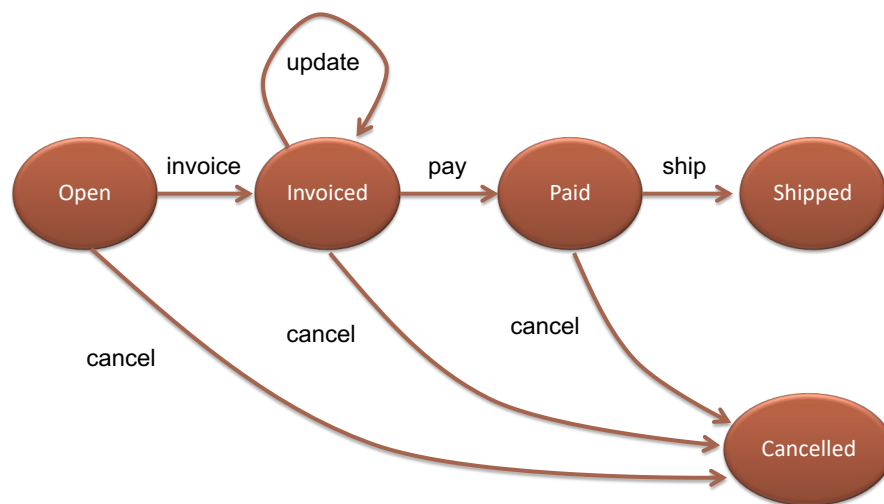


## DOMAIN APPLICATION PROTOCOL

65

65

## Protocol as State Machine



66

66

## Domain Application Protocol

- Express protocol via hypermedia
  - Links in responses
- Problem: JSON does not distinguish links!
- Solution 0: Use HTML or ATOM
- Solution 1: Web Linking
- Solution 2: Extend JSON with linking

67

67

## Web Linking

- RFC 8288 (replaces RFC 5988)
- Links in response headers
- For each link:
  - Context URI
  - Target URI
  - Semantic relationship: role in the workflow
  - Target attributes

68

68

## Web Linking

- Predefined target attributes (HTTP):
  - rel: relationship (required)
  - rev: reverse relationship
  - anchor: context
  - title:
  - type:

69

69

## Purchase Order Protocol: Invoice Representation

```
Link: <http://www.example.org/purchases/rest/invoice/12345>;  
  rel="http://www.example.org/purchases/rest/relations/pay";  
  type="application/vnd.purchases+json"
```

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
  rel="http://www.example.org/purchases/rest/relations/update";  
  type="application/vnd.purchases+json"
```

```
Link: <http://www.example.org/purchases/rest/purchase/67890>  
  rel="http://www.example.org/purchases/rest/relations/cancel"
```

```
{  
  "po:invoice-num" : 12345,  
  "po:purch-order-num" : 67890,  
  "po:amount" : 39.95  
}
```

70

70

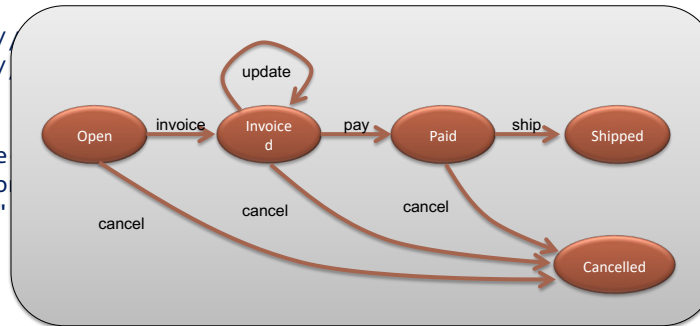
## Purchase Order Protocol: Invoice Representation

Link: <http://www.example.org/purchases/rest/invoice/12345>;  
rel="http://www.example.org/purchases/rest/relations/pay";  
type="application/vnd.purchases+json"

Link: <http://www.example.org/purchases/rest/purchase/67890>;  
rel="http://www.example.org/purchases/rest/relations/update";  
type="application/vnd.purchases+json"

Link: <http://  
rel="http://

```
{
  "po:invoice
  "po:purch-o
  "po:amount"
}
```



71

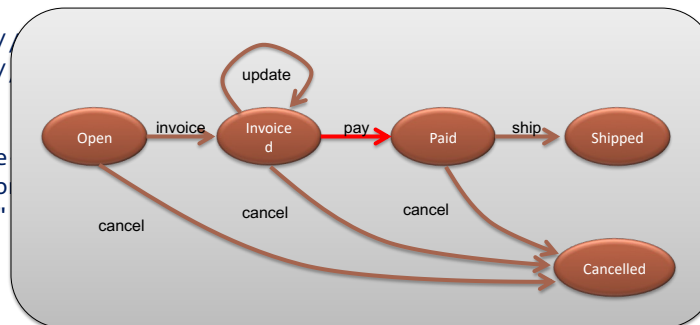
## Purchase Order Protocol: Invoice Representation

Link: <http://www.example.org/purchases/rest/invoice/12345>;  
rel="http://www.example.org/purchases/rest/relations/pay";  
type="application/vnd.purchases+json"

Link: <http://www.example.org/purchases/rest/purchase/67890>;  
rel="http://www.example.org/purchases/rest/relations/update";  
type="application/vnd.purchases+json"

Link: <http://  
rel="http://

