Simple LIS Specification

v0.3 Proposed

Michael Sofaer
Chief Architect, Inigral, Inc.
msofaer@inigral.com

Abstract

The Simple LIS Specification is a standard for the transmission of educational data using HTTP and XML. It is designed using REST, while retaining the data model from IMS Global's LIS Specification (which uses SOAP).

Introduction

This specification is designed to be as similar as possible to the draft IMS LIS standard, while remaining a REST interface, and being as simple as possible. It is, however, not an official IMS Global product, and there is no guarantee that anything implemented based on this document will be in any way interoperable with an IMS system. Special thanks to Dr. Charles Severance for guidance on this document.

Architecture

This specification is envisioned to be used within a REST web service environment. It is designed to support one-way data integration from an SIS to an LMS. It is envisioned that the specification could be adapted for use in a file-based environment , if a web service implementation is not desired.

This specification is designed to use HTTP as fully as possible, and leverage the tools HTTP provides as much as possible. Anything that can be handled at the HTTP level is handled at that level.

All text fields SHOULD be UTF8. The SIS is responsible for providing text in the correct language, there are no language fields. All text fields should be human-readable, and suitable for direct display with no massaging.
All time fields SHOULD be DateTime fields, for uniformity.

Security

SSL SHOULD be used in all production implementations. HTTP Basic Authentication is sufficient for security if SSL is used. Custom security can be added by an implementer.

Definitions

root_url is used to indicate the root URL of the instance of Simple LIS, e.g.
http://lis.inigral.com
LMS is used to indicate the Simple LIS implementation, that is receiving and replicating the data from the source
SIS is used to indicate the client, which is the reference data source

Requests

A Simple LIS implementation MUST support PUT and GET requests directly to any resource handler, ex: root_url/meetings/
A Simple LIS implementation MUST support GET and DELETE requests with a sourced_id to any resource handler, ex: root_url/people/mg332

Resources

A Simple LIS implementation MUST provide the following persistent resources:
Person
CourseTemplate
Term
Group
CourseOffering
CourseSection
Membership
Meeting

Every resource MUST have a *sourced_id* field, which is unique within the scope of the resource and SHOULD be globally unique within the scope of all resources.
A put request with a new sourced_id SHALL cause a new record to be created by the LMS.
A put request with an existing sourced_id SHALL cause the LMS to replace the current record.

Data Models

blue fields are to be present in all resource creation requests, and MUST be persisted and returned
orange fields are optional for SIS to provide in creation requests, the LMS MUST persist and return them
green fields are optional extensions of the specification, a Simple LIS implementation SHOULD accept and persist them.

Person
names
  given
  family
  middle
contact_info
  email   This maps to EmailPrimary in the IMS Global LIS Specification. Should you want to add additional email fields, please see that document for what to call them.

For guidance on extensions to the Person object (multiple last names, for example), please see Appendix B.

CourseTemplate
title   A title for the template, e.g. Calculus I. It should be usable for display as a title on a detail screen
code This is the short identifier. e.g. MATH101. It should be usable as a short display field for list screens
description 255 character maximum. A general description of the course template e.g. "An introductory Calculus course for math majors".

Term
title This should be a short name for the term, e.g. "Spring 2009" or "Spr '09", usable for tabs or as list headers
starts_at DateTime
ends_at DateTime

Group
title
category The general kind of group being described, intended mostly for organizing navigation through the groups
sub_category The specific kind of group being described, this is intended to be useful for establishing permissions
description
parent_sourced_id Group hierarchy is possible (for example, a department might belong to a school, which might belong to a campus), but not required.

pre-defined categories are:
AcademicUmbrella (sub-categories are College, School, Campus, Department)
AcademicProgram (sub-categories are Concentration, Minor, Degree, Certification)
Residence (sub-categories are Dormitory, Greek, Apartment)
Enterprise
Administration (sub-categories are Admissions, Registrar,
StudentOrganization (sub-categories are Athletic, Intramural, StudentGovernment, Cultural, StudentOrganization)

sourced_id 'Application' is reserved for an Enterprise group for application administration. Putting or Deleting a group with that sourcedId has undefined behavior.

