

Storicamente
Rivista del Dipartimento di Storia Culture Civiltà
Alma Mater Studiorum Università di Bologna

Open Source e transmedialità

L'esperienza del Cineca nel campo del Virtual Heritage

Daniele De Luca
Antonella Guidazzoli
Maria Chiara Liguori
Micaela Spigarolo

DOI 10.1473/stor470

ISSN 1825-411X

Art. No. 25

Issue No 9 - 2013

Editore: ArchetipoLibri

July 16th 2013

Tecnostoria

Indice

Introduzione	2
1. Apa l'Etrusco e la scelta	2
2. Pratiche di riutilizzo: il progetto Marcus Caelius	8
Conclusioni e prospettive future	12
Bibliografia	12

Introduzione

L'esperienza del Cineca nella ricostruzione archeologica virtuale si è indirizzata negli ultimi anni verso la realizzazione di ambientazioni 3D filologicamente corrette per applicazioni a carattere divulgativo come il cortometraggio 3D stereoscopico «Apa l'Etrusco e 2700 anni di storia bolognese», per il Museo della storia di Bologna [<http://www.comune.bologna.it/museoarcheologico/>], ed il cortometraggio in tecnica machinima «Marcus Caelius – Il valore del ricordo», per il Museo Civico Archeologico di Bologna [<http://www.comune.bologna.it/museoarcheologico/>]. In questi progetti si è cercato di sposare al meglio l'intento comunicativo – trasmettendo cultura attraverso una narrazione [Antinucci 2010] che funga da espediente per determinare un'esperienza personale nello spettatore – la cura filologica e l'implementazione di una modalità di creazione dei modelli che ne favorisca il riutilizzo, anche in un'ottica transmediale. *L'open source*, in primo luogo l'adozione del *software* Blender, ha fornito gli strumenti più adatti non solo per gestire le problematiche inerenti la produzione, ma anche per affrontare quelle specifiche di progetto, prima tra tutte l'introduzione dei vincoli filologici nel processo di produzione della pipeline tradizionale di un filmato 3D.

1. Apa l'Etrusco e la scelta

Il Museo della Storia di Bologna – Genus Bononiae [<http://www.genusbononiae.it>] commissiona nel 2009 a Cineca un filmato 3D stereoscopico, capace di raccontare la storia della città nelle sue tappe fondamentali, e la sala immersiva dedicata alla sua proiezione.

Figura 1.



Figura 2.

Apa l'Etrusco, il personaggio del corto 3D per il Museo della Storia di Bologna, in due shot presi dal film.

L'esperienza maturata da Cineca nel campo delle ricostruzioni archeologiche [Guidazzoli 2007] e l'impiego già profuso in numerosi precedenti progetti (per esempio in collaborazione con il CNR-ITABC [Calori, et. al. 2009, 177-180]) ha consentito di selezionare gli strumenti *software* più adatti e, cosa più importante, di farli dialogare tra loro in una pipeline quasi completamente *open source*.

L'adozione di *software* Open Source come centro della catena di produzione è stata quindi una scelta caratterizzante del progetto stesso. Il *software* più significativo nel lungo elenco di risorse open a cui si è fatto ricorso è sicuramente Blender [<http://www.blender.org>], *software* di modellazione 3D analogo a prodotti come 3D Studio Max o Maya largamente usati nel mondo dell'animazione. Si tratta di un prodotto versatile che ha consentito di costruire le scenografie, realizzare i personaggi, fare le animazioni, fino ad arrivare alla produzione dei fotogrammi finali sulla Renderfarm del Cineca.

Solo per il *software* di modellazione procedurale CityEngine [<http://www.procedural.com/cityengine/features.html>] (unitamente ad alcune licenze di Microsoft Windows) – utilizzato per ricreare la Bologna attuale, quella etrusca, romana e medioevale – sono stati necessariamente acquistati contratti in licenza d'uso (EULA) dato che, ad oggi, non esistono soluzioni open che possano sostituire in parte o totalmente le funzionalità di tale programma.

La scelta di *software* rilasciato con licenza Open, oltre ad impattare positivamente sulle spese di produzione per via della sua gratuità, è motivata da molteplici considerazioni che hanno condotto il Cineca a farne una linea guida. Innanzi tutto, l'accesso al codice e la possibilità di modificarlo ha permesso, per esempio, di ricompilare Blender per impiegarlo all'interno dei *cluster* Cineca in forma di *render farm* e di realizzarne una versione che includesse le funzionalità di *stereo-target* e *stereo-distance* per la camera stereo con cui sono state effettuate le riprese stereografiche (il progetto, infatti, è antecedente al rilascio della versione 2.5x del *software* che incorpora oggi tali funzionalità).