```
{
  "po:invoice
  "po:purch-o
  "po:amount"
}
```



72

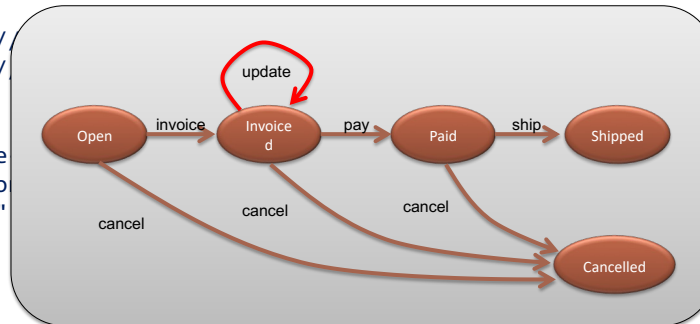
## Purchase Order Protocol: Invoice Representation

Link: <<http://www.example.org/purchases/rest/invoice/12345>>;  
rel="http://www.example.org/purchases/rest/relations/pay";  
type="application/vnd.purchases+json"

Link: <<http://www.example.org/purchases/rest/purchase/67890>>;  
rel="http://www.example.org/purchases/rest/relations/update";  
type="application/vnd.purchases+json"

Link: <<http://www.example.org/purchases/rest/invoice/12345>>;  
rel="http://www.example.org/purchases/rest/relations/pay";

```
{  
  "po:invoice":  
  "po:purch-o":  
  "po:amount":  
}
```



73

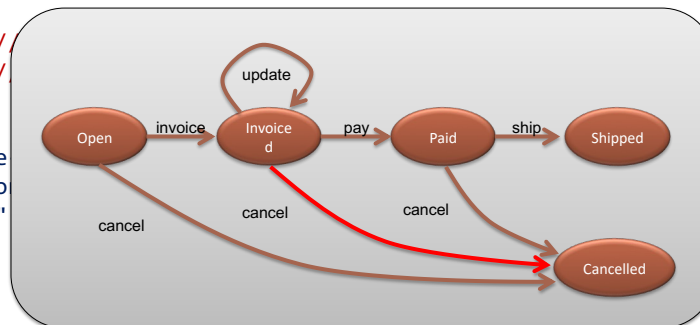
## Purchase Order Protocol: Invoice Representation

Link: <<http://www.example.org/purchases/rest/invoice/12345>>;  
rel="http://www.example.org/purchases/rest/relations/pay";  
type="application/vnd.purchases+json"

Link: <<http://www.example.org/purchases/rest/purchase/67890>>;  
rel="http://www.example.org/purchases/rest/relations/update";  
type="application/vnd.purchases+json"

Link: <<http://www.example.org/purchases/rest/invoice/12345>>;  
rel="http://www.example.org/purchases/rest/relations/pay";

```
{  
  "po:invoice":  
  "po:purch-o":  
  "po:amount":  
}
```



74

## Workflow Links

- Link to payment resource
  - **POST** to complete payment
- Link to purchase resource, rel="update"
  - **PATCH** to change purchase order
- Link to purchase resource, rel="cancel"
  - **DELETE** to cancel purchase

75

75

## Purchase Order Protocol: Payment Receipt

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
rel="http://www.example.org/purchases/rest/relations/ship";  
type="application/vnd.purchases+json"
```

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
rel="http://www.example.org/purchases/rest/relations/cancel"
```

```
{  
  "po:invoice-num" : 12345,  
  "po:purch-order-num" : 67890,  
  "po:amount" : 39.95  
}
```

76

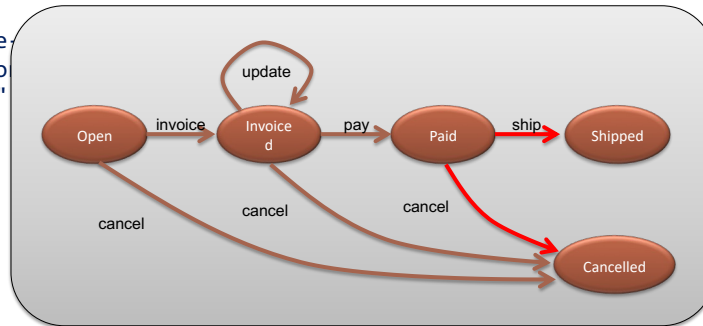
76

## Purchase Order Protocol: Payment Receipt

Link: <<http://www.example.org/purchases/rest/purchase/67890>>;  
rel="http://www.example.org/purchases/rest/relations/ship";  
type="application/vnd.purchases+json"

Link: <<http://www.example.org/purchases/rest/purchase/67890>>;  
rel="http://www.example.org/purchases/rest/relations/cancel"