Please contact me at msofaer@inigral.com if you want additional categories/sub-categories added to the spec.

CourseOffering
term_sourced_id
course_template_sourced_id
group_sourced_id This is intended for the department (or other group) offering the course

In the case of a cross-listed course (e.g. MATH101 being offered by both Math and Engineering), there would be two offerings, one for each deaprtment, each with a section. SectionAssociation will be added in a future version of this specification to associate such sections with each other.

CourseSection
course_offering_sourced_id
label A short label for display after the course code. e.g. "LEC 01", so the whole list display would be "MATH101 LEC 01"
description A more specific description of the section as it will be taught in this instance. Things like the specific poets covered in a poetry course.

Membership
person_sourced_id
target_type
target_sourced_id
role

name
stars_at
ends_at
term_sourced_id

term_sourced_id should be provided for course_sections memberships, and must be the same as the term_id of the course_offering
pre-defined roles to sections are 'Instructor', 'Student' and 'TeachingAssistant'
pre-defined roles to AcademicUmbrellas are 'Instructor', 'Student', 'Alumnus', 'Chair', 'Moderator', and 'Staff'
pre-defined roles to StudentOrganizations are 'Member', 'Fan', 'Officer', 'Advisor'
pre-defined roles to Administration groups are 'Staff', 'Manager' and 'Ambassador'
pre-defined roles to Enterprise groups are 'Administrator', 'Staff', 'Moderator' and 'Analyst'

Meeting
target_type
target_sourced_id
i_calendar
See external resource for information on the icalendar object to put in here. e.g. http://tools.ietf.org/html/rfc2445#section-4.6.1

Foreign Key Constraints

The LMS SHOULD use foreign key constraints, and return 403 Forbidden when an attempt is made to delete a record that is still needed.
Polymorphic targets (memberships and meetings) cannot have foreign key constraints, and care must be taken that a multi-threaded system does not create data corruption.

The LMS MUST cascade deletion of a user's memberships when the user is deleted, rather than returning 403

Reference Implementation:

Please see https://github.com/MikeSofaer/simple-lis/tree for sample code and RSpec examples.

Appendix A: Sample Requests

I. Resource Creation.

A. Creating People

1) An OK Request to create one Person

URL root/people/

METHOD PUT

BODY

```
<people>
<person>
  <sourced_id>bjones8</sourced_id>
  <names>
    <given>Bob</given>
    <family>Jones</family>
  </names>
  <contact_info>
    <email>bob@your_school.edu</email>
  </contact_info>
</person>
</people>
```

RESPONSE:
HTTP 200 OK
URI: root/people/bjones8

2) A bad request (missing a family name)

URL root/people/

METHOD PUT

BODY

```
<people>
<person>
  <sourced_id>bjones8</sourced_id>
  <names>
    <given>Bob</given>
  </names>
  <contact_info>
    <email>bob@your_school.edu</email>
  </contact_info>
</person>
</people>
```

RESPONSE:
HTTP 422 UNPROCESSABLE ENTITY

There is no field with the name 'family' in

```
<names>
    <given>Bob</given>
  </names>
```

3) More than one Person at once

URL root/people/

METHOD PUT

BODY

```
<people>
<person>
  <sourced_id>acarey</sourced_id>
  <names>
    <given>Alexandra</given>
    <family>Holloway</family>
  </names>
  <contact_info>
    <email>fire@your_school.edu</email>
  </contact_info>
</person>
<person>
  <sourced_id>mdwight</sourced_id>
  <names>
    <given>Mark</given>
    <family>Dwight</family>
  </names>
  <contact_info>
    <email>mirabilis@your_school.edu</email>
  </contact_info>
</person>
</people>
```