La scelta *open source* è inoltre risultata particolarmente utile nel garantire la riusabilità dei modelli. Impiegare formati aperti, infatti, consente di essere sempre in grado di riutilizzare i modelli prodotti anche a distanza di anni. La tecnologia di virtualizzazione, inoltre, propone numerose soluzioni *open source* robuste e mature (per esempio, le macchine virtuali KVM [http://www.linux-kvm.org/page/Main_Page] o User-Mode Linux [<http://user-mode-linux.sourceforge.net/>]) che consentono di congelare lo stato di un sistema (modello, *hardware* emulato e *software*) in modo da garantirne l'accessibilità *ad interim*.

Non da ultimo, soluzioni *open source* includono un'ottima reperibilità, una buona compatibilità, la facilità d'uso e d'installazione e la possibilità di fare affidamento su una *community* vivace ed un significativo *corpus* di materiale documentale. Per gli studenti che seguono i corsi di formazione al Cineca o che hanno svolto stage inerenti il progetto, questo ha significato apprendere conoscenze immediatamente riutilizzabili.

Come valore aggiunto, l'approccio *open source* ha consentito di affrontare il progetto in termini di Open Project di Blender [<http://www.blender.org/features-gallery/blender-open-projects/>], ovvero di studiare all'interno della pipeline di produzione reale funzionalità e caratteristiche che potessero essere di interesse in termini generali, ponendo l'accento sull'aspetto di "architettura della partecipazione" [<http://www.oreillynet.com/lpt/wlg/3017>] (che oggi rappresenta l'*open source* in maniera più significativa della condivisione stessa del codice).

1.1 Open source per la gestione della pipeline tradizionale

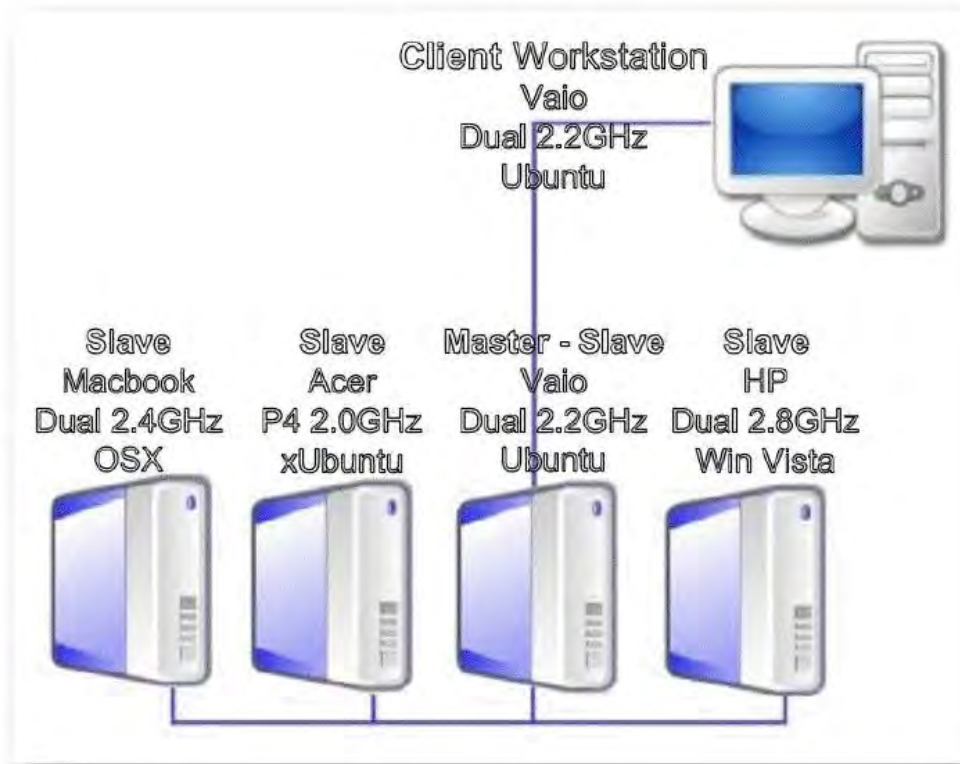
I *software* impiegati nella pipeline di produzione di "Apa" sono prevalentemente Ubuntu Linux [<http://www.ubuntu.com/>], Blender [<http://www.blender.org>] e The GIMP [<http://www.gimp.org>]. Ad essi sono stati affiancati Grass [<http://grass.fbk.eu>] e Qgis [<http://www.qgis.org/>] per la ricostruzione dei terreni a partire dalle georeferenziazioni; MeshLab [<http://meshlab.sourceforge.net/>] e Arch 3D [<http://www.arc3d.be/>] per la gestione dei dati acquisiti tramite scanner (p.e. Il Nettuno)