```
{  
  "po:invoice"  
  "po:purch-o  
  "po:amount"  
}
```



77

## Purchase Order Protocol: Shipping Receipt

```
{  
  "po:invoice-num" : 12345,  
  "po:purch-order-num" : 67890,  
  "po:shippingUri" : ...  
}
```

78

78

## Additional Target Attributes

- Example:

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
      rel="http://www.example.org/purchases/rest/relations/ship";  
      type="application/vnd.purchases+json"
```

- Specifies data model, but not resource-specific type

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
      rel="http://www.example.org/purchases/rest/relations/ship";  
      type="application/vnd.purchases+json";  
      schema="http://www.example.org/schema/purchase-order"
```

- Specifies relation, but not the HTTP operation

```
Link: <http://www.example.org/purchases/rest/purchase/67890>;  
      rel="http://www.example.org/purchases/rest/relations/ship";  
      type="application/vnd.purchases+json";  
      operation="POST"
```

79

79

## JSON LINKING

80

80



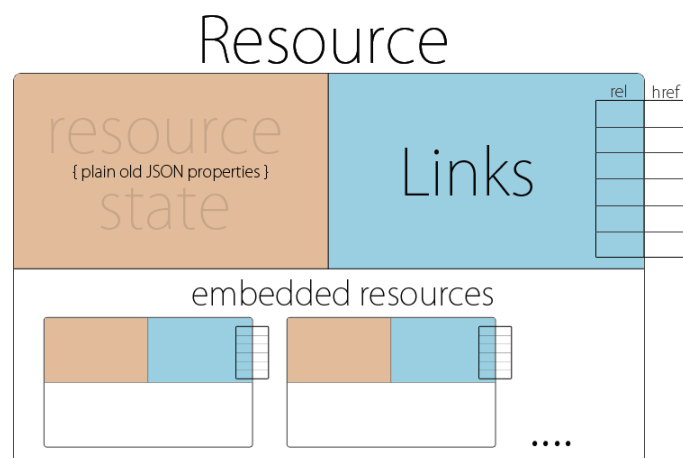
# Domain Application Protocol

- Express protocol via hypermedia
  - Links in responses
- Problem: JSON does not distinguish links!
- Solution 0: Use HTML or ATOM
- Solution 1: Web Linking
- **Solution 2: Extend JSON with linking**

81

81

# HAL: Hypertext Application Language



82

82

## Example: Patient

- No links

```
{
  "id" : 215,
  "patient-id" : 12345678,
  "name" : "Jane Doe",
  "dob" : "1992-02-17"
}
```

83

83

## Example: Patient

- Direct links

Semantic  
relationships

```
{
  "_links" : {
    "self" : {
      "href" : "http://www.example.org/patient/215",
    },
    "pcp" : {
      "href" : "http://www.example.org/provider/4634",
    }
  },
  "id" : 215,
  "patient-id" : 12345678,
  "name" : "Jane Doe",
  "dob" : "1992-02-17"
}
```

Link: <http://www.example.org/clinic/patient/215>; rel="self"

Link: <http://www.example.org/clinic/doctor/4634>; rel="doctor"

84

## Example: Patient

- Embedded Treatments

```
{
  "id" : 215,
  "patient-id" : 12345678,
  "name" : "Jane Doe",
  "dob" : "1992-02-17",
  "treatments" : [
    {
      "type" : "DRUG",
      "provider" : 8765,
      "name" : "Paxlovid",
      "dosage" : 135
    },
    {
      "type" : "SURGERY",
      "provider" : 5623,
      "date" : "2021-12-12"
    }
  ]
}
```

85

85

## Example: Patient

- Embedded Treatments with Links

```
{
  "_links" : {
    "self" : {
      "href" : ".../patient/215",
    },
    "pcp" : {
      "href" : ".../provider/4634",
    }
  }
  "id" : 215,
  "patient-id" : 12345678,
  "name" : "Jane Doe",
  "dob" : "1992-02-17",
```