RESPONSE:
HTTP 200 OK
URI: root/people/acarey
URI: root/people/mdwight

B. Creating a Group

1) An OK Request to create a School

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>baskin_engineering</sourced_id>
  <title>Baskin School of Engineering</title>
  <category>AcademicUmbrella</category>
  <sub_category>School</sub_category>
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/baskin_engineering

2) An OK Request to create a Department

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>baskin_engineering_bme</sourced_id>
  <title>Biomolecular Engineering</title>
  <category>AcademicUmbrella</category>
  <sub_category>Department</sub_category>
  <description>Department of Biomolecular Engineering at Baskin School of Engineering</description>
  <parent_sourced_id>baskin_engineering</parent_sourced_id>  #This is to establish the academic hierarchy
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/baskin_engineering_bme

3) An OK Request to create a Major

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>baskin_ug_bme</sourced_id>
  <title>BS in Bioengineering</title>
  <category>AcademicProgram</category>
  <sub_category>Major</sub_category>
  <description>Bachelor of Science in Bioengineering from the Baskin School of
Engineering</description>
  <parent_sourced_id>baskin_engineering_bme</parent_sourced_id>
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/baskin_ug_bme

4) **An OK Request to create a Concentration**

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>baskin_ug_bme_rehab</sourced_id>
  <title>BS in Bioengineering, Rehabilitation Concentration</title>
  <category>AcademicProgram</category>
  <sub_category>Concentration</sub_category>
  <description>Rehabilitation Engineering Concentration of the Bachelor of Science in
Bioengineering from the Baskin School of Engineering</description>
  <parent_sourced_id>baskin_ug_bme</parent_sourced_id>
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/baskin_ug_bme_rehab

5) **An OK Request to create a Minor**

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>baskin_ug_bme_bioinfo_minor</sourced_id>
  <title>Minor in Bioinformatics</title>
  <category>AcademicProgram</category>
  <sub_category>Minor</sub_category>
  <description>Minor in Bioinformatics from the Baskin School of
Engineering</description>
  <parent_sourced_id>baskin_ug_bme</parent_sourced_id>
</group>
</groups>
```

RESPONSE:

HTTP 200 OK
URI: root/groups/baskin_ug_bme_bioinfo_minor

6) An OK Request to create a Residence

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>albert_res_hall</sourced_id>
  <title>Albert Residence Hall</title>
  <category>Residence</category>
  <sub_category>Dormitory</sub_category>
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/albert_res_hall

7) An OK Request to create a Sports Team

URL root/groups/

METHOD PUT

BODY

```
<groups>
<group>
  <sourced_id>football</sourced_id>
  <title>Football Team</title>
  <category>StudentOrganization</category>
  <sub_category>Athletic</sub_category>
</group>
</groups>
```

RESPONSE:
HTTP 200 OK
URI: root/groups/football

C. Creating a Term

1) An OK Request to create a Term

URL root/groups/

METHOD PUT

BODY

```
<terms>
<term>
  <sourced_id>summer09</sourced_id>
  <title>Summer 2009</title>
  <starts_at>2009-07-01 00:00:00UTC</starts_at>
  <starts_at>2009-09-01 00:00:00UTC</starts_at>
</term>
</terms>
```

RESPONSE:
HTTP 200 OK
URI: root/terms/summer09

C. Creating Course_Sections

1) **First create the CourseTemplate**

URL root/course_templates/

METHOD PUT

BODY

```
<course_templates>
<course_template>
  <sourced_id>into_bioinform</sourced_id>
  <title>Introduction to Bioinformatics</title>
  <code>bme120</scode>
</course_template>
</course_templates>
```

RESPONSE:
HTTP 200 OK
URI: root/course_templates/intro_bioinform

2) **Then, the CourseOfferings. Biomolecular Engineering offering the course in Summer 2009**

URL root/course_offerings/

METHOD PUT

BODY