Ubuntu Linux

Ubuntu [<http://www.ubuntu.com/>] è una distribuzione Linux nata nel 2004, basata su Debian [<http://www.debian.org/>], che si focalizza sull'utente e sulla facilità di utilizzo. È corredata da un'ampia gamma di applicazioni libere, le quali sono scaricabili ed installabili gratuitamente e senza alcuna procedura di configurazione: sono presenti oltre 30.000 pacchetti *software* liberi dell'universo Linux, tra cui programmi come Inkscape [<http://inkscape.org/index.php?lang=it>] (grafica vettoriale) e i già citati GIMP-(fotoritocco) e Blender (grafica 3D) e molti altri. Per tale motivo Ubuntu è stata la piattaforma principale su cui sviluppare il progetto.

Il sistema operativo Ubuntu Linux e le sue derivate Xubuntu [<http://www.xubuntu.org/>] e Lubuntu [<http://lubuntu.net/>], preferite grazie all'utilizzo di gestori *desktop* più performanti e leggeri, hanno consentito di far fronte all'invecchiamento *hardware* rendendo nuovamente utilizzabili macchine datate al fine, ad esempio, di creare *network-renderfarm* per Blender per i test di *rendering* parziali ed in alternativa alla *renderfarm* di progetto.

Figura 3.



Esempio di *renderfarm* creata mediante Blender+Ubuntu con *hardware* non recente. Potenza di calcolo raggiunta 16.8GHz con 7 core a disposizione [De Luca 2010].

Blender

Blender [<http://www.blender.org/>], per gli scopi realizzativi perseguiti con “Apa”, ed in seguito con altri progetti, si è rivelato il *software* più adatto: è stato utilizzato non solo per la creazione delle scenografie e dei personaggi ma anche per l’animazione, l’illuminazione, il *rendering*, il *rigging* ed il *compositing*. Questo software dispone, inoltre, di funzionalità per mappature UV, simulazioni di fluidi, rivestimenti, particelle, altre simulazioni non lineari e creazione di applicazioni/giochi 3D grazie al *game engine* integrato. È disponibile per vari sistemi operativi, tra cui: Microsoft Windows, Mac OS X, Linux.

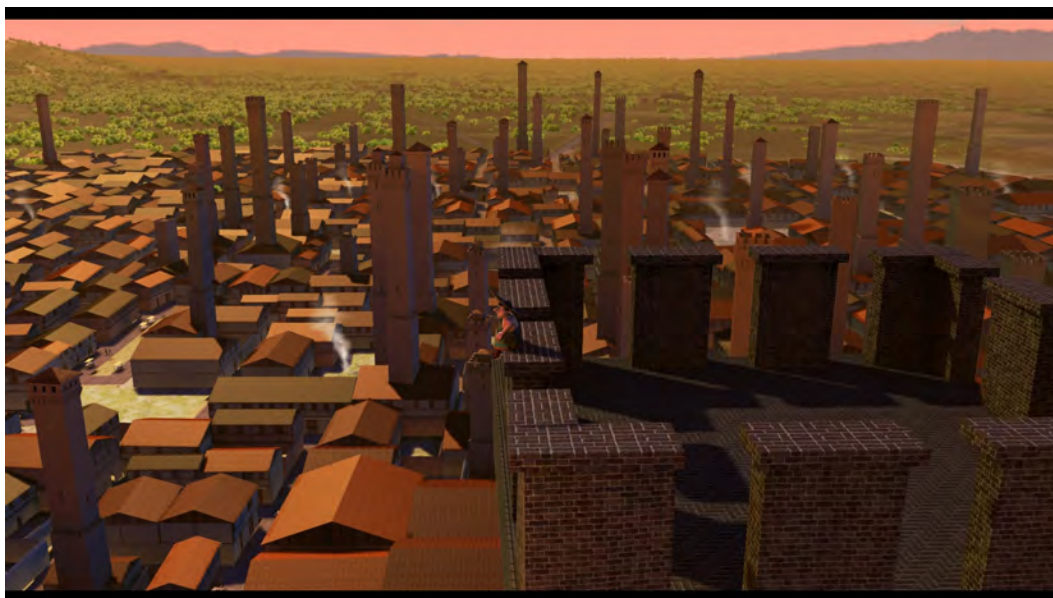
Blender è dotato, quindi, di un robusto insieme di funzionalità paragonabili, per caratteristiche e complessità, a programmi proprietari per la modellazione 3D come Softimage XSI, Cinema 4D, 3D Studio Max, LightWave 3D e Maya. Tra le funzionalità di Blender vi è anche l'utilizzo di *raytracing* e la possibilità di customizzarne il comportamento tramite *script* (in linguaggio Python [<http://www.python.it/>]). Avvalendosi di questa opportunità è stato possibile implementare una versione del programma che comprendesse le funzionalità della camera stereo (*stereo target* e *stereo distance* per le riprese in stereoscopia), non presenti nella versione 2.49 impiegata per il progetto. La peculiarità di accesso al codice ha inoltre consentito di ricompilare il programma per poterlo integrare nel *cluster* di super calcolatori del Cineca al fine di realizzare la Renderfarm che ha renderizzato appunto l'intero filmato.

