Quick Reference Guide

User Properties

Section	Property Description	eCollege Property Name	XPath to Property in IMS Enterprise Schema	Example of property element
3.3.2	Login Name	cn	enterprise/person/userid	<userid>*IKant*</userid>
3.3.2	Password	userPassword	enterprise/person/userid/@password	<userid password="*Think!*">
3.3.3.2.2	First Name	FirstName	enterprise/person/name/n/given	<given>*Immanuel*</given>
3.3.3.2.1	Last Name	LastName	enterprise/person/name/n/family	<family>*Kant*</family>
3.3.3.2.3	Middle Name	Middlename	enterprise/person/name/n/partname[@partnametype='Middlename']	<partname **partnametype="Middlename">***Alexander*</partname>
3..3.4.1	Gender	Gender	enterprise/person/demographics/gender	<gender>*M*</gender>
3.3.4.2	Birth Date	BirthDay	enterprise/person/demographics/bday	<bday>*April 22, 1724*</bday>
3.3.4.3	Disability	Disability	enterprise/person/demographics/disability	<disability>*hearing impaired*</disability>
3.3.5	Day-time phone number	TelephoneNumber	enterprise/person/tel*	<tel>*720.555.1212*</tel>
3.3.6	Email Address	Mail	enterprise/person/email	<email>*Imm.Kant@philosophers.net*</email>
3.3.7.3	Street Address	Street	enterprise/person/adr/street[position()='1']	<street>*123 Main Str.*</street>
3.3.7.3	Street Address 2nd	Street2	enterprise/person/adr/street[position()='2']	<adr>

User Properties

Section	Property Description	eCollege Property Name	XPath to Property in IMS Enterprise Schema	Example of property element
	line			`<street>above the bakery</street>` *`<street>123 Maint St.</street>`* `</adr>`
3.3.7.4	City/Town	City	enterprise/person/adr/locality	`<locality>`*Königsberg*`</locality>`
3.3.7.5	State/Province	State	enterprise/person/adr/region	`<region>`*Kalinigrad*`</region>`
3.3.7.6	Postal/Zip Code	PostalCode	enterprise/person/adr/pcode	`<pcode>`*99299*`</pcode>`
3.3.7.7	Country	Country	enterprise/person/adr/country	`<country>`*Russia*`</country>`
4.2	Any other user property	Exact name, including capitalization provided by Client Services	enterprise/person/extension/personproperty	`<personproperty` **propertyname="SomePropertyName">***Some Value*`</personproperty>`

Legend
- *Italics - Example values are in italics*
- **Bold - Bold text indicates information that must be provided in addition to the value and the element name**
- Sans Serif - Additional elements to provide context are in a Sans Serif typeface

*See section 3.3.5 of the annotated guide for additional important information

Required Elements	
3. MS elements consumed by the eCollege API for SIS	
3.1 <enterprise> Elements	
Description:	The enterprise element is the root of the document and the container for all the data objects of the IMS Enterprise Schema
eCollege Implementation:	The person, group, and membership elements are each optional under the IMS specification, but logically at least one of them must be present. For the purposes of eCollege, there must be at least one instance of each of the three elements. Because a person cannot exist within the eCollege system without being in a defined context beyond just the EP, those context(s) are defined in the group elements, and the relation of the person to the group is defined in a membership element. As a result, all three must be present in every IMS Enterprise document sent to eCollege, and are present in every document created by eCollege.
Data Type:	
Multiplicity:	Single, Required
Attributes:	xmlns (optional) – the enterprise element should reference the namespace for the document: http://schemas.ecollege.com/ims_epv1p1.xsd
Elements:	- properties - person - group - membership

Example:	<pre><enterprise xmlns="http://schemas.ecollege.com/Common/2006/01/ims_epv1p1.xsd">
	        <properties>...</properties>
	        <person>...</person>
	        <group>...</group>
	        <membership>...</membership>
	</enterprise></pre>

3.2 <properties> Elements	
Description:	Information about the entire document and the data being exchanged between the SIS and the eCollege API
eCollege Implementation:	eCollege ignores all elements within properties.
Data Type:	Elements
Multiplicity:	Single, Required

Required Elements	
Attributes:	`xmlns` (optional) – the enterprise element should reference the namespace for the document: http://schemas.ecollege.com/ims_epv1p1.xsd

