

International Image Interoperability Framework (IIIF): una panoramica

Alberto Salarelli^(a)

a) Università degli studi di Parma, Parma, Italy, <http://orcid.org/0000-0001-7352-1702>

Contact: Alberto Salarelli, alberto.salarelli@unipr.it

Received: 20 October 2016; **Accepted:** 25 October 2016; **First Published:** 15 January 2017

ABSTRACT

International Image Interoperability Framework (IIIF), is a project developed since 2011 by the Bodleian Library, the British Library and the Stanford University Library in order to enhance the usability of digital images stored in archives available through the Web. Using specific API, IIIF outlines a new information ecosystem to deal with heterogeneity problems of management and access to various silos containing digital images, improving response times and operations for users. Participation in the project of some of the most important libraries shows the great interest around what could constitute a true revolution in the way to travel and enjoy the immense cultural heritage digitized already available via the Web.

KEYWORDS

IIIF; digital images; Web AAPI; Digital image repositories; digital libraries.

CITATION

Salarelli, A. "International Image Interoperability Framework (IIIF): una panoramica". *JLIS.it* 8, 1 (January 2017): 50-66 doi: [10.4403/jlis.it-12090](https://doi.org/10.4403/jlis.it-12090).

Introduzione

Lo sviluppo delle biblioteche digitali verso una dimensione sempre più multimediale è un dato di fatto difficilmente confutabile: dalle prime esperienze risalenti agli anni Novanta del secolo scorso ad oggi si è assistito ad un progressivo allargamento delle tipologie di materiali inseriti in collezioni informatizzate a conferma di come gli scenari prefigurati dei primi teorici in merito alla convergenza digitale fossero, una volta tanto, non meri esercizi di futurologia ma previsioni che lo scorrere del tempo ha dimostrato rivelarsi del tutto fondate. Tuttavia, in questa congerie sempre più eterogenea di documentazione distribuita attraverso la Rete, le immagini digitali continuano a ricoprire un ruolo prioritario sia per quanto concerne le collezioni digitali native sia per quanto attiene agli ingenti patrimoni analogici, con particolare attenzione all'ambito dei beni culturali, oggetto di innumerevoli campagne di digitalizzazione. Ed allora se l'immagine statica, in senso lato, si pone come un medium al momento insostituibile in un panorama generale di sviluppo delle digital libraries, rimangono del tutto aperte le questioni, in senso proprio, relative alle pratiche concrete di digitalizzazione, tanto in riferimento ai formati così come alle caratteristiche dei repositories nei quali esse sono conservate. In altri termini, ogni progetto digitale che comporti il trattamento di immagini persegue i propri obiettivi con specifiche soluzioni tecniche, ideate ed implementate ad hoc. Tale atteggiamento, sia chiaro, appare perfettamente legittimo e giustificabile in un'ottica centripeta ma è altrettanto ovvio come questa libertà di scelta sia foriera di una frammentazione destinata a non sfruttare in pieno le potenzialità attuali degli strumenti di recupero dell'informazione. Ci troviamo, quindi, nella paradossale situazione di un mondo di immagini digitali sempre più ricco ma, al contempo, poco agevole da percorrere in modo trasversale: il problema – insomma – della caratteristica chiusura dei silos informativi che ha giocato come leva per lo sviluppo dei Linked Open Data si presenta, sotto le medesime spoglie, anche per quanto concerne le biblioteche digitali che contemplan l'uso di immagini. E questa non è l'unica complicazione che emerge da questo panorama variegato di strumenti e soluzioni, almeno altre due vanno tenute da conto (Snydman et al. 2015, 16): i lunghi tempi di caricamento in presenza di immagini ad alta o altissima definizione (un'opzione, quella dello scaricamento in HD, offerta sempre più frequentemente dai siti Web culturali) e la difformità degli strumenti di visualizzazione disponibili per ogni singola banca dati rendono l'esperienza di navigazione attraverso queste risorse un percorso accidentato, soprattutto laddove si presentino delle esigenze specifiche, come spesso sono quelle degli utenti professionali che necessitano di strumenti di analisi approfondita in termini di identificazione, confronto, commento etc. Allo scopo di fornire una risposta efficace a queste carenze è stato sviluppato l'International Image Interoperability Framework (IIIF): una panoramica di quelle che sono le sue caratteristiche fondamentali costituisce l'oggetto di questo articolo.

IIIF: un nuovo ecosistema informativo

L'idea per una soluzione capace di scardinare le autoreferenzialità dei silos di immagini viene elaborata alla fine del 2011: secondo quanto riportato sul sito ufficiale di IIIF¹ l'abbozzo di quelle

¹ Il sito del progetto è: <http://iiif.io/>. In particolare, per quanto concerne i cenni storici sull'origine dell'idea, si veda: <http://iiif.io/news/2015/06/17/iiif-consortium>.

che sarebbero diventate le linee portanti dell'iniziativa furono schizzate su un tovagliolino di carta al termine di una cena in un ristorante cubano di Palo Alto a cui partecipavano alcuni informatici provenienti dalla Bodleian Library, dalla British Library e dalla Stanford University Library.² Come si può notare l'origine del progetto affonda le proprie radici in un ben arato sostrato biblioteconomico, a dimostrazione del fatto – se ce ne fosse ancora bisogno – che la vocazione e la tradizione dei bibliotecari a lavorare sulla base di procedure standardizzate e condivise, anche allo scopo di rendere agevole l'interscambiabilità dei dati, si rivela un valore fondamentale per lo sviluppo di risorse realmente condivisibili in Rete. Alle tre importanti istituzioni che hanno dato il via a IIF oggi se ne sono affiancate molte altre (una sessantina, al momento in cui scriviamo)³: le grandi biblioteche sono ancora il soggetto più rappresentato (in Europa, oltre alle inglesi, si notano la Bayerische Staatsbibliothek e la Bibliothèque Nationale de France), ma non mancano i più significativi progetti di biblioteche digitali confederate (Europeana, World Digital Library, DPLA) oltre a due organismi di importanza capitale per la conservazione e la trasmissione dei contenuti culturali attraverso Internet quali sono Wikimedia e Internet Archive. Alla metà del 2015 alcuni dei membri di IIF si sono consorziati allo scopo di migliorare la sostenibilità complessiva del progetto, anche in termini economici, appoggiandosi in tal senso alle competenze messe a disposizione dal Council on Library and Information Resources (CLIR).