The Gimp

The GIMP [<http://www.gimp.org/>] (GNU Image Manipulator Program) è un *software* professionale di fotoritocco ed elaborazione di immagini 2D. Supportando *addon* e *plugin*, è in grado di estendere in modo teoricamente illimitato le proprie potenzialità e funzioni. Nell’ambito del progetto, GIMP è stato utilizzato nell’elaborazione delle *texture* e nel loro adattamento alle *mesh* 3D. Tale strumento ha permesso, inoltre, di effettuare alcuni studi sui colori e sui *layout* finali mediante *mockup*, cioè raffigurazioni fittizie delle scene finali utili a “rendere l’idea” ed utilizzate come linea guida in termini di resa, atmosfera, impatto e cromia per la successiva modellazione della scena.

Figura 4.



Figura 5.

Esempio di mock-up realizzato da Enrico Valenza [<http://www.enricovalenza.com>], Art Director del cortometraggio “Apa”, per la Bologna medioevale; resa definitiva della ricostruzione 3D.

Attualmente The GIMP si trova in una versione stabile 2.6.x, ed una versione in sviluppo (la 2.7.x che convoglierà il suo *development* nella futura 2.8). Quest'ultima introduce notevoli miglioramenti e funzionalità, di modo tale da uniformarsi a *software* professionali come Photoshop e Paint Shop Pro. Vengono implementate anche funzionalità studiate per la UserExperience andando dalla presentazione dei comandi all'immediatezza di alcune funzioni (VD SINONIMI). Il *software*, pur soffrendo qualche lacuna in termini di supporto e controlli della cromia avanzata per la stampa digitale o la tipo/seri-grafia, può essere definito come professionale nell'implementazione delle sue altre peculiarità. Ultimo rilevante aspetto, trattandosi di un programma *free ed open source* è eventualmente modificabile ed estendibile a piacimento, in ogni sua forma e non solo mediante l'aggiunta di *addon* o *plugin*.

City Engine

CityEngine [<http://www.procedural.com/cityengine/features.html>] è l'unico strumento proprietario inserito nella pipeline di produzione. Attraverso l'elaborazione procedurale di mappe reali e dati *gis*, City Engine consente di ricostruire rapidamente interi spazi urbani. Questo ha permesso di realizzare ricostruzioni della città di Bologna attuale, etrusca, romana e medioevale.

Il *software* permette una malleabilità totale del modello procedurale, anche a partire da dati GIS (Geographic(al) Information System), consentendo l'acquisizione, la registrazione, l'analisi, la visualizzazione e la restituzione di informazioni derivanti da dati geografici (geo-riferiti). Ottenuti i dati georeferenziati storici e quelli attuali, riguardanti le principali strade, gli edifici chiave del complesso urbano e, se disponibile, anche la vegetazione presente, il *software* si occupa semi-automaticamente, guidato mediante un semplice linguaggio di programmazione interno, della creazione delle costruzioni, delle strade storicamente fedeli e di tutti i caratteri salienti della città. Successive manipolazioni dei dati GIS portano all'aggiornamento da parte del programma delle zone interessate da modifiche, riprocessando e ricalcolando i parametri per adattare le costruzioni. Modifiche specifiche possono riguardare costruzioni chiave da inserire in zone ben definite. In tali casi CityEngine dà spazio alle personalizzazioni ed a tali rifacimenti, adattandosi a qualunque tipo di modifica intrapresa. Attualmente non esiste alternativa *open* altrettanto potente.