3.3 \<person\> Elements	
Description:	The container for information about a particular person relevant to the eCollege LMS.
eCollege Implmentation:	The majority of the information provided in the IMS specification is ignored by eCollege (see the child elements list). Extensions to the person element are supported to allow the EP and eCollege to transfer specific information necessary for the EP and information agreed upon between the two organizations.This element is required in every document sent to the eCollege API for SIS so that eCollege can identify and/or create the user.
Data type:	`Elements`
Multiplicity:	Single, **eCollege Required**
Attributes:	`recstatus` (optional and ignored by eCollege)
Elements:	`sourcedid` `name` `email` `tel` `adr`
Example:	```<person> <sourcedid> <source>Muppet University</source> <id>KERM148</id> </sourcedid> <userid password="BeingGreen">Kfrog1</userid> <name>...</name> <email>frogk@Disney.com</email> <tel teltype="Voice">303-555-1212</tel> <adr>...</adr> <extension>…</extension></person>```

3.3.1 \<sourceid\>	
Description:	The identifier of the person as defined by the source system. The sourcedid must uniquely identify the person within the document so that it can be used as a referential key between one or more member elements and the person.

Required Elements	
eCollege Implementation:	The eCollege SIS API will permanently associate the provided sourcedid with the user in our datastore.
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
Elements:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
3.3.2 <userid>	
Description:	The person's ID to access the eCollege LMS.
eCollege Implementation:	Also often called the "Login ID", the userid is the unique value that eCollege uses to identify a person within the context of an EP in the eCollege LMS. Because the IMS Enterprise Schema specifies this element as optional, the eCollege SIS for API will attempt to identify a person by the sourcedid/id value if, and only if, the userid is absent. *Also note that the string lengths for the userid and password are more restrictive that the IMS Enterprise Schema.* Because the userid is the account a person uses to access the eCollege LMS, and that account is what is associated with enrollments (through the membership structure), while it is possible for a physical person to have more than one login that they might use, it is **not** possible for a person element to have more than one userid. As a consequence, while the IMS specification for multiplicity is Many, Optional, the eCollege implementation is Single, Optional. Because we cannot guarantee that the first userid element will always be used, no more than one userid element should be included when sending data to eCollege.
Data type:	`string 255 – invalid characters are: %][+<>";'=:/|\ and any white space.`
Multiplicity:	Single, **eCollege Required**
Attributes:	eCollege currently supports only the `password` attribute. • `password` (optional) – the password used to validate the person when logging in to the eCollege LMS. The password may be left blank if the person does not need to be created in the eCollege LMS, or the EP has opted to have random passwords generated. The length can be up to 50 characters. Invalid characters are: `%][ [+<>";'=:/|\_` and any white space .
Elements:	None
3.3.3 <name> Elements	
Description:	Data element for the name of the person.

Required Elements	
eCollege Implementation:	No additional information.
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None
| Example: | ```
<name>
    <fn>Mr. Kermit The Frog</fn>
    <n>…</n>
</name>
``` |

3.3.3.1 <fn>

Description:	The complete formatted name of the person.
eCollege Implementation:	eCollege SIS for API does not make use of this value.
Data type:	`string 256`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None

3.3.3.2 <n> Elements

Description:	The name of the person broken into all of its distinct components.
eCollege Implementation:	The eCollege SIS for API uses the elements of the n element to obtain all of the naming information about a person. As a consequence, this element—while optional in the IMS Enterprise specification—is Required for the eCollege API for SIS.
Data type:	`Elements`
Multiplicity:	Single, **eCollege Required**
Attributes:	None

Required Elements	
Elements:	None
| Example: | ```
<n>
    <family>Frog</family>
    <given>Kermit</given>
    <partname partnametype="Middlename">The</partname>
</n>
``` |

3.3.3.2.1 \<family\>

Description:	The family name of the person. Because the IMS Enterprise specification is culture neutral, this is not necessarily the last name.
eCollege Implementation:	While optional in the IMS Enterprise specification, this element is Required for the eCollege API for SIS. The string length is also more restrictive than the IMS Enterprise Schema.
Data type:	`string 40`
Multiplicity:	Single, **eCollege Required**
Attributes:	None
Elements:	None

3.3.3.2.2 \<given\>

Description:	The given name of the person. Because the IMS Enterprise specification is culture neutral, this is not necessarily the first name.
eCollege Implementation:	While optional in the IMS Enterprise specification, this element is Required for the eCollege API for SIS. The string length is also more restrictive than the IMS Enterprise Schema.
Data type:	`string 40`
Multiplicity:	Single, **eCollege Required**
Attributes:	None
Elements:	None

3.3.4 \<email\>

Description:	Email address used to contact a person.
eCollege Implementation:	No additional information.