Ma veniamo alle caratteristiche portanti di IIF e chiediamoci innanzitutto perché esso viene definito come 'framework'. Il significato di questo termine nell'ambito dei sistemi informativi è stato declinato, volta per volta, in modi differenti: strumento, modello, standard, infrastruttura. Tutte queste accezioni sono corrette ma, a nostro avviso, nel caso di IIF non colgono con esattezza nel segno. Una definizione che si presta meglio al nostro caso è la seguente:

A framework in general can be understood as a software environment designed to support future development work. It is often created for a specific development domain and normally contains a set of common and reusable building blocks so that developers can use, extend, or customize for their specific business logic (Yu 2011, 467).

Quindi, in buona sostanza, l'obiettivo di IIF consiste nell'elaborare un ambiente interoperabile in grado di permettere ai diversi software applicativi con cui si gestiscono le immagini digitali via Web di poter dialogare reciprocamente in una modalità molto più efficace di quanto oggi non avvenga, in modo da fornire agli utenti "an unprecedented level of uniform and rich access to image-based resources hosted around the world".⁴ Tradotta in pratica, la definizione di questo ambiente si concentra al momento su tre API destinate, in specifico, alla descrizione delle immagini, alla strutturazione dei repository e alle funzionalità di ricerca. Prima di esaminare in dettaglio ciascuna di queste interfacce applicative, vale la pena di spendere alcune parole per delineare le caratteristiche comuni da esse condivise.

² Qualcuno, prima o poi, dovrà cimentarsi nella scrittura di una storia delle invenzioni originariamente concepite sui tovagliolini dei ristoranti (i cosiddetti "napkin sketches"), una pratica che pare soprattutto in auge in campo informatico. In attesa di un novello Braudel che voglia imbarcarsi nell'impresa, si può leggere un sapido articolo (Brown 2011) che descrive, con opportune illustrazioni, come vennero elaborati in origine, fra bibite e resti di cibo, alcuni grandi progetti.

³ Un elenco (non completo) in: <http://iif.io/community>.

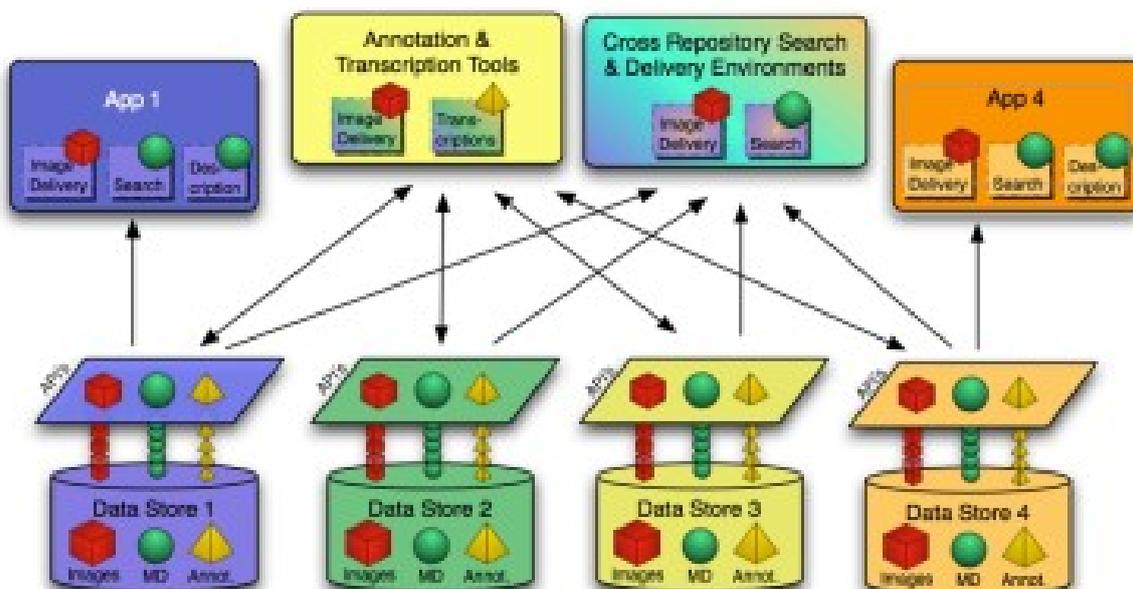
⁴ <http://iif.io/about>.

In linea di massima un'API (acronimo di Application Programming Interface) è definibile come un insieme di procedure standardizzate che consente allo sviluppatore di un determinato software di 'richiamare' all'interno di esso parti di quei programmi con cui il software stesso deve interagire, risparmiandosi così la fatica di scrivere decine e decine di righe di codice per definire, ogni volta che se ne presenti la necessità, quelle routine di base comuni a tutte le applicazioni di un medesimo applicativo. Come è facile intuire, esistono svariate tipologie di API che vanno dalle interfacce pubbliche di accesso alle librerie di software proprietari a quelle di dialogo tra applicazioni e sistemi operativi. Nel caso di IIF siamo di fronte a un'ulteriore tipologia definibile come Web API,⁵ cioè a dire un insieme di applicazioni volte a potenziare l'interazione tra client e server attraverso specifiche istruzioni scambiate tramite il protocollo 'nativo' del Web, l'HTTP. In particolare l'architettura logica delle API di IIF è basata sulle specifiche REST⁶ (Fielding and Taylor 2000) che, utilizzando i semplici metodi di richiesta previsti per in HTTP per il recupero delle pagine Web, ed identificando opportunamente le risorse con uno schema URI, permettono ai server di fornire dati strutturati ai client in formati standard aperti come HTML, XML o JSON.