```
  "_embedded" : {
    "treatments" : [
      {
        "_links" : {
          "self" : {
            "href" : ".../treatment/39876"
          },
          "provider" : { "href" : ... }
        }
        "type" : "DRUG",
        "provider" : 8765,
        "name" : "Paxlovid",
        "dosage" : 135
      },
      {
        "_links" : {
          "self" : {
            "href" : ".../treatment/67345"
          },
          "provider" : { "href" : ... }
        }
        "type" : "SURGERY",
        "provider" : 5623,
        "date" : "2021-12-12"
      }
    ]
  }
}
```

86

86

## JSON HYPER SCHEMA

87

87

## JSON Hyper Schema

- Extend JSON Schema to describe linking
  - "**base**": base for relative URIs
  - "**links**": list of LDOs
- Link Description Object (LDO)
  - "**rel**": Uniquely identify the resource
  - "**href**": URI template (populated from JSON data)
  - "**templatePointers**": instantiations for URI templates
  - "**targetMedia**", "**targetSchema**": representations
  - "**targetHints**": allowable operations

88

88

## Example: Patient

```
{
  "type" : "object",
  "properties" : {
    "patient-id" : { "type" : "number" },
    "name" : { "type" : "string" },
    "dob" : { "type" : "string" }
  }
}
```

89

89

## Example: Patient

```
{
  "type" : "object",
  "properties" : { "patient-id" : { "type" : "number" }, ... },
  "links" : [
    { "rel" : "self",
      "href" : "http://www.example.org/patient/215",
      "targetMediaType" : "application/json",
      "targetSchema" : { "$ref" : "#" },
      "targetHints" : { "allow" : ["GET", "PUT"] }
    },
    { "rel" : "pcp",
      "href" : "http://www.example.org/provider/4634",
      "targetMediaType" : "application/json",
      "targetSchema" : { "$ref" : "..." }
      "targetHints" : { "allow" : ["GET"] }
    }
  ]
}
```

90

90

## Example: Patient

```
{ "type" : "object",
  "properties" : {
    "patient-id" : { "type" : "number" },
    "name" : { "type" : "string" },
    "dob" : { "type" : "string" },
    "treatments" : {
      "type" : "array",
      "items" : {
        "type" : "object", "properties" : { "id": {"type": "number"} }
      }
    }
  }
}
```

91

91

## Example: Patient

```
{ "type" : "object",
  "properties" : {
    "patient-id" : { "type" : "number" }, ...
    "treatments" : {
      "type" : "array",
      "items" : {
        "type" : "object", "properties" : { "id": {"type": "number"} },
        "links" : [{
          "rel" : "treatment",
          "href" : "/treatment/{tmt_id}",
          "templateRequired" : ["tmt_id"],
          "templatePointers" : { "tmt_id": "/treatments/items/id" },
          "targetSchema" : { "$ref": "/properties/treatments/items" }
        }]
      }
    }
  }
}
```



92

92

## SESSION TYPES

93

93

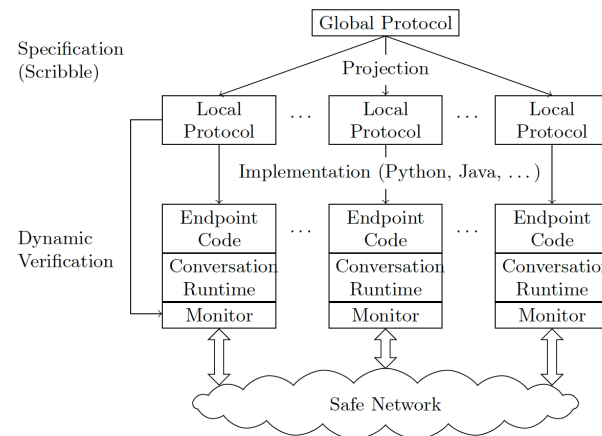
## Session Types

- Compile protocol to state machine at each participant
  - Check protocol compliance
- Global Type:
  - Global message exchanges
- Session Type
  - Messages sent and received by participant
- Implementation: Scribble

94

94

# Scribble



95

95

# Global Protocol

```

module TravelAgency;
  global protocol BookJourney(role Customer as C,
                              role Agency as A, role Service as S) {
    rec LOOP {
      choice at C {
        query(journey:String) from C to A;
        price(Int) from A to C;
        info(String) from A to S;
        continue LOOP;
      } or {
        choice at C {
          ACCEPT() from C to A;
          ACCEPT() from A to S;
          Address(String) from C to S;
          (Date) from S to C;
        } or {
          REJECT() from C to A;
          REJECT() from A to S;
        }
      }
    }
  }
}
    
```

96

96



# Local Protocol

```

module TravelAgency_BookJourney_Customer;
local protocol BookJourney_Customer at Customer
(role Customer as C, role Agency as A,role Service as S) {
  rec LOOP {
    choice at C {
      query(journey:String) to A;
      price(Int) from A;
      continue LOOP;
    } or {
      choice at C {
        ACCEPT() to A;
        Address(String) to S;
        (Date) from S;
      } or {
        REJECT() to A;
      }
    }
  }
}

```