Figura 6.

Ricostruzione procedurale della Bologna etrusca realizzata con il sw City Engine.

Pur trattandosi di *software* proprietario, City Engine è stato integrato nella pipeline di produzione grazie alla possibilità di esportarne i risultati in formato OBJ, a sua volta importabile in Blender.

1.2 Open source per la gestione della pipeline integrata

Come già sottolineato nell'introduzione, la realizzazione del filmato ha sotteso la duplice sfida – comunicativa ed implementativa – che si riassume in due problematiche principali: l'integrazione di vincoli filologici nella pipeline di produzione tradizionale di un filmato 3D stereo e l'implementazione di una modalità di creazione dei modelli che ne favorisca il riutilizzo, anche in un'ottica transmediale.

La gestione dei vincoli filologici ha influenzato innanzitutto l'implementazione (e gestione) del *repository* (deposito centralizzato dei *file*) per la lavorazione e il mantenimento degli *asset* (ovvero, qualsiasi elemento costituente uno scenario 3D) che ha accolto anche tutto il materiale documentale. Il vincolo filologico delle ricostruzioni, infatti, implica una modellazione basata su materiale di riferimento accurato (a partire, quando possibile, dall'oggetto e dal confronto con esperti del settore e testi). L'approccio tradizionale alle fonti (*de visu*, bibliografia, incontro con esperti) è stato integrato con le opportunità fornite dal Web e, per esempio, i motori di ricerca, le mappe *online* e le fonti digitalizzate. I maggiori servizi utilizzati ai fini del corto sono stati:

1. Reperimento di fonti con motori di ricerca, in primis Google [<http://www.google.it/>] e Bing [<http://it.bing.com/>];
2. Reperimento di immagini e foto mediante Google Immagini;
3. Confronto di mappe e *location* grazie a Google Maps [<http://maps.google.it/>] e Bing Maps [<http://www.bing.com/maps/>];
4. Interrogazione di Wiki ed altre enciclopedie *online*;
5. Uso di Wikimapia [<http://wikimapia.org/>] per effettuare dei *match* tra le *location* e le relative informazioni storiche.

1.3 Cloud computing

L'approccio al progetto come *open project*, ed in un contesto lavorativo *d'equipe* (tra gli altri, Spark DE [<http://www.sparkde.com/intro.html>] per l'animazione e Lilliwood [<http://www.lilliwood.eu/>] per la stereoscopia), ha evidenziato la necessità di implementare servizi di supporto alla collaborazione tra *team* eterogenei, geograficamente distribuiti e con specifiche IT differenti.

Ciascuna di queste realtà, infatti, possedeva già una propria modalità di lavoro, talvolta imposta dalla specifica IT di supporto implementata dall'azienda.

Spark DE, che ha curato l'animazione, per la gestione del ciclo di produzione (ciclo di vita) degli *asset* si appoggia ad un *software* implementato internamente (SPAM) e basato su tecnologia NFS che non è stato possibile importare in Cineca ove, per via dell'architettura intrinseca delle macchine di super calcolo, si è preferito adottare una soluzione basata su RapidSVN [<http://rapidsvn.tigris.org/>]. Si tratta di un *software* di versionamento *open* generalmente impiegato per mantenere le varie release del codice, e che può essere ugualmente applicato ad altri tipi di *file*. In particolare, le sue funzionalità rimangono invariate per la gestione di file di testo come documentazione o i manuali. Nel nostro caso, però, è stato utilizzato per il versionamento di ampi *file* binari, ovvero gli *asset* delle scene realizzati con Blender. Questo ha consentito di automatizzare compiti come:

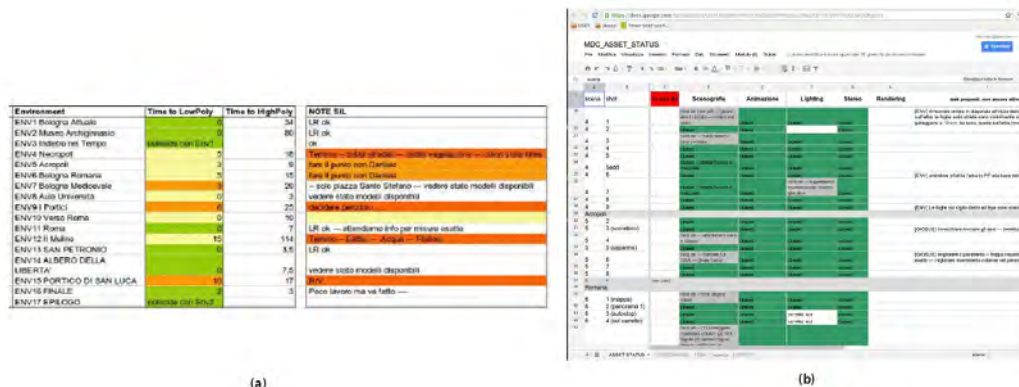
- tenere traccia di chi sta modificando che cosa;
- unire le modifiche una volta completate;
- segnalare eventuali conflitti;
- conservare una copia di ogni versione e fare in modo che sia facilmente recuperabile.

A RapidSVN è stato affiancato un *repository* centralizzato su un *cluster* di super calcolo interno (PLX).

Al contempo, è stato necessario implementare uno strumento di gestione unificata del ciclo di vita degli *asset*. Per i file binari, infatti, Rapid SV non è in grado di realizzare automaticamente il *merge* di modifiche simultanee al medesimo *file* (come fa, invece, per i file di testo). Senza uno strumento di coordinamento degli accessi si rischiava, pertanto, di perdere *in toto* il lavoro dell'autore che avesse salvato per primo il proprio *file*, in quanto sarebbe stato completamente sovrascritto dall'autore successivo.

Lo strumento di coordinamento necessario per ovviare a questo inconveniente è stato realizzato avvalendosi della forma di *cloud computing* implementata da Google. Un Google Doc denominato Assets Status ha consentito, infatti, di coordinare e condividere in tempo reale la produzione ed il suo stato di avanzamento.

Figura 7.



Asset status: a) Porzione del documento sull'organizzazione delle scene in sviluppo per il cortometraggio [De Luca 2010]; (b) evoluzione del documento allo stato finale di produzione.

L'attivazione di un Blog di progetto [<https://rvn05.plx.cineca.it:12001/php/MDC/portal/wordpress>] – realizzato con Wordpress, una piattaforma *open* di CMS (Content Management System) – ha fornito ai partecipanti un ulteriore spazio di condivisione per mettere in comune idee e documentazione di pubblica utilità, come *tutorial* e guide, oltre a dare visibilità esterna al progetto mediante la pubblicazione di *news* ed eventi.

2. Pratiche di riutilizzo: il progetto Marcus Caelius

L'approccio *open* non influenza solo le scelte relative al *software*, ma diventa un fulcro concettuale in grado di caratterizzare anche l'uso dei modelli, inserendoli in una prospettiva comunitaria di riutilizzo. Così come numerose ricostruzioni impiegate per la realizzazione di "Apa" sono l'elaborazione di creazioni precedenti, prodotte per

progetti come il Museo Virtuale della Certosa [<http://www.certosadibologna.it/>] e NUME [Bocchi, et. al. 2011], allo stesso modo modelli sviluppati per “Apa” hanno già trovato la via del riutilizzo. È il caso, per esempio, della ricostruzione procedurale della Bologna romana di epoca augustea (Bononia) modificata nel *rendering* in una nuova resa visiva coerente con i differenti intenti comunicativi per il Progetto “Marcus Caelius – Il valore del ricordo”, un corto di animazione realizzato con tecnica Machinima e attualmente in fase di elaborazione da parte del Museo Civico Archeologico di Bologna¹ e Cineca.

Figura 8.

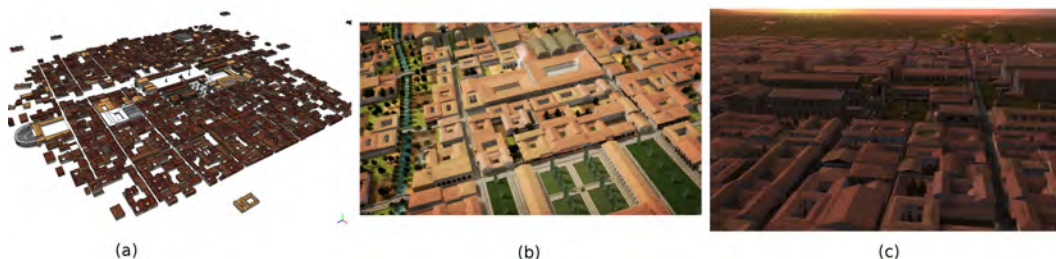


La stele di Marcus Caelius conservata al museo di Bonn.