Tutto ciò consente, a rigor di logica, un enorme passo avanti nella gestione delle immagini, infatti

If multiple repositories supported the same APIs for access to images, metadata and annotations, one can envision the beginnings of an ecosystem that leverages shared application code across different sites, and interoperable image-based resources, enabling simultaneous access to multiple scholarly repositories through a single interface (Snydman et al. 2015, 17).

L'ecosistema informativo preconizzato da Snydman, Sanderson e Cramer è illustrato nel loro articolo da questo schema:



⁵ http://en.wikipedia.org/wiki/Web_API.

⁶ Representational state transfer.

Come si può osservare, alla base dello schema stanno i diversi archivi in grado di esporre i loro contenuti attraverso le specifiche comuni delle API di IIIF le quali consentono alle diverse applicazioni di alto livello di fruire indifferentemente delle molteplici fonti documentali per raggiungere gli obiettivi previsti per l'utenza in termini di miglioramento dell'esperienza di visualizzazione, ricerca, confronto e annotazione delle immagini. Il tutto attraverso un'unica interfaccia.

Contestualmente, dal punto di vista di chi gestisce le collezioni, si evidenzia un indiscutibile vantaggio in termini di visibilità delle medesime, infatti “if you have images stored in a IIIF-compatible server, you could have multiple front-end discovery platforms access the images through API, either at your own institution or other institutions that would be interested in providing gateways to your content” (Magnuson 2016).

Vediamo ora di seguito, in maniera più dettagliata, le caratteristiche delle tre API elaborate all'interno del progetto.

Image API⁷

La prima API sviluppata nell'ambito del progetto IIIF riguarda l'identificazione di un'immagine e delle sue specifiche attraverso una URI veicolata tramite i protocolli HTTP o HTTPS. Questa soluzione consente ad un client di 'puntare' ad un'immagine disponibile sul Web richiedendo, contestualmente, la fornitura della medesima da parte del server con determinati parametri formali espressi in modo esplicito nella formulazione della stessa URI secondo questa sintassi:

{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}

La parte iniziale della stringa non presenta elementi di particolare novità visto che si basa, in sostanza, su quella che è la logica di indirizzamento sviluppata fin dagli albori del Web; quindi per {scheme} si intende l'utilizzo del protocollo HTTP o HTTPS; {server} è l'indirizzo Web del sito ove le immagini sono rese disponibili; {/prefix} è il percorso (directory e subdirectory) nel file system del server; per ultimo {identifier} è, ovviamente, il codice identificativo dell'immagine specificata che può essere espresso secondo diverse modalità (URN, DOI, ARK etc.).

A questo punto entrano in gioco i parametri espressamente definiti da IIIF per la fornitura delle immagini che devono essere elencati obbligatoriamente secondo l'ordine previsto:

region	
determina la porzione di immagine che deve essere fornita al client.	
<i>valori ammessi</i>	<i>Descrizione</i>
full	l'immagine è restituita completa, senza essere ritagliata.

⁷ La documentazione tecnica che descrive in dettaglio le caratteristiche e il funzionamento di questa API è disponibile all'URL: <http://iiif.io/api/image/2.1>.

square	viene fornita un'immagine quadrata il cui lato è pari al valore del lato minore dell'immagine originale.
x,y,w,h	viene restituito un ritaglio dell'immagine originale espresso in pixel: 'x' e 'y' indicano il punto di riferimento cartesiano da cui iniziare il ritaglio, 'w' e 'h' la dimensione (larghezza/altezza) del ritaglio stesso.
pct: x,y,w,h	rispetto alla voce precedente il ritaglio, in questo caso, non è basato su un computo assoluto dei pixel ma sulla percentuale espressa rispetto alla dimensione originale dell'immagine.

size	
determina il livello di ridimensionamento dell'immagine ottenuta applicando il parametro precedente.	
<i>valori ammessi</i>	<i>Descrizione</i>
full	l'immagine è restituita completa, senza essere ridimensionata.
max	l'immagine è restituita alla massima dimensione disponibile; il risultato può essere simile a quello ritornato col parametro precedente se non diversamente specificato nei metadati.
w,	viene restituita un'immagine di larga 'w' pixel; l'altezza viene calcolata automaticamente per mantenere la corretta ratio.
,h	vedi alla voce precedente, in questo caso il parametro fornito è l'altezza;
pct:n	l'immagine viene scalata percentualmente sul valore di 'n'.
w,h	l'immagine restituita corrisponde agli esatti valori in pixel indicati.
!w,h	l'immagine viene ridimensionata automaticamente ad un livello ottimale pari o inferiore ai valori indicati.

rotation	
determina il livello di rotazione dell'immagine espresso in gradi.	
<i>valori ammessi</i>	<i>Descrizione</i>
n	l'immagine è ruotata di 'n' gradi (range = 0-360, sono ammessi valori decimali).
!n	l'immagine, prima di essere ruotata, è invertita specularmente.

Quality	
determina la qualità cromatica dell'immagine.	
<i>valori ammessi</i>	<i>Descrizione</i>
color	full color

gray	scala di grigio
bitonal	bianco/nero
default	l'immagine è restituita sulla base del valore di default stabilito dal server.

