Wizualizacja spuścizny kulturowej: otwarte Linked Data w Carnegie Hall cz. 1

Rob Hudson - Photo by Gino Francesconi

Przedstawiamy gościnny blog Roberta Hudsona, archiwisty z Carnegie Hall w Nowym Jorku. Rob jest z wykształcenia muzykiem, zainteresowany archiwami, pracuje w Carnegie Hall od 1977 roku. Odkrywszy bazę danych występów w Carnegie Hall sięgających 19 wieku, Rob postanowił nauczyc sie programowania i dokonać konwersji danych w postać otwartego Linked Data tak, aby można było odkrywać powiązania i informacje o kompozytorach, wykonawcach i koncertach. Wielu polskich twórców i wykonawców przez lata brało udział w przedstawieniach w Carnegie Hall. Inicjatywa Roba przyczyni się, miejmy nadzieję, do udostępnienia ciekawego rozdziału z historii muzyki również polskim fanom.

Part I: Process

My name is Rob Hudson, and I'm the Associate Archivist at Carnegie Hall , where I've had the privilege to work since 1997. I'd like to tell you about my experience transforming Carnegie Hall's historical performance history data into Linked Open Data, and how within the space of about two years I went from someone with a budding interest in linked data, but no clue how to actually create it, to having an actual working prototype.

First, one thing you should know about me: I'm not a developer or computer scientist. (For any developers and/or computer scientists out there reading this right now: skip to the next paragraph, and try to humor me.) I'm a musician who stumbled into the world of archives by chance, armed with subject knowledge and a love of history. I later went back and got my degree in library science, which was an incredibly valuable experience, and which introduced me to the concept of Linked Open Data (LOD), but up until relatively recently, the only lines of

programming code I'd ever written was a "Hello, World!" - type script in Basic — in 1983. I mention this in order to give some hope to others out there like me, who discovered LOD, thought "Wow, this is fantastic — how can I do this?", and were told "learn Python." Well, I did, and if I can do it, so can you — it's not that hard. Much harder than learning Python — and, one might argue, more important — is the much more abstract process of understanding your data, and figuring out how to describe it. Once you've dealt with that, the transformation via Python is just process — perhaps not a cakewalk, but nonetheless a methodical, straightforward process that you can learn and tackle, step by step.

Now let me tell you a bit about the data that I worked with for my linked data prototype. The Carnegie Hall Archives maintains a database that attempts to track every event, both musical and nonmusical, that has occurred in the public performance spaces of Carnegie Hall since 1891. (Since the CH Archives was not established until 1986, there are some gaps in these records, which we continue to fill in using sources like digitized newspaper listings and reviews, or missing concert programs we buy on eBay.) This database now covers more than 50,000 events of nearly every conceivable musical genre: classical, folk, jazz, pop, rock, world music, and no doubt some I'm overlooking. But Carnegie Hall has always been about much more than music; its stages have also featured dance and spoken word performances, as well as meetings, lectures, civic rallies, political conventions — there was even a children's circus , complete with baby elephants, in 1934. Our database has corresponding records for more than 90,000 artists, 16,000 composers and over 85,000 musical works. Starting in 2013, we began publishing some of these records to our website , where you can now find the records for nearly 18,000 events between 1891 and 1955. The limited release reflects our ongoing process of data cleanup, and we're continuing to publish new records each month. For my linked data prototype, I chose to use this published data set, since I knew it was good, clean data.

In their breadth and depth these records, reflecting musical performance practice and standards, programming choices, and even current events, offer a vivid cross-section of the cultural and societal history of the past 124 years. They illustrate one of the things I love about Carnegie Hall: beyond its iconic status, which of course has helped to attract the greatest performers of every era, the Hall has functioned as a kind of focal point for culture and society. The events are like a snapshot of the world at that point in time, a mirror to reflect what people were listening to, interested in, and thinking about at that moment. And best of all, to me at least: for a lover of history, Carnegie Hall's timeline has featured not only the greatest and most famous, but also the less great and not-so-famous — there's a bit of everything there, from the sublime to the ridiculous.