Required Elements	
Data type:	`String 256`
Multiplicity:	Single, **eCollege Required**
Attributes:	None
Elements:	None
3.4 <group> Elements	
Description:	The container for all of the information about a group and its relationship to other groups. A group can be a collection of individuals, a set of curriculum definitions, or any other collection of relevant objects. The group structure is a convenient abstract container for any collection of common objects.
eCollege Implementation:	eCollege uses the `group` element to represent an area in which a user can be enrolled – typically this is a course, although less frequently it may be an Enrollable Node. (Nodes are a concept unique to eCollege, relating to hierarchical administrative structures. An eCollege Client Services Consultant can help determine when it is appropriate to enroll a user specifically in a node.) The actual enrollment is described by the `membership` elements. While the IMS Enterprise Schema considers the `group` element optional, in the context of a document for the eCollege API for SIS it is Required. Since the document describes enrollments (`membership`) of users (`person`) in courses (`group`), the data is meaningless without at least one instance of each.
Data type:	Elements
Multiplicity:	Many, **eCollege Required**
Attributes:	`recstatus` (optional) – because courses are not created using the API for SIS, this attribute is ignored.
Elements:	• `sourcedid` • `grouptype`
| Example: | ```
<group>
     <sourcedid>
          <source>Muppet University</source>
          <id>19791007</id>
     </sourcedid>
     <grouptype>
``` |

Required Elements	

```
                <typevalue>Call Number</typevalue>
            </grouptype>
        </group>
```

3.4.1 <sourceid>

Description:	The identifier of the group as defined by the source system. The sourcedid must uniquely identify the group with the document so that it can be used as a referential key between one or more membership elements and the group
eCollege Implementation:	The sourcedid for a course is expected to have the Course Call Number as the id and the EP's SIS system as the source. Since Course Call Number is a value generated by the EP and stored in the eCollege system at the time of course creation, that value is already mapped to a course in the eCollege system. If it is necessary to provide Enrollable Node information, the sourcedid is the eCollege system, and the source should be ECLG while the id is the Client Sort String of the enrollable node.
Data type:	`Elements`
Multiplicity:	Single, **eCollege Required**
Attributes:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
Elements:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.

3.4.2 <grouptype> Elements

Description:	Category information for the group.
eCollege Implementation:	The grouptype element is required by the IMS Enterprise Schema and is used to determine which course or enrollable node the group relates to. The element typevalue indicates whether the group is a course or an enrollable node. If it is a course, typevalue must be *Call Number;* for an enrollable node, it must be *Enrollable Node.* Elements with other typevalue values will be discarded; however, including numerous extraneous groups can have a negative impact on processing time.
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	None
Elements:	`typevalue`
| Example: | ```
<grouptype>
    <typevalue>Call Number</typevalue>
</grouptype>
``` |

Required Elements

3.5 <membership> Elements	
Description:	The container for all of the information about the members (as defined in the `person` and/or `group` structures) of a particular Group. This structure is used to establish the membership relations between `Groups` and `Groups/Persons`.
eCollege Implementation:	This is where an enrollment in a course or node (rarely a node) is established, by relating a person to a group. Because the API for SIS only handles enrollment information, an incoming document is meaningless without at least one membership element. The entities involved in the relationship are determined through the descendent sourcedid elements (see below for more detail).
Data type:	`Elements`
Multiplicity:	Many, **eCollege Required**
Attributes:	None
Elements:	• `sourcedid` • `member`
Example:	(Because it is important to see most of the membership element at once to understand the function of this key element, this example drills further into the child elements than others.) ``` <membership> <sourcedid> <source>Muppet University</source> <id>19791007</id> </sourcedid> <member> <sourcedid> <source>Muppet University</source> <id>KERM148</id> </sourcedid> <role>...</role> <member> </membership> ``` Note that the `membership/sourcedid` exactly matches the `sourcedid` for the `group` example, while the `membership/member/sourcedid` exactly matches the `sourcedid` for the `person` example, enrolling Mr. Kermit The Frog in BUS 201 with the specified `role`.

Required Elements

3.5.1 <sourceid>

Description:	The identifier of the group as defined by the source system. The sourcedid must uniquely identify the group with the document so that it can be used as a referential key between one or more membership elements and the group. While the sourcedid must be unique within the context of all group elements, the same sourcedid may appear in multiple membership elements (although this would be unusual). The effect of having multiple membership elements with the same sourcedid is the same as including every child member element of each of the membership elements under a single membership element.
eCollege Implementation:	`No additional information.`
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
Elements:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.