¹Il gruppo di lavoro del Museo Archeologico è composto dal direttore Paola Giovetti e da Laura Bentini, Federica Guidi e Marinella Marchesi, che hanno collaborato nella stesura, composizione e sviluppo del progetto, in particolare per quanto riguarda gli aspetti storico-archeologici, l'individuazione dei reperti del Museo e la corretta ricostruzione filologica dei contesti. Le musiche originali del cortometraggio sono invece di Alessandro Pirotti.

Il *concept* del filmato, che avrà una durata di circa otto minuti, deriva da un noto avvenimento storico: nel 9 D.C., durante il processo di espansione verso Nord dell'Impero romano, tre legioni, guidate da Publius Quintilio Varo, furono condotte in un'imboscata e distrutte da tribù germaniche. L'evento, trasmessoci dallo storico Tacito, è noto come *clades Variana*, o Battaglia di Teutoburgo. La trama del corto trae ispirazione dal cenotafio del centurione bolognese Marcus Caelius, attualmente conservato presso il Museo di Bonn, e racconta il momento in cui al fratello Publius giunge notizia della sconfitta militare e della morte di Marcus.

Figura 9.



Tre versioni della Bologna romana: (a) la ricostruzione con City Engine a partire dai dati Gis, (b) l'elaborazione di questa ricostruzione per il filmato «Apa»; (c) e per il Progetto Marcus Caelius.

Come noto le ricostruzioni 3D sono un compito particolarmente impegnativo. Logica vorrebbe, quindi, che i modelli, in particolare quelli già sottoposti ad un rigoroso percorso di validazione storica o archeologica, venissero condivisi e riutilizzati in diverse produzioni. Lungo questo percorso si sta muovendo il Cineca, anche con l'impegno di creare un *repository* per modelli 3D, da condividere con i partner del *network* europeo V-Must [<http://www.v-must.net>] (Virtual Museum Transnational Network) secondo il Framework WP4.

L'obiettivo del filmato "Marcus Caelius" è divulgare contenuti di carattere storico-archeologico scientificamente ineccepibili attraverso un prodotto appetibile ad un pubblico di giovani, quello solitamente più disattento alle attività del Museo e più assente dalle sue sale, al di là dell'ambito scolastico.

Proprio per questo si è cercato di creare una sceneggiatura che prevedesse la possibilità di inserire nella trama e nella scenografia materiali archeologici effettivamente conservati nelle collezioni del Museo, permettendo così un continuo confronto tra il reale e il virtuale. Anche lo sviluppo dei personaggi, per lo più realmente attestati, tiene conto delle abitudini, dei rituali e delle consuetudini sociali della Bologna augustea.

Figura 10.



Un esempio del processo di modellazione dei reperti archeologici del Museo civico: l'oggetto nella collocazione abituale; la sua acquisizione fotografica; uno screenshot durante la modellazione; il rendering con Blender; l'oggetto ricostruito ed inserito nella scena.

Anche in questo caso la soluzione è stata trovata nell'uso di soluzioni *open*, con una indubbia centralità del *software* Blender.

La pipeline di produzione del cortometraggio, in questo caso, è stata integrata con *software* e tecniche di produzione di filmati, anche eterogenei tra loro, al fine di ridurre l'onere di tempo e costi necessari alla creazione di *avatar* e all'animazione delle scene.

Gli oggetti e parte dei *set* ricreati tramite Blender sono stati integrati con i restanti *set*, i personaggi, l'abbigliamento e le animazioni realizzate attraverso un motore di *rendering* di gioco, che consentisse di impiegare *machinima* (*machine cinema*), una tecnica per fare video e film in animazione tridimensionale. Questa tecnica, sfruttando in presa diretta gli ambienti e i personaggi di mondi virtuali (p.e. metaversi come OpenSim o SecondLife, o videogiochi come the Sims3 di EA), non richiede una lunga e costosa elaborazione di *rendering*.

Chiave della gestione comune degli *output* generati dai differenti *software* (*rendering* Blender e animazioni nel motore di gioco) è Blender stesso, che integra la capacità di montaggio del *chroma-key* – *tecnica di far recitare attori su sfondo monocromatico (blu o verde) per poter aggiungere in un secondo momento l’ambientazione o uno sfondo*. Nel nostro caso, consentirà di montare le animazioni girate nel gioco sui set renderizzati in Blender.

Figura 11.



Uno screenshot ripreso dentro il metaverso The Sims 3 durante la personalizzazione del personaggio come antico romano della Bologna augustea.

