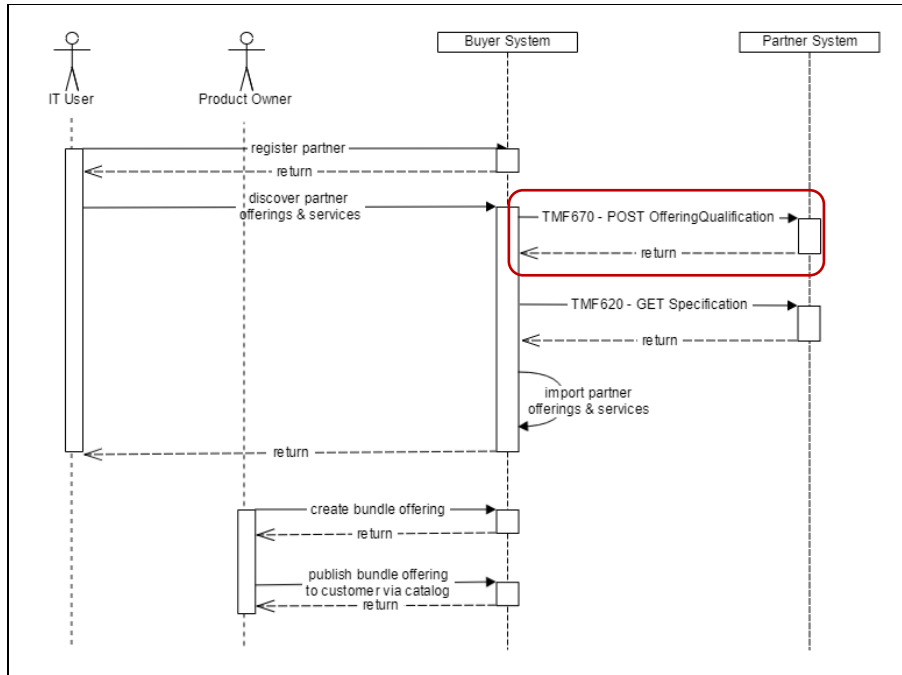


Following is an example of “Engaged Party” Specification, with simplified version (trimmed) SpecCharacteristics.

```
{
  "id": "ZTM-SPC-CFG-0001",
  "href": "https://portal.demo.telflow.com:9293/catalogManagement/engagedPartySpecification/ZTM-SPC-CFG-0001",
  "name": "Engaged Party",
  "lifecycleStatus": "PUBLISHED",
  "version": "1.0",
  "@type": "EngagedParty",
  "engagedPartySpecCharacteristic": [{
    "id": "partnerName",
    ... trimmed ...
  }],
  {
    "id": "endpoints",
    ... trimmed ...
  }
}]
}
```

Extension to TMF679 Offering Qualification

Once new PARTNER being registered into BUYER system, the next step is to discover the offerings and services in partner's catalog and import it into buyer's catalog. Following sequence diagram depicts the process of design-time journey of ZTM solution. The highlighted step (in red) shows that BUYER system sends TMF679 POST Offering Qualification to PARTNER system to learn the list of qualified offerings.



Design-Time Sequence Diagram

Existing TMF679 only supports asynchronous method where API consumer needs to send POST Offering Qualification with at least one qualification item to qualified, and receives a qualification id in the response. API consumer then can check the qualification status and get the qualification result by sending GET Offering Qualification by qualification id. In ZTM solution, we would like to be able to get an instant response to deliver a seamless outcome for automated cases. In addition to it as the buyer has no knowledge about the partner's offering, a wildcard Offering Qualification request is required for the solution.

To cover the needs mentioned above, following are the suggestion for extension of TMF679 Offering Qualification

1. Added a synchronous method to the API.
2. Added a wildcard Offering Request, that supports having no qualification item in the request and returns all qualified offerings in the response.

Following is an example of the extended request and the response.