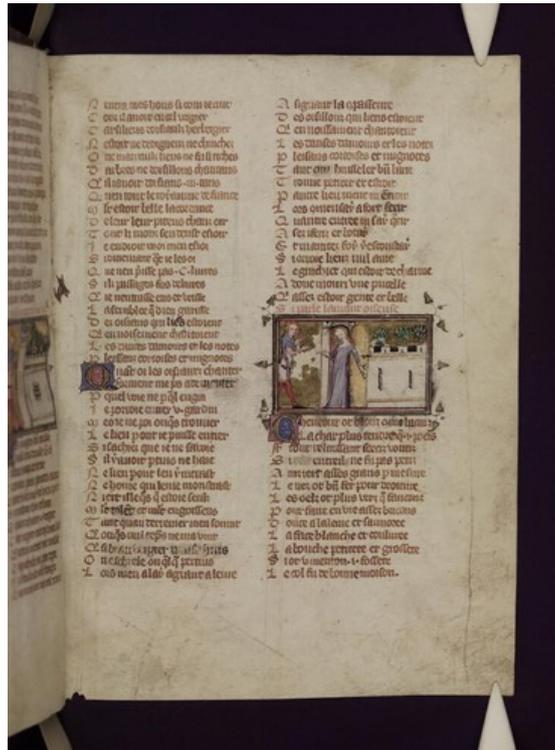
Format	
l'immagine è restituita nel formato indicato come estensione al termine della stringa.	
<i>valori ammessi</i>	<i>Descrizione</i>
jpg	image/jpeg
tif	image/tiff
png	image/png
gif	image/gif
jp2	image/jp2
pdf	application/pdf
webp	image/webp

Una possibile URI derivante dall'applicazione di questi parametri si presenta in questo modo:

<http://iiif.bodleian.ox.ac.uk/iiif/image/ec27d96e-d8dc-40a3-a205-1c0ad732949a/full/full/0/default.jpg>

Essa rimanda alla pagina (5r) di un codice contenente il *Le Roman de la Rose* (MS. Selden Supra 57, Paris 1348) conservato presso la Bodleian Library:⁸

⁸ Rights: Photo: © Bodleian Libraries, University of Oxford - Terms of Access: <http://digital.bodleian.ox.ac.uk/terms.html>.



Ora, agendo opportunamente sui valori dei parametri, possiamo – ad esempio – ‘ritagliare’ la miniatura presente a metà della colonna di destra:

.../2100,2500,1250,750/full/0/default.jpg



Ridimensionarla al 30%

.../2100,2500,1250,750/pct:30/0/default.jpg



E, magari, ruotare l'immagine di 90° e contemporaneamente convertirla in scala di grigi

.../2100,2500,1250,750/pct:30/90/gray.jpg



Naturalmente un'azione diretta sull'URI per modificare manualmente la restituzione di un'immagine può avere poco senso vista la laboriosità del procedimento. Il discorso, invero, si presenta in modo ben diverso se si ragiona in termini di potenzialità di elaborazione che, grazie alla semplice definizione ed espressione dei parametri sopra indicati,⁹ vengono offerte come basi per costruire un software. È ovvio infatti come la qualità della fruizione delle immagini sia destinata ad aumentare enormemente laddove si abbia a disposizione un modo semplice (e quindi rapido) per poter esprimere le modalità dell'immagine da restituire nell'interazione tra client e server, tenendo presenti le caratteristiche supportate dal primo (per esempio la dimensione e la risoluzione dello schermo di uno smartphone o un tablet) sulla base della gamma di possibilità gestibili dal secondo (vale a dire sulla base di come le immagini sono disponibili nel repository: quali versioni, definizioni etc.). E si pensi al vantaggio che se ne può trarre nell'ambito dei Linked Open Data grazie all'opportunità di poter usufruire di una URI che consenta non solo di identificare un'immagine ma una porzione di essa, restituibile con diversi parametri di dimensione e cromia. Tutto questo è ciò che viene reso possibile dall'utilizzo di applicazioni IIF compatibili di cui parleremo in seguito. Qui, invece, è opportuno aggiungere che la Image API di cui stiamo trattando supporta altresì la facoltà di interscambiare metadati tra server e client affinché quest'ultimo possa conoscere le specifiche di trattamento delle immagini che il repository mette a disposizione allo scopo di

⁹ In merito ai software in grado di supportare IIF si veda il relativo paragrafo nella prosiegua di questo articolo. Intanto segnaliamo la possibilità di sfogliare virtualmente il manoscritto che abbiamo utilizzato come esempio collegandosi a questo indirizzo: <http://iif.bodleian.ox.ac.uk/iif/viewer/2cc90d04-1e18-40d3-a8b2-5fb077172ee9>.

aumentare l'efficacia della transazione: tale funzionalità viene definita "image information request" ed è un servizio che, obbligatoriamente, i server IIF devono garantire; anch'esso è basato su una sintassi standard e sull'utilizzo del linguaggio JSON:¹⁰

```
{scheme}://{server}/{prefix}/{identifier}/info.json
```

Per riprendere la pagina del manoscritto bodleiano osservata poc'anzi, l'URI sarà la seguente:

```
http://iif.bodleian.ox.ac.uk/iif/image/ec27d96e-d8dc-40a3-a205-1c0ad732949a/info.json
```

Ed ecco il tracciato che il client otterrà in risposta:

```
{
  "@context": "http://iif.io/api/image/2/context.json",
  "@id": "http://iif.bodleian.ox.ac.uk/iif/image/ec27d96e-d8dc-40a3-a205-1c0ad732949a",
  "height": 5944,
  "profile": [ "http://iif.io/api/image/2/level1.json",
    {
      "formats": [ "jpg" ],
      "qualities": [ "native", "color", "gray" ],
      "supports": [ "regionByPct", "sizeByForcedWh", "sizeByWh", "sizeAboveFull", "rotationBy90s", "mirroring",
"gray" ]
    }
  ],
  "protocol": "http://iif.io/api/image",
  "sizes": [
    { "height": 185, "width": 136 },
    { "height": 371, "width": 273 },
    { "height": 743, "width": 547 }
  ],
  "tiles": [
    { "height": 256, "scaleFactors": [1,2,4,8,16,32], "width": 256 }
  ],
  "width": 4380
}
```

Riassumendo: lo scopo di una "image information request" consiste nel fornire al client "sufficient technical metadata in order to anticipate how to efficiently request images from the image server" (Snydman et al. 2015, 18), specificando la disponibilità delle differenti versioni dell'immagine in termini di formato (nel nostro caso solo e unicamente jpeg), colore (native, color, gray) e dimensioni (vedi la sezione "sizes") aggiungendo, inoltre, quali sono le modalità di manipolazione ammesse (ad esempio, sempre per il manoscritto oxoniense, l'impossibilità di poter effettuare una rotazione per un numero di gradi arbitrario rispetto ai tradizionali 90, 180, 270, cfr. il parametro "rotationBy90s").