Figura 12.



Fasi di lavorazione eseguite come test per verificare il compositing tra l’animazione in machinima e il set renderizzato con Blender: (a) uno shot delle riprese in chroma key nel motore di gioco The Sims 3 e (b) uno screenshot di editing in Blender.

Il progetto ha forti potenzialità, soprattutto in ambito didattico e formativo: la divulgazione della particolare ed innovativa tecnica di produzione messa in opera per la creazione del cortometraggio può portare, infatti, al coinvolgimento attivo di adolescenti e giovani nella realizzazione di nuove storie, scenari o personaggi, facendo avvicinare all’ambito storico e museale fasce d’età generalmente poco interessate a questi contesti, permettendone la fruizione attraverso un punto di vista non consueto. Particolarmente importante a questo fine è l’uso del machinima, strumento che presuppone di per sé la creazione condivisa di contenuti.

Punto di forza di questa esperienza è certamente l’interdisciplinarietà tra ambiti che di solito non vengono accomunati o che addirittura non sono percepiti dai più come avvicinabili: da una parte il Museo, idealmente identificato con la statica esposizione degli oggetti e la catalogazione delle collezioni da parte degli specialisti, dall’altra il mondo dei computer – del Web e dei videogiochi – intrinsecamente interattivo, in cui tutti gli utenti possono avere un ruolo fondamentale nella creazione e nella divulgazione di nuovi contenuti.

La pipeline di produzione sviluppata per il progetto “Marcus Caelius” porta ad una drastica riduzione dei costi, ed il riutilizzo di modelli già esistenti contribuisce in maniera significativa al perseguimento di tale obiettivo. Naturalmente anche i nuovi modelli realizzati in occasione di questa produzione entreranno a far parte del

summenzionato *repository* 3D, pronti ad essere sfruttati in nuovi progetti, come “The Four Corners”, una realizzazione europea di V-Must prevista per il 2013 e relativa all’Impero Romano.

Conclusioni e prospettive future

Le recenti esperienze Cineca nella ricostruzione di ambienti virtuali filologicamente corretti e comunicativamente accattivanti hanno rinforzato la fiducia nell’approccio *open source* come risorsa versatile ed efficace, in grado di agevolare anche le piccole produzioni grazie all’abbattimento dei costi per le licenze *software*. In particolare, l’adozione di Blender ha consentito la gestione delle problematiche inerenti la produzione e quelle specifiche di progetto, introducendo, per esempio, i vincoli filologici nel processo di produzione della pipeline tradizionale di un filmato 3D. Non solo, ha permesso l’apertura dei progetti alle *community* già esistenti e la nascita di nuove. Ha favorito inoltre il concepimento di un *repository* 3D aperto, da offrire a sostegno della creatività, pensato per agevolare un immediato riutilizzo in senso transmediale dei modelli, che potranno essere adattati a differenti ambiti, oltre che a differenti piattaforme, a seconda dei messaggi che si desidera veicolare.

L’approccio cooperativo, decentralizzato e fondato sulla libera aggregazione consente alle persone di realizzare quello che non sarebbero in grado di fare da sole e rappresenta oggi la carta vincente per l’innovazione. L’approccio *open source*, per sua natura affine a questo processo, fornisce gli strumenti più adatti per incentivare e far crescere questa attitudine.

Figura 13.



Logo progetto europeo V-Must.

Bibliografia

Antinucci F. 2010, *Comunicare nel museo*, Roma: Percorsi Laterza.

Bocchi F., Smurra R. (eds.) 2011, *La storia della città per il museo virtuale di Bologna*, Bologna: Bononia University Press.

Calori L., Camporesi C., Pescarin S. 2009, *Virtual Rome: a FOSS approach to WEB3D*, in Spencer S. N. (ed.) 2009, *Proceedings of the 14th International Conference on 3D Web Technology (Web3D '09)*, New York: ACM, 177-180.

De Luca D. 2010, *Cortometraggio 3D per il progetto Cineca MDC – Museo della Città di Bologna: modellazione, rendering e studio delle fonti storiche*, Corso di Laurea in Scienze di Internet Specialistica, Università degli studi di Bologna.

Guidazzoli A. 2007, *L’esperienza del CINECA nel campo della Virtual Archaeology*, in Colarini A., Scagliarini Corlàita D. (eds.) 2007, *UT NATURA ARS. Virtual Reality e archeologia. Studi e Scavi (22)*, Imola: University Press Bologna, 81-89.