Request

```
POST:https://host/api/v2/productOfferingQualification
```

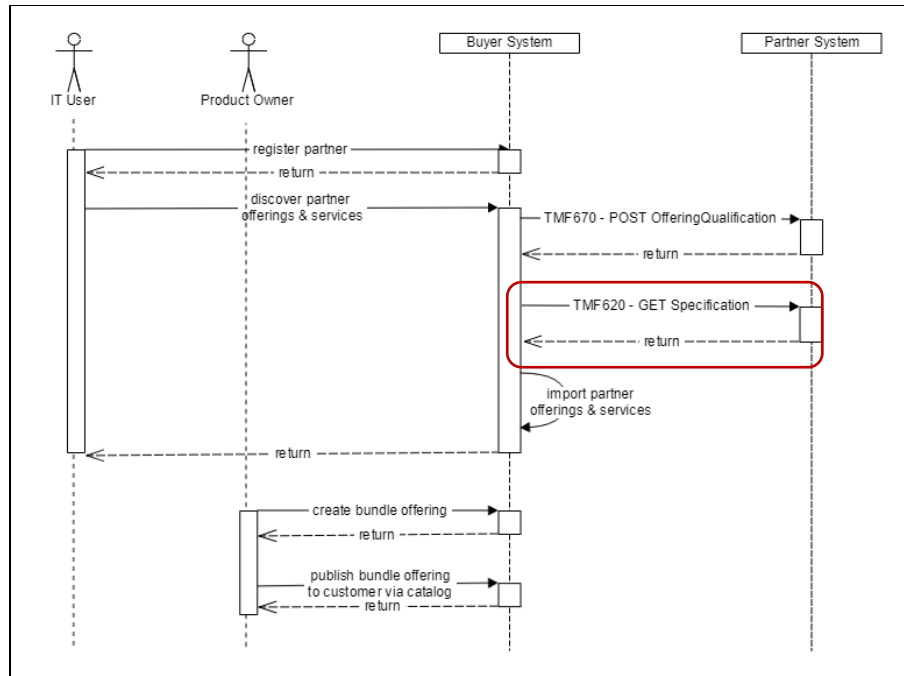
```
{
  "relatedParty": [
    {
      "id": "PTR000000000057"
    }
  ]
}
```

Response

```
{
  "@type": "ProductEligibility",
  "qualificationType": "ProductEligibility",
  "qualificationResult": "Serviceable",
  "productOfferingQualificationItem": [
    {
      "@type": "ServiceQualificationItem",
      "productOffering": {
        {
          "id": "ZTM-OFF-PRD-0003",
          "name": "Sinefa Probe",
          "validFor": null,
          "version": "1.0",
          "specifiedBy": {
            "id": "SPC000000000012"
          }
        },
        "type": "Product",
        "status": "Published",
        "specificationMadeAvailableAs": {
          "id": "ZTM-SPC-PRD-0003"
        }
      },
      "businessInteractionSpecificationRelationships": [
        {
          "type": "Managed",
          "businessInteractionSpecification": {
            "id": "ZTM-SPC-NEW-0003"
          }
        }
      ],
      "designStrategy": "None",
      "pricingStrategy": "None"
    }
  ]
}
```

Extension to TMF620 Product Catalog Management

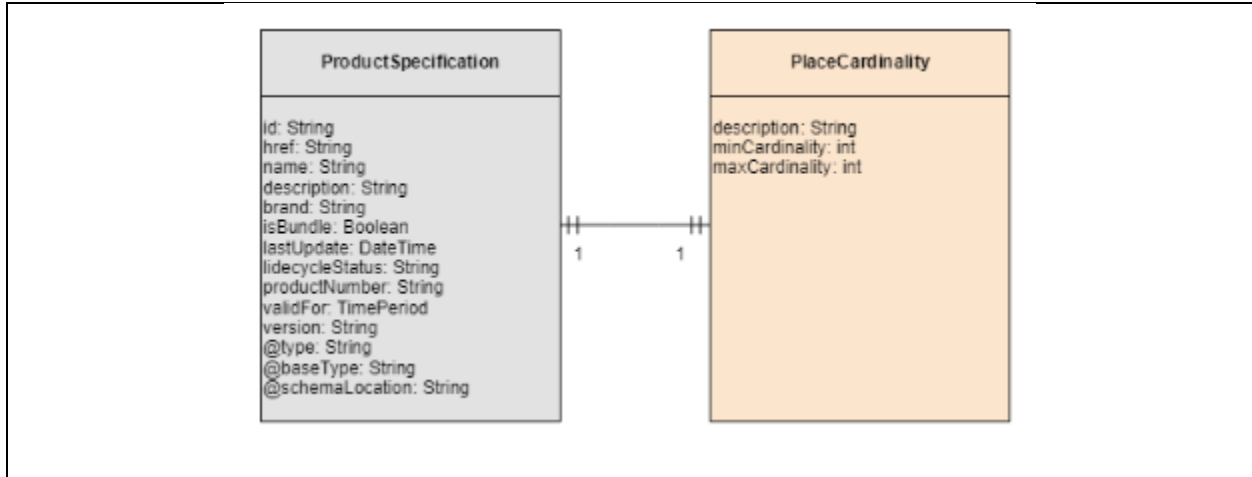
Prior ordering a product/service, BUYER needs to understand whether the service is address-dependent or not, and if yes, how many addresses that needs to be included during address/service qualification request. For example, for some complex products/services such as E-Line, it requires at least 2 addresses. This information can be retrieved during design-time. Following sequence diagram depicts the process of design-time journey of ZTM solution. The highlighted step (in red) shows that BUYER system sends TMF620 GET Specification to PARTNER system to get the Specification details and import in into their own system.