Come regola generale, maggiore è il numero di funzionalità che il server rende disponibili, maggiore è il livello di 'compliance' che esso assume all'interno del framework, vale a dire che per poter

¹⁰ Oltre a JSON, IIF ammette (o forse sarebbe meglio dire 'raccomanda') anche l'uso di JSON-LD (JSON for Linking Data).

validare un archivio di immagini come compatibile con IIIF è necessario perlomeno adottare le specifiche minime previste dal livello '0' così come stabilito nell'apposita documentazione tecnica.¹¹

Presentation API¹²

Nel precedente paragrafo ci siamo soffermati su alcuni elementi tecnici anche allo scopo di fornire un ragguglio generale in merito alle specifiche stabilite da IIIF. Vale a dire che anche le due API che andremo a presentare ora seguono gli stessi principi stabiliti nella prima: definizione di proprietà degli oggetti digitali espresse in linguaggio JSON e accessibili tramite un'URI standard. Quindi, da questo momento in poi, cercheremo di procedere più celermente cercando innanzitutto di chiarire gli obiettivi fondamentali di queste interfacce applicative ed evitando, per quanto possibile, di addentrarci in 'technicalities'.

Partiamo dalla Presentation API il cui scopo è quello di fornire uno schema di riferimento per oggetti digitali complessi, risultanti cioè dall'aggregazione di singole immagini come, ad esempio, l'intera scansione di un libro o di un periodico. In tal caso le informazioni che il server trasmetterà al client comprenderanno la sequenza dei singoli fotogrammi e tutte le possibili opzioni in merito all'orientamento e allo sfogliamento dei medesimi, oltre all'indicazione di eventuali parti testuali (ex. una trascrizione) e/o annotazioni di approfondimento; a questi metadati di natura gestionale vanno aggiunti anche quelli relativi al copyright e alle condizioni di utilizzo. È opportuno osservare come l'architettura di questa API abbia alle spalle una riflessione teorica di natura più generale attorno alla definizione di un modello descrittivo basato sulla logica dei Linked Open Data per oggetti digitali appartenenti soprattutto (ma non esclusivamente) al novero dei beni culturali; questo modello, denominato Shared Canvas Data Model,¹³ che vede la luce in seno al gruppo di ricerca della Stanford University sulla digitalizzazione dei manoscritti medievali (Sanderson et al. 2012), è stato successivamente adottato da IIIF proprio come base per il trattamento di oggetti digitali basati su immagini tout-court, quindi senza alcun riferimento all'ambito codicologico di origine.

Gli elementi su cui è articolata questa API sono presentati nello schema raffigurato nella Figura 1. 'Manifest' rappresenta la descrizione complessiva dell'oggetto digitale e, quindi, contiene tutte le informazioni utili per la sua identificazione e per poter correlare in maniera corretta i diversi componenti;¹⁴ 'sequence', come ovvio, indica la sequenza di default (ed eventualmente le altre possibili diverse sequenze) con cui l'oggetto digitale risulta percorribile; la sequenza di default deve necessariamente essere inserita all'interno del 'manifest'; 'canvas' è una sorta di contenitore nel quale può essere inserito il riferimento ad uno specifico contenuto digitale ovvero il 'content'. Ci si potrebbe legittimamente domandare, a questo punto, perché un contenuto digitale non possa essere elencato direttamente all'interno del 'manifest' ma debba essere incapsulato in un ulteriore involucro, il 'canvas' appunto. Il vantaggio consiste nel poter disporre di un livello descrittivo

¹¹ <http://iiif.io/api/image/2.1/compliance>.

¹² <http://iiif.io/api/presentation/2.1>.

¹³ <http://iiif.io/model/shared-canvas/1.0>.

¹⁴ Per chi volesse esaminare il 'manifest' del manoscritto bodleiano, il link di riferimento è: <http://iiif.bodleian.ox.ac.uk/iiif/manifest/2cc90d04-1e18-40d3-a8b2-5fb077172ee9.json>.

ulteriore che consente al client di formattare in modo opportuno ed omogeneo tutti gli elementi che compongono un oggetto digitale complesso. Nulla vieta, come abbiamo visto parlando dell'Image API, che client e server si accordino sulle specifiche di visualizzazione di una singola immagine, ma quando le immagini sono più di una, e si vuole ottenere una certa uniformità di sfogliamento, allora diventa necessario definire dei parametri di raccordo che, appunto, vengono esplicitati nel 'canvas'.