```
<course_offerings>
<course_offering>
  <sourced_id>into_bioinform_summer09</sourced_id>
  <course_template_sourced_id>into_bioinform</course_template_sourced_id>
  <term_sourced_id>summer09</term_sourced_id</term_sourced_id>
  <group_sourced_id>baskin_engineering_bme</group_sourced_id>
</course_offering>
</course_offerings>
```

RESPONSE:
HTTP 200 OK
URI: root/course_offerings/intro_bioinform_summer09

3) **This term the CourseSections are 1 lecture and 2 computer labs**

URL root/course_sections/

METHOD PUT

BODY

```
<course_sections>
  <course_section>
    <sourced_id>into_bioinform_summer09_l1</sourced_id>

<course_offering_sourced_id>into_bioinform_summer09</course_offering_sourced_id>
    <label>Lec 1</label>
  </course_section>
  <course_section>
    <sourced_id>into_bioinform_summer09_lab1</sourced_id>
```

```
<course_offering_sourced_id>into_bioinform_summer09</course_offering_sourced_id>
    <label>Lab 1</label>
  </course_section>
  <course_section>
    <sourced_id>into_bioinform_summer09_lab2</sourced_id>

<course_offering_sourced_id>into_bioinform_summer09</course_offering_sourced_id>
    <label>Lab 2</label>
  </course_section>
</course_sections>
```

RESPONSE:
HTTP 200 OK
URI: root/course_sections/intro_bioinform_summer09_l1
URI: root/course_sections/intro_bioinform_summer09_lab1
URI: root/course_sections/intro_bioinform_summer09_lab2

C. Adding Meetings

1) A meeting for a section

URL root/meetings/

METHOD PUT

BODY

```
<meetings>
<meeting>
  <sourced_id>into_bioinform_summer09_l1_m1</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <i_calendar>BEGIN:VCALENDAR
VERSION:2.0
METHOD:PUBLISH
BEGIN:VEVENT
DTSTAMP:20090611T075810Z
DTSTART:20090824T000000
DTEND:20090824T000100
LOCATION:Baskin Bioinformatics Computer Lab
RRULE:FREQ=WEEKLY;UNTIL=20091208;BYDAY=TU,TH
END:VEVENT
END:VCALENDAR</i_calendar>
</meeting>
</meetings>
```

RESPONSE:
HTTP 200 OK
URI: root/meetings/into_bioinform_summer09_l1_m1

2) Daily practice for the Football team

URL root/meetings/

METHOD PUT

BODY