3.5.2 <member> Elements

Description:	A member of the group defined by the sourcedid in the parent membership
eCollege Implementation:	No additional information
Data type:	`Elements`
Multiplicity:	Many, Required
Attributes:	None
Elements:	• sourcedid • role
Examples:	``` <member> <sourcedid> <source>Muppet University</source> <id>KERM148</id> </sourcedid> <role>...</role> </member> ```

Required Elements	
3.5.2.1 \<sourcedid\>	
Description:	The identifier of the person as defined by the source system. The sourcedid must uniquely identify the person within the document so that it can be used as a referential key between the member element and the person. While the sourcedid must be unique within the context of all person elements, the same sourcedid may appear in multiple membership/member elements (i.e., the person may be associated with more than one group in a single document).
eCollege Implementation:	No additional information.
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
Elements:	See Section 3.6 of the *Annotated Guide to the IMS Spec* for details of this common data structure.
3.5.2.2 \<role\> Elements	
Description:	The role of the member in the group.
eCollege Implementation:	While the IMS Enterprise specification allows a member to have multiple roles in a single group, the eCollege system does not support this. If more than one role is defined for a particular membership/member element, only one may have an active status.
Data type:	`Elements`
Multiplicity:	Many, Required
Attributes:	`recstatus` (NMToken: 1=Add; 2=Update; 3=Delete) optional. This is ignored by eCollege since the required action is determined by comparing with any existing information. Delete is not supported by the eCollege API for SIS. Members must be given a Drop role instead. `roletype` (NMToken) – optional – the member's function with a Group. This value is too coarse for effective use within the eCollege system and is ignored.
Elements:	`Subrole`

Example:	`<role>`
	    `<subrole>2</subrole>`
	`</role`

3.5.2.2.1 \<subrole\>	
Description:	Further qualifies the member's role in the group.
eCollege	The eCollege role identifier for the person in the group. Because the eCollege system defines roles in a much more granular fashion, tailored

Required Elements	
Implementation:	to the needs of each EP, the `role/@roletype` attribute is too coarse to be effective in determining the role a person should be assigned within the eCollege system. A Client Services Consultant can provide specific information about the role identifiers for a specific EP. **Note:** The IMS Specification defines `subrole` as a string; however, the eCollege system is looking for a numeric RoleID for this field. Refer to the *Use case and test data* sample in the Developer's Guide, under Sample Code for an example of an XML file.
Data type:	`string 32`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None
3.6 Common <sourcedid> Elements	
Description:	The ID of the data object as defined by the source system. The sourcedid is key to relating data elements to one another in the membership data structure. The sourcedid must uniquely identify the person or group in the document.
eCollege Implementation:	The sourcedid is key to relating elements of the XML document to one another; it is also the means used by the eCollege Sis API to identify entities that already exist in the eCollege system. For persons and groups that are courses, the source of the id should be the EP's Sis. For groups that are enrollable nodes, the source should be ECLG.
Data type:	`Elements`
Multiplicity:	Single, Required
Attributes:	`sourcedidtype` (optional) – ignored by eCollege
Elements:	`source` `id`
Example:	``` <sourcedid> <source>Muppet University</source> <id>19791007</id> </sourcedid> ```
3.6.1 <source> Element	
Description:	The source system generating the identifier
eCollege	The source system generating the identifier

Required Elements	
Implementation:	
Data type:	`string 32`
Multiplicity:	Single, Required
Attributes:	None.
Elements:	None.

3.6.2 \<id> Element

Description:	The unique identifier of the data element (`person` or `group`) in the `source` system.
eCollege Implementation:	eCollege does not retain this id.
Data type:	`string 256`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None

4. eCollege Extensions

4.1.1 \<resultcode>

Description:	A numeric code for the error encountered. If the result is successful, informational, or a warning, the value is 0. A list of known result codes where the source system may reasonably attempt to fix the problem is available from your Client Services Consultant. The IMS schema accepts either the numeric or string value for type; however, due to how the eCollege system serializes the code, the result type will always return as a string value (Success, Warning, Error).
Data type:	`integer`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None

Required Elements	
4.1.2 \<message\>	
Description:	A human-readable message describing the result.
Data type:	`string 4096`
Multiplicity:	Single, Required
Attributes:	None
Elements:	None

*See section 3.3.5 of the annotated guide for additional important information