I hope you can begin to see why I wanted to take all of this and transform it into Linked Open Data: imagine all of this, unlocked and released, to become part of the networked fabric of the web, filled with the raw materials of discovery and ready to be explored, with the potential to stretch farther beyond the walls of Carnegie Hall than we ever imagined. I decided that was a goal worth enduring a bit of brain torture as I tried to learn some Python.

Modeling the Data

My first task was to figure out what our data really had to say, to parse out the raw materials and relationships it contained. I realized that even though I'd been working with the Hall's history for 15 years, I'd never really done this kind of deep, conceptual dive into all of this performance history data. I started by looking at the biggest, most obvious "kernels" of data: our events. I needed to identify the key elements of an event, and to find a way to describe each of its components:

- **Where** → venues: the location of each event
- **When** → date/time of each event
- **Who** → names: of performers/participants, creators
- **What** → musical/creative works, or for non-performance events, what took place

A quick note about venues: you might be wondering why the venue in question wouldn't always be simply "Carnegie Hall". But Carnegie Hall actually has three different auditoriums within its walls (at one time, there were actually four), and the names for each of these have changed throughout the years. In fact one of our auditoriums, Zankel Hall (on CH's lower level) was completely gutted and rebuilt, as a totally new space, in 2003, and went through no fewer than four name changes before then: 1) Recital Hall (1891-1896); 2) Carnegie Lyceum (1896-1956, following significant interior alterations from the Recital Hall); 3) Carnegie Hall Playhouse (1956-1960); and Carnegie Hall Cinema (1960-1997). I needed to find a way to clearly identify these auditoriums and deal with the name changes. Here is a fragment, the full file can be vie wed in RDF/XML format
):

chVenues:Carnegie_Hall rdfs:label "Carnegie Hall"@en ;
owl:sameAs < http://sws.geonames.org/5111573/ > .

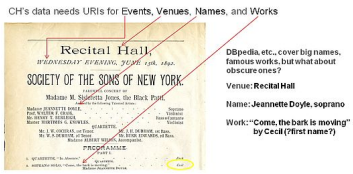
chVenues:Recital_Hall rdfs:label "Recital Hall"@en ;

geoNames:parentFeature chVenues:Carnegie_Hall .

chVenues:Carnegie_Lyceum rdfs:label "Carnegie Lyceum"@en ;
geoNames:parentFeature chVenues:Carnegie_Hall ;
geoNames:historicalName "Recital Hall"@en .

Finding URIs

Once I had identified the key components in our data and had modeled all the relationships, I needed to find URIs for them — Uniform Resource Identifiers, the stable, unambiguous, HTTP-based "names" we use for things in the world of linked data. I began by looking at some of the "big names" in linked data, well-established and widely-used data sources such as DBpedia (essentially a linked data version of Wikipedia), the Library of Congress Authorities (their subject headings, name authority file, etc.), and the Virtual International Authority File, or VIAF. I quickly realized this wasn't going to work: there were far too many obscure names and little-known musical works in our data. And the biggest problem was staring me right in the face: no other data set could possibly have identifiers for 50,000 Carnegie Hall events (or even the 17,000+ I was starting with)!



Example of a Carnegie Hall Event

My choices were either 1) cobble together a grab-bag of URIs from different sources (which would still leave many gaps), or 2) mint my own URIs. I decided to mint my own URIs. While this solution had the disadvantage of adding to a growing plethora of overlapping URIs in circulation, I felt this was outweighed by the clarity and uniformity it would bring to my data set. Also, if Carnegie Hall were recognized as a trustworthy and knowledgeable source of cultural heritage information, which I hoped we were, our URIs stood a chance of becoming "canonical" identifiers in our section of the information space, at least if people began to use them and link to them.