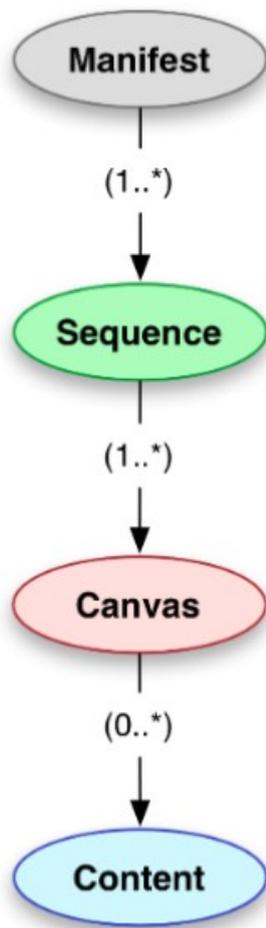


Figura 1: Elementi della API

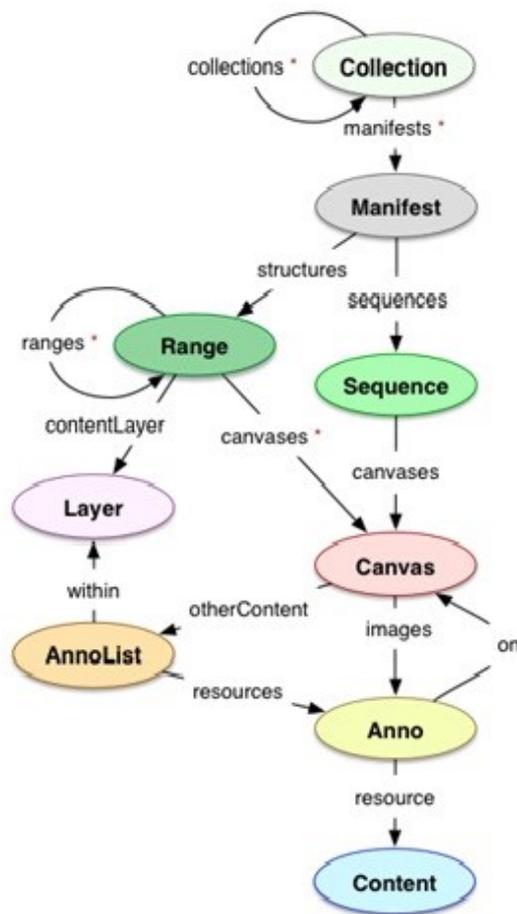


Figura 2: Rappresentazione di un oggetto digitale complesso

Si osservi ora l'ulteriore schema rappresentato in Figura 2. I quattro elementi che abbiamo elencato in precedenza rappresentano i fondamentali ('basic types') della *Presentation API*. In realtà, come è facile intuire, la descrizione di un oggetto digitale complesso può comportare la necessità di definire ulteriori entità che, nel nostro caso, vengono accorpate sotto la dicitura 'additional types'. In sintesi: 'collection' consente di raggruppare e descrivere, con gli opportuni metadati, una collezione di oggetti digitali; 'range' è un'entità che consente di elencare parti distinte di un oggetto digitale (funzione utile, ad esempio, per strutturare un indice); 'anno' sta per 'annotations': infatti in un

‘canvas’ oltre, ad esempio, all’immagine di una pagina potrebbe essere contenuta anche la trascrizione testuale della medesima o, nel caso di uno spartito, la notazione musicale. Quindi, come si diceva, il ‘canvas’ funge da contenitore in grado di raggruppare le singole immagini e i rispettivi contenuti correlati (‘annotations’) attraverso una relazione denominata ‘AnnoList’ (‘Annotation List’): come viene dichiarato nell’allegato tecnico, questa separazione delle annotazioni dalle immagini “is intended to allow clients to quickly display the images to the user, and then populate the display with further content and commentary when the user navigates to a particular canvas. It also allows the annotation list to be generated dynamically, while the manifest is static and more easily cached”.¹⁵ Le diverse annotazioni, infine, possono essere raggruppate in un ‘layer’ indipendentemente dal ‘canvas’ a cui sono riferite: non va infatti dimenticato come fra le caratteristiche salienti di questo modello di dati vi sia la completa aderenza ad un’architettura distribuita sull’intero spazio del Web. Quindi, secondo IIIF, un oggetto digitale può essere assemblato prendendo contenuti, o interi ‘canvas’, da fonti differenti e, allo stesso modo, anche le annotazioni possono essere prodotte tanto da quei soggetti che hanno la responsabilità primaria nella pubblicazione dei contenuti, quanto da altre istituzioni consimilari o da singoli utenti. Immaginiamo, in ipotesi, il lavoro di un codicologo che potrebbe creare un ‘layer’ contenente i propri commenti e annotazioni relativi ai testimoni di interesse per la propria ricerca, individuati in diverse biblioteche e visualizzati contemporaneamente nella stessa schermata del client. Il fondamento tecnico per rendere possibile tutto ciò (e sui cui, ribadiamo, si fonda questa API) è l’identificazione di ogni entità del modello tramite una specifica URI, unitamente all’utilizzo di JSON-LD (JSON for Linking Data), linguaggio il cui utilizzo, a differenza dell’API precedente (dove esso rappresentava un’alternativa possibile al JSON standard, cfr. nota 10), risulta in questa obbligatorio proprio in virtù della sua vocazione di modello di dati aperto, distribuito e interoperabile.

Content search API¹⁶

Veniamo, infine, all’API più recente sviluppata nell’ambito di IIIF. Si tratta – è opportuno sottolinearlo – di un’interfaccia applicativa ancora in fase di definizione (è stata rilasciata la versione 1.0) il cui scopo consiste nel fornire all’utente una serie di funzionalità per la ricerca testuale dei contenuti delle immagini. In particolare si fa riferimento a trascrizioni ottenute tramite OCR, commenti, annotazioni, traduzioni che possono accompagnare la scansione di una pagina testuale secondo le modalità previste nella Presentation API; naturalmente l’entità a cui viene applicata la funzione di ricerca vincola l’ambito su cui quest’ultima avrà effetto: “a service associated with a manifest will search all of the annotations on canvases or other objects below the manifest, a service associated with a particular range will only search the canvases within the range, or a service on a canvas will search only annotations on that particular canvas”.¹⁷ I parametri tecnici previsti in questa API hanno lo scopo di mettere a disposizione dell’utente una serie di funzioni quali l’evidenziazione dei termini di ricerca nell’immagine, la loro collocazione grafica nella barra di lettura della

¹⁵ <http://iiif.io/api/presentation/2.1/#annotation-list>.

