**Sports Recruiter**

**Server Interface**

**Video-related Interfaces**

- `selectAthleteVideos(guid athleteID, VideoSelectionCriteria selection) : videoList`
  - `athleteID` required
  - `selection` may be null (returns all)
  - `videoList` is a list of video information objects

- `selectTeamVideos(guid teamID, VideoSelectionCriteria selection) : videoList`
  - `teamID` required
  - `selection` may be null (returns all)
  - `videoList` is a list of video information objects

**Video-related Interfaces**

- `getVideoStream(guid videoID) : VideoStreamDescriptor`
  - returns a `VideoStreamDescriptor` for the video specified by the `videoID`
  - `VideoStreamDescriptor` identifies the server and port to use, will be extended to include methods for authentication and authorization to video service

- `downloadVideo(guid videoID) : Url`
  - returns a `Url` from which the video can be downloaded
  - `QUESTION: should we just stream the video to a data sink that saves to a file instead of to the athlete/editor?`
    - More complicated than `Url` download?
    - Eliminates the need to expose a `URL` that is not ‘publicly’ available

**Video-related Interfaces**

- `uploadAthleteVideo(guid athleteID, guid teamID, guid opponentID, string title, string description, int year) : UploadDescriptor`
  - `athleteID`, `title`, and `description`, and `year` are required
  - `teamID`, `opponentID` may be null
  - returns server/port for upload sequence

- `uploadTeamVideo(guid teamID, guid opponent, string title, string description, int year) : UploadDescriptor`
  - `teamID`, `title`, `description`, and `year` are required
  - `opponentID` may be null
  - returns server/port for upload sequence
Video-related Interfaces

- `getVideoAuthorizations( ) : GuidList`
- `allowVideoAccess( GuidList colleges ) : bool`
  - if colleges is null, all access is granted
  - if colleges is empty, no access is granted

Authentication Interface

- `enrollAthleteClient(Athlete athlete, BillingInfo info)`
- `enrollCollegeClient(College college, BillingInfo info)`
- `enrollHighSchoolClient(HighSchool hs, BillingInfo info)`
- `login( string id, string password ) : string`
  - returns session identifier to be used to authorize access in other methods
- `logout( string session )`
  - ends session, new login required for further access

Message Interface

- `getMessages( guid id, guid lastViewedID ) : MessageList`
  - id: team or athlete id
  - lastViewedID: guid of last request of this type viewed
  - returns a (possibly empty) list of message ids for the current user matching the given type
- `getMessage( guid messageID ) : Message`
- `sendMessage( guid addressee, Message message ) : bool`
  - extension: implement multicast by accepting guidList
- `deleteMessage( guid messageID ) : bool`

Interface to Athlete Information

- `getAthletes( AthleteSelectionCriteria selection ) : AthleteList`
  - selection criteria includes state, region, high school id, and position
  - returns a list of athlete guids
- `getAthlete( guid athleteID ) : Athlete`
  - Athlete class contains directory information
- `getAcademicInfo( guid highSchoolID, guid athleteID ) : AcademicInfo`
- `getForm( guid highSchoolID, FormType form ) : Form`
- `updateAthlete( guid athleteID, Athlete athlete ) : bool`
Interface to Athlete Information

- `getAcademicAuthorizations() : GuidList`
- `allowAcademicAccess( GuidList colleges ) : bool`
  - colleges cannot be null or empty
- `removeAcademicAccess( GuidList colleges ) : bool`
  - colleges cannot be null or empty

Interface to College Information

- `getColleges( CollegeSelectionCriteria selection ) : CollegeList`
  - returns a list (possibly empty) of college ids matching the given selection criteria
- `getCollege( guid collegeID ) : College`
  - College includes directory information (includes up to 3 title/URL pairs of links to college web sites)
- `updateCollege( guid collegeID, College college ) : bool`

Interface to High School Information

- `getHighSchools( HighSchoolSelectionCriteria selection ) : HighSchoolList`
  - returns a list (possibly empty) of high school ids matching the given selection criteria
  - selection criteria includes state, region, and school size
- `getHighSchool( guid highSchoolID ) : HighSchool`
  - HighSchool includes directory information, generic forms, and list of current athletes
- `updateHighSchool( guid highSchoolID, HighSchool highSchool ) : bool`
- `sendForm( guid highSchoolID, FormType type, Form form ) : bool`