Roles in Schema.org - final draft

Overview

Historically, schema.org descriptions have been simple and unqualified. For example, the <http://schema.org/actor> property relates a Movie, TVEpisode, RadioSeries etc. to a Person. Although this simplicity was well motivated, and appropriate to schema.org's emphasis on ease of adoption for publishers and webmasters, there are many scenarios when it is useful to provide a richer description. This document presents a data structure and supporting vocabulary for such situations, motivated by practical considerations around sports and other schemas.

Status of this Document

An earlier version of this document was shared as a proposal to the Web Schemas community (Mar 26 PDF). This version is a final draft for schema.org adoption, and refines the design to adopt a more symmetrical approach. The appendix has notes on similar related and earlier designs. *Where possible we use current real schema.org vocabulary; but in a few cases we might introduce convenient but proposed or fictional properties for ease of explanation.*

Example scenarios

1. **Sports:** modeling time-bounded sporting roles.

In describing sports-related information, it is common to want to temporarily qualify information.

For example: **JoeMontana** was an **athlete** in the *FootballTeam* **SF49ers**, between a **startDate** of **2002** and an **endDate** of **2008**; his **position** was that of **QuarterBack**.

Here's a preview of the Roles approach expressed in JSON-LD. We'll then build up to this step by step:

```
{
    "@context": "http://schema.org",
     "@type": "AmericanFootballRole",
     "roleSubject": {
            "@type": "AmericanFootballTeam",
            "name": "San Francisco 49ers"
     },
     "roleProperty": {
```

```
        "http://schema.org/athlete"
    },
    "roleObject": {
        "@type": "Person",
        "name": "Joe Montana"
    },
    "startDate": "1979",
    "endDate": "1992",
    "position": "Quarterback"
}
```

2. **Media**: additional details about the relationship between two entities, e.g. an actor and a movie.

Simple schema.org already allows us to say that **Bill Murray** was an actor in the *Movie Ghostbusters*. For a role-based example, we might want to add that he played the character whose name was **Dr. Peter Venkman**.

3. **alumni:** attaching 'graduationYear' to an alumni relationship.

Schema.org already has explicit names for both 'alumni' (from an EducationalOrganization to a Person) and its inverse 'alumniOf' (from a Person to an EducationalOrganization). This makes for a convenient example, as we can try out descriptions of this situation from both a Person-centric and from an Organization-centric perspective.

For example we might want to say not only that 'university_of_bristol alumni dan', but that this situation also involved a graduationYear of 1994, or a startDate of 1991.

Vocabulary: introducing Role, roleSubject, roleProperty and roleObject

The schema.org approach here is to introduce a new general-purpose type called "Role" which addresses these and related usecases. This allows us to add additional information as properties of the Role, since it is not possible in the schema.org datamodel to attach annotations directly to the relationship between two entities. Sub-types of Role can optionally be used to help organize role-oriented terminology, but are not strictly necessary. There are potentially many distinct roles that involve any pair (or larger collection) of entities.

The proposed model for using this new terminology is that any entity-valued property (e.g. actor, alumni, player, …) can now be used in a Role. In addition to 'actor' being expected on Movie or

TVSeries or RadioEpisode, we also can expand upon 'actor' information using a Role in which it is the roleProperty. Similarly, 'alumni' could be used directly on an EducationalOrganization, or it could be used further described using a Role.

This gives us two patterns. The traditional simple pattern, and a richer variant which uses the additional role entity:

In simple text-form, our 3 examples:

- **SF49ers** --quarterback→ **JoeMontana**.
- **Ghostbusters** --actor→ **BillMurray.**
- **BristolUniversity** --alumni→ **DanB**.

In each of these cases, we can also represent the relationship explicitly as a Role with subject, property and object indicated. By doing so, we can then attach additional information to the Role, such as temporal ranges. We can also reverse the process by taking the subject/predicate/object triple from a Role, and expressing those directly as a simple flat description.

Here are these three examples again, showing subject, property and object for each:

roleSubject	roleProperty	roleObject
SF49ers	quarterback	JoeMontana
Ghostbusters	actor	BillMurray
BristolUniversity	alumni	DanB

Markup Examples

How does the Sports example look in concrete markup? Let's try it first with markup that begins with a Team. First we'll start with the unqualified, simple version. Then we'll extend this to add a Role.

Simple flat property: `athlete` described directly via original defined range of `Person,` telling us that someone is an athlete in the team (but leaving lots of other questions unanswered):

```
{
  "@context": "http://schema.org",
  "@type": "AmericanFootballTeam",
  "name": "San Francisco 49ers",
  "athlete": {
    "@type": "Person",
    "name": "Joe Montana"
  }
}
```

This is straightforward, but doesn't let us say *when* Joe was an athlete for this Team; or specifically which *position* he played. To do that, we need to introduce another entity: a Role.

This gives us the finished Role-based description:

```
{
    "@context": "http://schema.org",
     "@type": "AmericanFootballRole",
     "roleSubject": {
            "@type": "AmericanFootballTeam",
            "name": "San Francisco 49ers"
     },
     "roleProperty": {
         "http://schema.org/athlete"
     },
     "roleObject": {
         "@type": "Person",
         "name": "Joe Montana"
     },
     "startDate": "1979",
     "endDate": "1992",
     "position": "Quarterback"
}
```

Our other two examples are similar, and shown below without further discussion.

Example 2: Ghostbusters actor BillMurray

```
{
    "@context": "http://schema.org/",
     "@type": "MovieRole",
     "roleSubject": {
            "@type": "Movie",
            "name": "Ghostbusters"
     },
     "roleProperty": {
         "http://schema.org/actor"
    },
    "roleObject": {
        "@type": "Person",
        "name": "Bill Murray"
    },
    "characterName": "Dr. Peter Venkman"
}
```

Example 3: UniversityBristol alumni DanB

```
{
    "@context": "http://schema.org/",
     "@type": "Role",
     "roleSubject": {
            "@type": "CollegeOrUniversity",
            "name": "University of Bristol"
     },
     "roleProperty": {
         "http://schema.org/alumni"
    },
    "roleObject": {
        "@type": "Person",
        "name": "Dan Brickley"
```

```
    },
    "graduationYear": "1994"
}
```