Design-Time Sequence Diagram

In existing TMF620, the Resource Model for ProductOffering supports relationship between ProductOffering and Place. This let us know which offerings are address-dependent and which are not, however this does not tell how many places required for an offering. In ZTM solution, we extend ProductSpecification resource model to cover number of required addresses information. Reason putting it into ProductSpecification instead of ProductOffering is that a ProductSpecification can be bundles into different ProductOfferings but the number of required addresses for a product will remain the same.

To cover the needs mentioned above, following is our suggestion for extension of ProductSpecification resource model. ProductSpecification will have one-to-one relationship with PlaceCardinalities. PlaceCardinalities resource will have at least *minCardinality* and *maxCardinality* attributes.



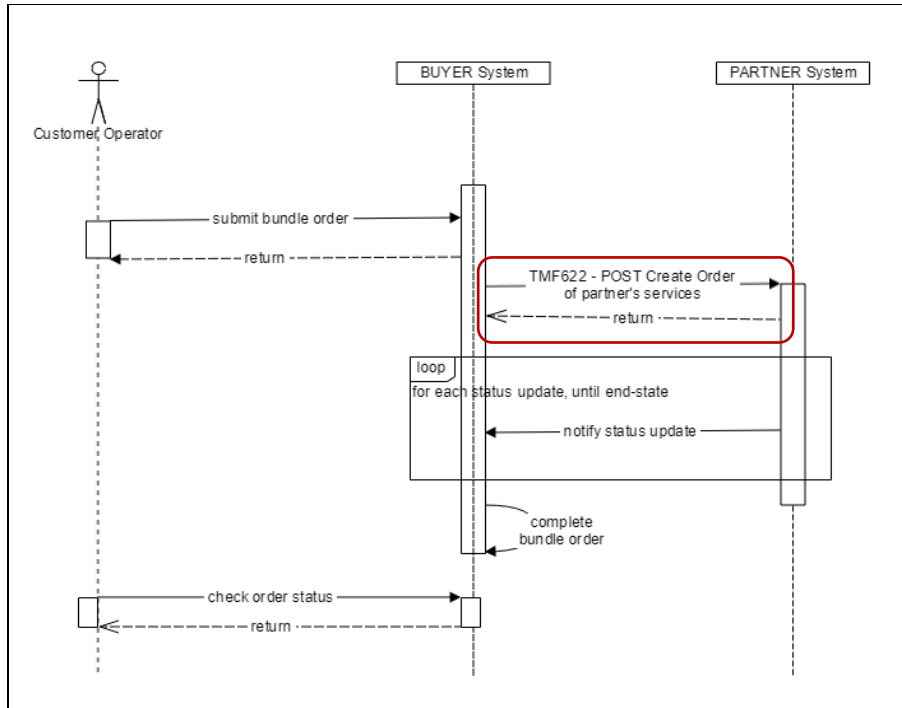
Following is an example of the extended ProductSpecification detail.

```

{
  "placeCardinalities": [
    {
      "type": "SERVICE_ACCESS_TO",
      "minCardinality": 0,
      "maxCardinality": 1
    }
  ],
  "id": "ZTM-SPC-PRD-0003",
  "href": "https://host/catalogManagement/productSpecification/ZTM-SPC-PRD-0003",
  "name": "Sinefa Probe",
  "isBundle": false,
  "lifecycleStatus": "PUBLISHED",
  "version": "1.0",
  "@type": "PRODUCT",
  "relatedParty": [],
  "productSpecCharacteristic": [
    ... trimmed ...
  ],
  "serviceSpecification": [],
  "productSpecificationRelationship": [],
  "resourceSpecification": [],
  "attachment": [],
  "bundledProductSpecification": [],
  "tags": [],
  "rateCodes": []
}
  