Best practices for LOD, as defined by the W3C's Government Linked Data Working Group, emphasize stability and persistence (although their document is still in draft form, I don't think this concept would find any argument in the LOD community). Few things are more frustrating

on the web than a broken hyperlink, so if you're going to mint your own URIs, it's a good idea to ensure that they will stick around — which means you need a stable namespace that won't disappear soon. Carnegie Hall owns the carnegiehall.org namespace, and we've made it almost 125 years, so I hope that we (along with our namespace) won't disappear anytime soon. For my URIs, I decided to create four categories, based on the key elements I'd defined in my data set (see the above example):

- **Events:** <http://data.carnegiehall.org/**events**/RH_18920615_2000>
- **Venues:** <http://data.carnegiehall.org/**venues**/Recital_Hall>
- **Names:** <http://data.carnegiehall.org/**names**/Jeannette_Doyle>
- **Works:** <http://data.carnegiehall.org/**works**/Come%2C_the_bark_is_moving_(Cecil)>

I opted to create "human-readable" URIs, within a structure that was as straightforward and transparent as possible. I added the sub-domain "data", to clearly separate the URIs from the regular Carnegie Hall website. For Events, each would be identified by a combination of a venue code + date/time.Venues and Names are self-explanatory; Names would include both event participants (individuals and organizations) as well as creators (e.g. composers, arrangers, playwrights, choreographers, etc.). Works follow a format of work title + composer name (in parentheses).

Finding Vocabularies (Predicates)

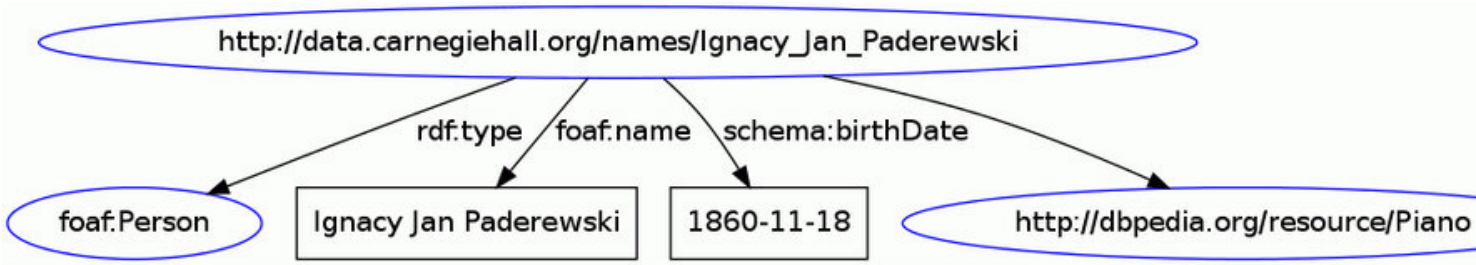
The wide range of musical and nonmusical events and performer types/roles reflected Carnegie Hall's performance history presented some challenges when choosing vocabularies for predicates. In looking at my needs for describing the what, where, when, and who of our data, in addition to straightforward terms for general concepts and labels — such as using the Resource Description Framework Schema's rdfs:type for data typing and rdfs:label for event titles — I needed a fairly broad mixture of terminology, but I tried hard to find the right balance of specificity and clarity, opting for widely-used and well-tested vocabularies wherever I could.

