
Alain Houf
REST and JSON
InterSystems

REST

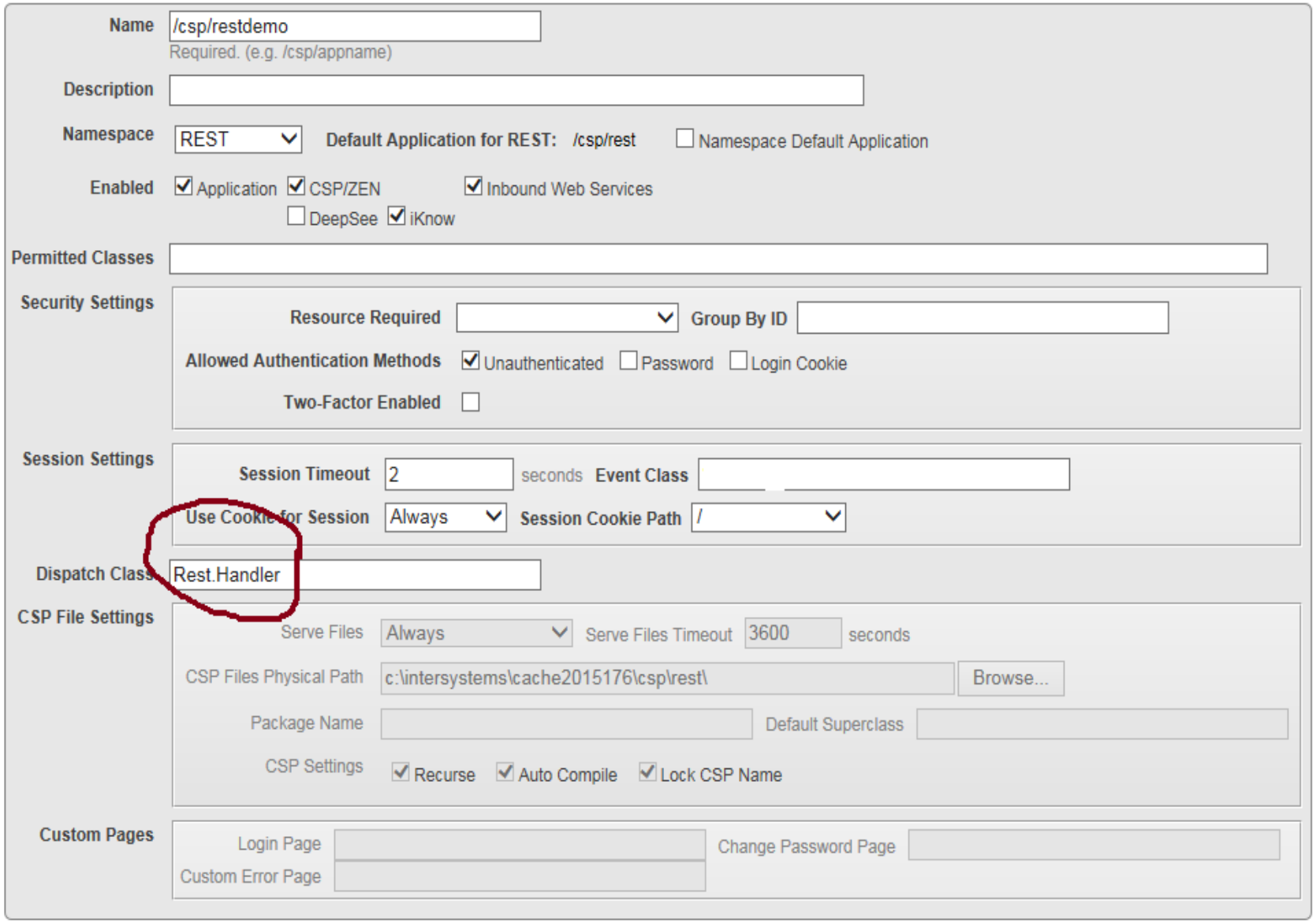
REpresentational State Transfer

- REST is not a technology, not even a standard!
- REST is just HTTP
- Any system that supports HTTP supports REST

Using REST in Caché

- You need to:
  - Create a new CSP application pointing on the target namespace
  - Declare in this CSP application the name of a "dispatch class"
  - Create a class that extends %CSP.REST
  - Use the "UrlMap" XData block to declare how to dispatch an HTTP request into Caché (to a classmethod)

Using REST in Caché



Using REST in Caché

- Class Rest.Handler Extends %CSP.REST
  {

  XData UrlMap
  {
  <Routes>
  <Route Url="/person/:person" Method="GET" Call="GetPerson"/>
  <Route Url="/sql" Method="GET" Call="GetSQL"/>
  <Route Url="/push" Method="POST" Call="PushPerson"/>
  <Route Url="/manuel/:person" Method="GET" Call="GetManuel"/
  >
  </Routes>
  }

Using REST in Caché

```
ClassMethod GetPerson(pPID As %String) As %Status
{
 Try {
     Set objPerson=##class(User.Person).%OpenId(pPID)
     Set %response.ContentType="application/json
   Do ##class(%ZEN.Auxiliary.jsonProvider).%ObjectToJSON(objPe
   rson)
 }
 Catch (e) {
  Set tSC=e.AsStatus()
 }
 Quit tSC
}
```

Using JSON in Caché

- Use class %ZEN.Auxiliary.jsonProvider to parse a JSON stream into Caché objects and to output JSON from any Caché structure

- Methods:
- %ObjectToJSON to output a Caché object as JSON (to the current device)
- %ConvertJSONToObject parses a JSON string or stream into a Caché object
- %WriteJSONFromSQL to output a ResultSet