```

Extension to TMF622 Product Order Management

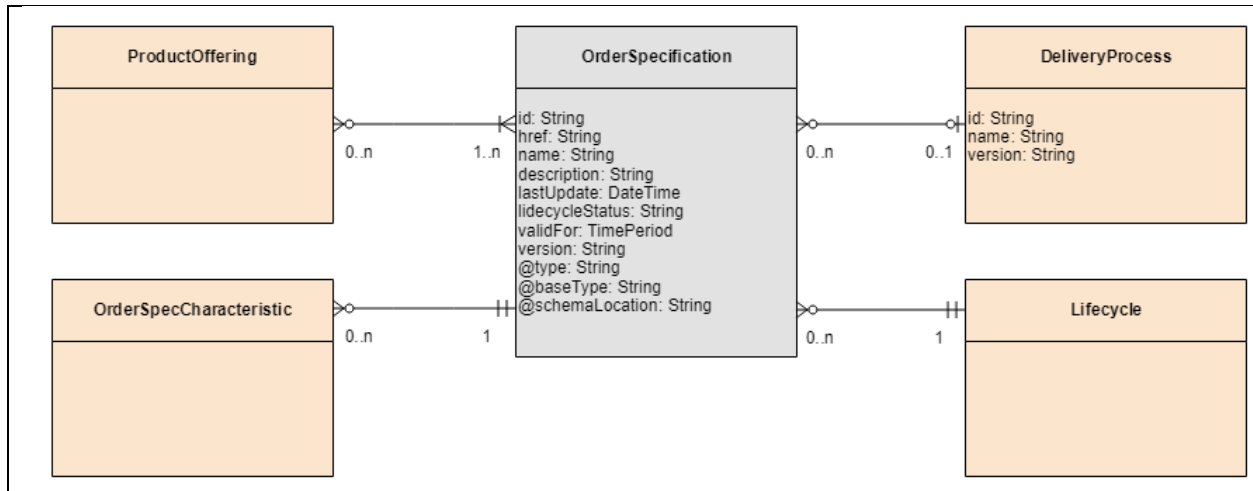
Run-time journey on ZTM solution starts with user login to BUYER system and submit a bundle order which may contain several partner's services. Following sequence diagram depicts the process of run-time journey of ZTM solution. The highlighted step (in red) shows that BUYER system sends TMF622 POST Create Order for each partner's service to multiple PARTNER systems.



Run-Time Sequence Diagram

To accommodate requirement of having characteristics which do not persist beyond the order lifecycle, we use Order Specification that also supports dynamically assigned state-lifecycle and process. In addition to it Order specifications provide structure and governance around different orders. Currently the TMForum API have unmanaged string to specify types of orders. With an order specification we can have structured order types like new, change, terminate, move, transfer etc.

Following is data definitions of Order Specification.



OrderSpecification may have it's own SpecCharacteristics, which will not persist beyond the order lifecycle. An Offering should have at least one OrderSpecification and may have multiple with the same type or different types, such as New, Change, Terminate, etc. OrderSpecification may have 1 delivery process, and should have one Lifecycle.

Following is an example of the OrderSpecification detail.

```

{
  "id": "ZTM-SPC-NEW-0003",
  "href": "https://host/catalogManagement/orderSpecification/ZTM-SPC-NEW-0003",
  "name": "New Sinefa Order",
  "lifecycleStatus": "PUBLISHED",
  "version": "1.0",
  "@type": "NEW_CUSTOMER_ORDER",
  "orderSpecCharacteristic": [
    ... trimmed ...
  ],
  "deliveryProcess": {
    "id": "catalyst-zero-touch-partner"
  },
  "lifecycle": {
    "id": "LXF00000000232"
  },
  "orderSpecificationRelationship": [],
  "tags": []
}

```