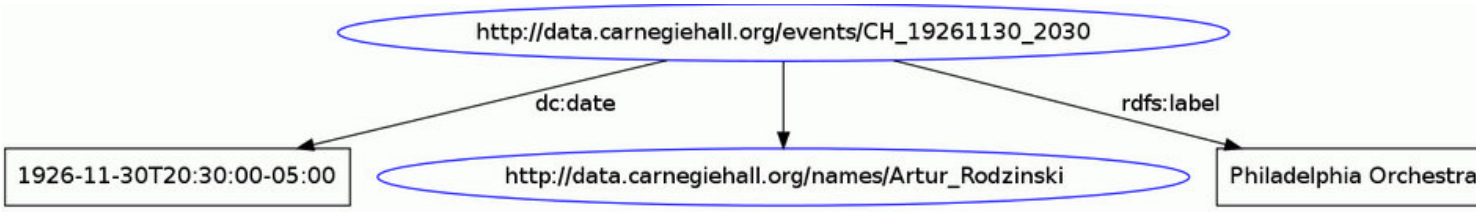
For the what and where of each event, I went with the Event Ontology , which is broad enough to cover both musical and nonmusical events (I wouldn't have to worry about a complicated process of separating and identifying these), but specific enough to make it clear we were still talking about "events". For describing when, the obvious choice was
Dublin Core
: widely adopted, well-documented, transparent, and unambiguous (I also used Dublin Core to identify composers and arrangers —
creators
of and
contributors

to — musical works).

The Music Ontology also worked well for describing who was involved with events, e.g. conductors and performers, but for performance-related data (such as associating works and venues to each event) it began to fall apart, since it was created primarily to describe recorded music. For these I used broader Event Ontology properties (on which the Music Ontology was modeled), such as event:place and event:product .

This is just a small sample, to illustrate the basics. I also used terms from the DBpedia Ontology , Friend of a Friend (FOAF) , the Gemeinsame Normdatei Ontology (from the Deutsche Nationalbibliothek), the GeoNames Ontology , Library of Congress Authorities (the MARC Code List for Relators), and Schema.org . So that you can get a better sense of how this all came together in context, here are some sample triples from my data, presented in graphic form, for an individual (pianist and composer Ignacy Jan Paderewski) and for an event (the Carnegie Hall debut of conductor Artur Rodzinski).

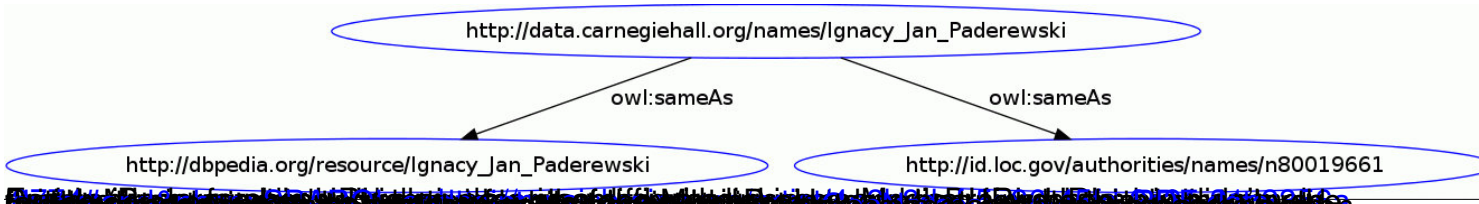




Links to Other Data Sets

So far, I've talked about how I modeled and described things and relationships within Carnegie Hall's performance history data, but of course the real power of Linked Open Data comes into play when we can link to other data sets. Since I had minted my own URIs for Carnegie Hall's data, I felt it was important to identify any existing URIs already published in major data sets and link to them using the Web Ontology Language's term owl:sameAs . I've mainly provided links to DBpedia and the Library of Congress Name Authority File:

or, in more visual form:



Samples

The RDF data can be represented in a number of data formats, and the choice often depends on the particular tool one uses to manipulate them. Fortunately, they can be all converted into each other, and also into a graphical form, using for example this handy web-based RDF format converter. You can get all the full examples referenced in the blog in canonical RDF/XML form: SameAs example, Carnegie Hall Venues, Rodzinski and Paderewski performances. Using the tool, you can convert the files into other RDF dialects and experiment with them.

Rob Hudson

Artykuł ukazał się 10 marca 2015 w *Blogu archiwistów i bibliotekarzy Instytutu Piłsudskiego*

Może Cie tez zainteresować:

- Wstęp do Linked Data
- Linked Data cz. 2: Gdzie sa dane?
- Czy jesteś Glam?
- Koperty na zdjęcia cyfrowe