```
<meetings>
<meeting>
  <sourced_id>football_practice</sourced_id>
  <target_sourced_id>football</target_sourced_id>
  <target_type>Group</target_type>
  <i_calendar>BEGIN:VCALENDAR
VERSION:2.0
METHOD:PUBLISH
BEGIN:VEVENT
DTSTAMP:20090611T075810Z
DTSTART:20090701T170000
DTEND:20090701T190000
LOCATION:Football Field
RRULE:FREQ=DAILY;UNTIL=20091208
END:VEVENT
END:VCALENDAR</i_calendar>
</meeting>
</meetings>
```

RESPONSE:
HTTP 200 OK
URI: root/meetings/football_practice

C. Adding Memberships

1) Teaching a section

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_001</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Instructor</name>
    <term_id>summer09</term_id>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_001

2) Teaching a lab

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_002</sourced_id>
  <target_sourced_id>into_bioinform_summer09_lab1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
```

```
    <name>Instructor</name>
    <term_id>summer09</term_id>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_002

3) Taking a lab and a lecture

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_003</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
    <term_id>summer09</term_id>
  </role>
</membership>
<membership>
  <sourced_id>mem_004</sourced_id>
  <target_sourced_id>into_bioinform_summer09_lab1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
    <term_id>summer09</term_id>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_003
URI: root/memberships/mem_004

4) A Student's academic groups

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_005</sourced_id>
  <target_sourced_id>baskin_engineering</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_006</sourced_id>
  <target_sourced_id>baskin_ug_bme</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_007</sourced_id>
  <target_sourced_id>baskin_ug_bme_rehab</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_008</sourced_id>
  <target_sourced_id>baskin_ug_bme_bioinfo_minor</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_005
URI: root/memberships/mem_006
URI: root/memberships/mem_007
URI: root/memberships/mem_008

5) Memberships to non-academic groups

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_009</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Resident</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_010</sourced_id>
  <target_sourced_id>football</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Member</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_011</sourced_id>   #Bob Jones moderates the Football team's organization page
  <target_sourced_id>football</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
    <name>Moderator</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_012</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>ResidentDirector</name>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_009
URI: root/memberships/mem_010
URI: root/memberships/mem_011
URI: root/memberships/mem_012

6) A Graduate Student's academic groups

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_013</sourced_id>
  <target_sourced_id>baskin_engineering</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
```

```
<membership>
  <sourced_id>mem_014</sourced_id>
  <target_sourced_id>baskin_ug_bme</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Alumnus</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_015</sourced_id>
  <target_sourced_id>baskin_ug_bme_rehab</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Alumnus</name>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_013
URI: root/memberships/mem_014
URI: root/memberships/mem_015

7) An Application Administrator

URL root/memberships/

METHOD PUT

BODY

```
<memberships>
<membership>
  <sourced_id>mem_016</sourced_id>
  <target_sourced_id>Application</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
    <name>Staff</name>
  </role>
</membership>
</memberships>
```

RESPONSE:
HTTP 200 OK
URI: root/memberships/mem_016

I. Resource Retrieval.

A. Retrieving People

1) All the People

URL root/people/

METHOD GET

RESPONSE:
HTTP 200 OK
BODY

```
<people>
<person>
  <sourced_id>bjones8</sourced_id>
  <names>
    <given>Bob</given>
    <family>Jones</family>
  </names>
  <contact_info>
    <email>bob@your_school.edu</email>
  </contact_info>
</person>
<person>
  <sourced_id>acarey</sourced_id>
  <names>
    <given>Alexandra</given>
    <family>Holloway</family>
  </names>
  <contact_info>
    <email>fire@your_school.edu</email>
  </contact_info>
</person>
<person>
  <sourced_id>mdwight</sourced_id>
  <names>
    <given>Mark</given>
    <family>Dwight</family>
  </names>
  <contact_info>
    <email>mirabilis@your_school.edu</email>
  </contact_info>
</person>
</people>
```

2) One Person

URL root/people/acarey
METHOD GET

RESPONSE:
HTTP 200 OK
BODY

```
<people>
<person>
  <sourced_id>acarey</sourced_id>
  <names>
    <given>Alexandra</given>
    <family>Holloway</family>
  </names>
  <contact_info>
    <email>fire@your_school.edu</email>
  </contact_info>
```

```
</person>
</people>
```

A. Retrieving Memberships

1) All the memberships

URL root/memberships/

METHOD GET

RESPONSE:

HTTP 200 OK

BODY

```
<memberships>
<membership>
  <sourced_id>mem_001</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Instructor</name>
    <term_id>summer09</term_id>
  </role>
</membership>
<membership>
  <sourced_id>mem_002</sourced_id>
  <target_sourced_id>into_bioinform_summer09_lab1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
    <name>Instructor</name>
    <term_id>summer09</term_id>
  </role>
</membership>
<membership>
  <sourced_id>mem_003</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
    <term_id>summer09</term_id>
  </role>
</membership>
<membership>
  <sourced_id>mem_004</sourced_id>
  <target_sourced_id>into_bioinform_summer09_lab1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
    <term_id>summer09</term_id>
  </role>
</membership>
```