¹⁶ <http://iiif.io/api/search/1.0>.

¹⁷ <http://iiif.io/api/search/1.0/#service-description>.

pubblicazione, l'autocompletamento in fase di digitazione della stringa e la comparazione dei risultati ottenuti da diverse fonti. Di fatto, al momento, ancora nessuno dei grandi attori che collaborano a IIF ha implementato la funzionalità di ricerca nei propri archivi di immagini, tuttavia cercando in Rete si può individuare qualche prototipo¹⁸ che consente di farsi un'idea sommaria dei risultati attesi dallo sviluppo di questa applicazione.

Software

Un'intera sezione¹⁹ del sito Web di IIF è dedicata ai software compatibili con le API che abbiamo descritto suddivisi fra applicazioni server e client.

Per quanto concerne le applicazioni lato server, si tratta di pacchetti (sia open source che commerciali) installabili perlopiù in ambiente UNIX/Linux (si raccomanda l'utilizzo di Apache): se, in linea generale, i server di immagini “are web servers that not only deliver images to the web, but also apply some transformation to the image to support enhanced functionality” (Snydman et al. 2015, 19), nel nostro caso specifico siamo di fronte a programmi in grado di processare le URI e le istruzioni JSON in modo da esporre le collezioni di immagini con le funzionalità previste dalle specifiche del framework IIF. Va da sé che un fattore strategico per consentire a questi server di offrire la massima efficacia in termini di fruizione è costituito dalla qualità delle immagini che, tramite essi, vengono rese disponibili ai client. Questo significa che, ad esempio, per garantire un buon effetto di zooming è necessario prevedere l'utilizzo di immagini in formati ‘multi-resolution’ come Pyramid TIFF o JPEG2000, formati cioè che contengono nel medesimo file molteplici versioni dell'immagine in differenti risoluzioni. Fra i vari software menzionati nel sito di IIF vale almeno la pena di menzionare IIPImage²⁰ per molteplici motivi: qui ne sottolineiamo due. Il primo consiste nella sua compatibilità con diversi sistemi operativi, quindi non solo Unix ma anche Mac OS X e Windows. Il secondo concerne la storia di questo software, sviluppato a partire dalla metà degli anni Novanta dalla National Gallery di Londra e dalla University of Southampton nell'ambito dei progetti europei Viseum (1996) e Acohir (1999). Originariamente concepito come un server per la gestione efficace delle immagini ad alta risoluzione via Web (e, quindi, con un occhio attento al consumo di banda passante: problema critico quando si affrontano file di dimensioni particolarmente onerose), oggi è in grado di operare su immagini multispettrali e 3D, naturalmente in piena compatibilità con le specifiche IIF. La lunga lista di istituzioni,²¹ sia nell'ambito dei beni culturali sia in quello delle scienze dure, che hanno realizzato gallerie di immagini o vere e proprie mostre virtuali grazie a IIPImage è una palese dimostrazione della sua versatilità d'uso e delle ottime prestazioni anche a fronte della manipolazione di fotogrammi estremamente complessi.

Ad ogni buon conto, è molto probabile – come ha osservato Lauren Magnuson (Magnuson 2016) – che uno degli elementi nevralgici per il futuro del progetto consisterà nello sviluppo di add-on in grado di convertire in formato IIF i metadati dei più diffusi software oggi utilizzati per gestire archivi digitali senza richiedere l'installazione di un nuovo specifico server.

¹⁸ Ad esempio qui: <http://wellcomelibrary.org/item/b19956435>.

¹⁹ <http://iif.io/apps-demos>.

²⁰ <http://iipimage.sourceforge.net>.

²¹ <http://iipimage.sourceforge.net/links>.

Entrando nel merito delle applicazioni lato client, occorre innanzitutto osservare come esse si traducano perlopiù in software anch'essi da installare su server remoti per offrire all'utente la possibilità di sfruttare a pieno titolo le potenzialità di IIIF di cui abbiamo parlato in questo articolo: effettuare zoom e rotazioni, calibrare luminosità, contrasto e saturazione, identificare e taggare specifiche aree dell'immagine, confrontare immagini provenienti da collezioni differenti. Perciò è evidente come il ruolo prioritariamente assegnato a questi applicativi sia quello di fungere da front end, cioè interfacce utente, per le installazioni lato server: in altre parole, una volta implementato un archivio di immagini con un server IIIF compatibile, è quasi inevitabile che venga installato, nello stesso luogo, un visualizzatore che consenta a tutti gli utenti che si collegano a quel sito di poter percorrere le collezioni ivi contenute senza dover imporre loro l'obbligo di scaricare e installare sui rispettivi dispositivi alcun software specifico. Poniamo il caso, ad esempio, del Rembrandt Database: questa base dati di immagini del grande maestro olandese utilizza IIPImage come server e IIPMooViewer come client; analogamente la collezione dei libri rari della Biblioteca Hertziana implementa il server digilib e l'omonimo client. Nulla vieta, naturalmente, che la scelta delle applicazioni server e client cada su prodotti differenti, come nel caso della Princeton University Library che utilizza il server Loris e il viewer OpenSeadragon.²² Quest'ultimo software è particolarmente importante poiché rappresenta una delle poche soluzioni possibili (insieme a Leaflet) per un utente che voglia installare su un proprio dispositivo uno strumento di visualizzazione di immagini divulgate secondo le specifiche IIIF, infatti OpenSeadragon è disponibile anche come componente aggiuntivo per Chrome e Firefox: una volta caricata nel browser l'immagine, essa può essere zoomata o ruotata a piacimento. Sarebbe opportuno – e questo è un auspicio che ci auguriamo possa presto diventare uno degli obiettivi di IIIF – che il numero delle applicazioni installabili e configurabili dagli utenti sui propri pc o dispositivi mobili possa aumentare, ma soprattutto che tali programmi contemplino tutte le funzionalità già presenti nei migliori visualizzatori online, come ad esempio Mirador,²³ un software open source, sviluppato – grazie a un finanziamento della Andrew W. Mellon Foundation – nell'ambito degli studi codicologici condotti presso le università di Stanford e Harvard. Per chi oggi volesse testare le potenzialità di questo programma sono disponibili alcune demo online²⁴ che, più di tante parole, forniscono un'idea particolarmente efficace di come i presupposti teorici di IIIF possano tradursi in pratica in modo efficace e, diciamo pure, suggestivo.