```
<membership>
  <sourced_id>mem_005</sourced_id>
  <target_sourced_id>baskin_engineering</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_006</sourced_id>
  <target_sourced_id>baskin_ug_bme</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_007</sourced_id>
  <target_sourced_id>baskin_ug_bme_rehab</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_008</sourced_id>
  <target_sourced_id>baskin_ug_bme_bioinfo_minor</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_009</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Resident</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_010</sourced_id>
  <target_sourced_id>football</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>mdwight</person_sourced_id>
  <role>
    <name>Member</name>
  </role>
</membership>
```

```
<membership>
  <sourced_id>mem_011</sourced_id>
  <target_sourced_id>football</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
    <name>Moderator</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_012</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>ResidentDirector</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_013</sourced_id>
  <target_sourced_id>baskin_engineering</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_014</sourced_id>
  <target_sourced_id>baskin_ug_bme</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Alumnus</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_015</sourced_id>
  <target_sourced_id>baskin_ug_bme_rehab</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Alumnus</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_016</sourced_id>
  <target_sourced_id>Application</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>bjones8</person_sourced_id>
  <role>
    <name>Staff</name>
  </role>
```

```
</membership>
</memberships>
```

2) One Person's memberships

URL root/memberships?person_sourced_id=acarey
METHOD GET

RESPONSE:
HTTP 200 OK
BODY

```
<memberships>
<membership>
  <sourced_id>mem_001</sourced_id>
  <target_sourced_id>into_bioinform_summer09_l1</target_sourced_id>
  <target_type>Section</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Instructor</name>
    <term_id>summer09</term_id>
  </role>
</membership>
<membership>
  <sourced_id>mem_012</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>ResidentDirector</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_013</sourced_id>
  <target_sourced_id>baskin_engineering</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Student</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_014</sourced_id>
  <target_sourced_id>baskin_ug_bme</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>Alumnus</name>
  </role>
</membership>
<membership>
  <sourced_id>mem_015</sourced_id>
  <target_sourced_id>baskin_ug_bme_rehab</target_sourced_id>
  <target_type>Group</target_type>
```

```
     <person_sourced_id>acarey</person_sourced_id>
     <role>
       <name>Alumnus</name>
     </role>
   </membership>
   </memberships>
```

I. Resource Removal.

A. Retrieving People

1) #Removing a Person also removes all associated memberships

URL root/people/bjones8
METHOD DELETE

RESPONSE
HTTP 204 NO CONTENT

Additional test cases are available in the prototype specification code at http://github.com/MikeSofaer/simple-lis/tree/master/spec

Appendix B.

Data model extensions

I People

It is possible to extend the names method of the Person object to allow for more name types and values. Here is an example of multiple family names with an explicit order:

```
<names>
  <name>
    <type>family</type>
    <value>Miller</value>
    <order>1</order>
  </name>
  <name>
    <type>family</type>
    <value>Baker</value>
    <order>2</order>
  </name>
</names>
```

II Memberships

It is possible to add additional role objects to a membership, as is done in the IMS Global LIS Spec:

```
<membership>
  <sourced_id>mem_012</sourced_id>
  <target_sourced_id>albert_res_hall</target_sourced_id>
  <target_type>Group</target_type>
  <person_sourced_id>acarey</person_sourced_id>
  <role>
    <name>ResidentDirector</name>
    <term_sourced_id>summer09</term_sourced_id>
  </role>
  <role>
    <name>Resident</name>
    <term_sourced_id>spring09</term_sourced_id>
  </role>
  <role>
    <name>Resident</name>
    <term_sourced_id>fall08</term_sourced_id>
  </role>
  <role>
    <name>Resident</name>
    <term_sourced_id>spring08</term_sourced_id>
  </role>
  <role>
    <name>Resident</name>
    <term_sourced_id>fall07</term_sourced_id>
  </role>
</membership>
```