Conclusioni

È un dato di fatto che gli studiosi dediti allo studio dei codici e dei libri antichi e, in particolare, come abbiamo accennato all'inizio, alcune grandi biblioteche custodi di fondi rari di inestimabile pregio, abbiano rivestito un ruolo essenziale nello sviluppo di IIIF. Ancora oggi fra gli esempi più eclatanti di applicazioni del framework vi sono quelli offerti dalle istituzioni partner originarie del progetto come Stanford²⁵ o la Bodleian Library.²⁶ Tuttavia il novero dei partecipanti all'iniziativa sta

²² <http://openseadragon.github.io>.

²³ <http://projectmirador.org>.

²⁴ <http://projectmirador.org/demo>;

²⁵ <https://exhibits.stanford.edu>.

²⁶ <http://digital.bodleian.ox.ac.uk>.

rapidamente aumentando e sono sempre più le collezioni digitali rese disponibili in Rete con le specifiche di IIF fra le quali val la pena menzionare quelle della Biblioteca Vaticana che da maggio 2016 rilascia i ‘manifest’ delle opere presenti nella Digital Vatican Library²⁷ e che ospiterà a giugno 2017 uno dei meeting che periodicamente la comunità IIF organizza in giro per il mondo. Nondimeno la filosofia del progetto, e cioè la facoltà di operare sulle immagini in Web con standard condivisi, può offrire l’occasione di ideare progetti di assoluto interesse anche in situazioni ove manchi una collezione digitale da gestire in proprio. In maniera analoga a quanto avviene nell’ambito delle risorse ad accesso aperto che condividono il medesimo protocollo (OAI-PMH), dove vi sono sia fornitori di contenuti sia di servizi, così nell’ambito di IIF possono darsi casi di aggregatori tematici di risorse possedute da istituzioni altre. Un esempio paradigmatico in tal senso è il caso di Biblissima,²⁸ sito di riferimento a livello internazionale per lo studio del patrimonio scritto medievale e rinascimentale, che si sta proponendo come un hub di risorse codicologiche digitalizzate secondo i principi IIF. È scontato ritenere come la fattibilità e l’efficacia di questi progetti miglioreranno enormemente non solo al crescere del numero di istituzioni coinvolte e all’allargamento della prospettiva di ricerca dal mondo delle biblioteche a quello di tutti gli altri ambiti culturali ove le immagini digitali giocano un ruolo preponderante, ma soprattutto se verranno istituiti elenchi organizzati e aggiornati di tutte le risorse digitalizzate con i rispettivi manifesti: “the community is starting to come to grips with this issue, but at the moment it’s not terribly easy” (Albritton 2015). Se si creasse, insomma, un vero movimento di sostegno alla crescita di IIF come ‘lingua franca’ nella gestione di collezioni di immagini online potremmo avere in tempi molto brevi una vera rivoluzione nel modo di percorrere e fruire gli immensi patrimoni culturali digitalizzati già disponibili via Web. Ci sentiamo di poter affermare che le premesse affinché ciò possa accadere ci sono tutte.

Riferimenti bibliografici²⁹

- Albritton, Benjamin. 2015. “Fellow travelers: The Canterbury Tales and IIF.” *Digital Manuscripts at Stanford*, July 14, 2015. <http://web.stanford.edu/group/dmstech/cgi-bin/wordpress/author/blalbrit>.
- Brown, Bob. 2011. “Napkins: Where Ethernet, Compaq and Facebook’s cool data center got their starts.” *Network World*, July 19, 2011, <http://www.networkworld.com/article/2220218/ethernet-switch/napkins--where-ethernet--compaq-and-facebook-s-cool-data-center-got-their-starts.html>.
- Fielding Roy T., and Richard N. Taylor. 2000. “Principled design of the modern Web architecture.” In *Proceedings of the 22nd international conference on Software engineering (ICSE ‘00)*, 407-416. ACM: New York. doi:[10.1145/337180.337228](https://doi.org/10.1145/337180.337228).
- Magnuson, Lauren. 2016. “Store and display high resolution images with the International Image Interoperability Framework (IIF).” *ACRL TechConnect Blog*, February 25, 2016,

²⁷ <http://digi.vatlib.it>.

²⁸ <http://doc.biblissima-condorcet.fr/iif>.

²⁹ Ultimo accesso ai siti Web: 22 settembre 2016.

<http://acrl.ala.org/techconnect/post/store-and-display-high-resolution-images-with-the-international-image-interopability-framework-iiif>.

Sanderson, Robert, Benjamin Albritton, Rafael Schwemmer and Herbert Van de Sompel. 2012. "SharedCanvas: a collaborative model for digital facsimiles." *International Journal on Digital Libraries* 13 (1): 3-16.

Snydman, Stuart, Robert Sanderson and Tom Cramer. 2015. "The International Image Interoperability Framework (IIIF): a community & technology approach for web-based images." *Archiving Conference* (1): 16-21.

Yu, Liyang. 2011. *A Developer's Guide to the Semantic Web. Heidelberg - New York: Springer.*