Digitisation Fundamentals and their Application

Robin Davies
Calleigh Lim
Welcome to DHSI 2018!

Thanks for joining the DHSI community!

In this booklet, you will find essential course materials prefaced by some useful information about getting settled initially at UVic, finding your way around, getting logged in to our network (after you’ve registered the day before our courses begin), and so on.

Given our community’s focus on things computational, it will be a surprise to no one that we might expect additional information online for some of the classes - your instructors will let you know - or that the most current version of all DHSI-related information may be found on our website at dhsi.org.

To access the DHSI wifi network, simply go into your wireless settings and connect to the “DHSI” network and enter the password “dhsi2018”.

And please don’t hesitate to be in touch with us at institut@uvic.ca or via Twitter at @AlyssaA_DHSI or @DHInstitute if we can be of any help....
The 2018 schedule is just about ready! A very few things to confirm, add, etc, but this is the place to be to find out what is happening when / where …

### Sunday, 3 June 2018 [DHSI Registration + Suggested Outings]

If you’re here a day or two before we begin, or staying a day or two afterwards, here are a few ideas of things you might consider doing …

- **Suggested Outing 1, Botanical Beach** (self-organised; car needed)
  
  A self-guided visit to the wet, wild west coast tidal shelf (and historically-significant former research site) at Botanical Beach; we recommend departing early (around 8.00 am) to catch low tide for a better view of the wonderful underwater life! Consider bringing a packed lunch to nibble-on while looking at the crashing waves when there, and then have an afternoon drink enjoying the view from the deck of the Port Renfrew Hotel.

- **Suggested Outing 2, Butchart Gardens** (self-organised)
  
  A shorter journey to the resplendently beautiful Butchart Gardens and, if you like, followed by (ahem) a few minutes at the nearby Church and State Winery, in the Saanich Peninsula. About an hour there by public bus from UVic, or 30 minutes by car.

- **Suggested Outing 3, Saltspring Island** (self-organised; a full day, car/bus + ferry combo)
  
  Why not take a day to explore and celebrate the funky, laid back, Canadian gulf island lifestyle on Saltspring Island. Ferry departs regularly from the Schwartz Bay ferry terminal, which is about one hour by bus / 30 minutes by car from UVic. You may decide to stay on forever …

- **Suggested Outing 4, Paddling Victoria’s Inner Harbour** (self-organised)
  
  A shorter time, seeing Victoria’s beautiful city centre from the waterways that initially inspired its foundation. A great choice if the day is sunny and warm. Canoes, kayaks, and paddle boards are readily rented from Ocean River Adventures and conveniently launched from right behind the store. Very chill.

And more!

Self-organised High Tea at the Empress Hotel, scooter rentals, visit to the Royal BC Museum, darts at Christies Carriage House, a hangry breakfast at a local diner, whale watching, kayaking, brew pub sampling (at Spinnaker’s, Swans, Moon Under Water, and beyond!), paddle-boarding, a tour of used bookstores, and more have also been suggested!

### Monday, 4 June 2018

Your hosts for the week are Alyssa Arbuckle, Ray Siemens, and Dan Sondheim.

### Early Class Meeting: 4. [Foundations] DH For Department Chairs and Deans (Hickman 120, Classroom)

Further details are available from instructors in mid May to those registered in the class. Registration materials will be available in the classroom.

### DHSI Registration (MacLaurin Building, Room A100)

After registration, many will wander to Cadboro Bay and the pub at Smuggler’s Cove OR the other direction to Shelbourne Plaza and Maude Hunter's Pub OR even into the city for a nice meal.

### Welcome, Orientation, and Instructor Overview (MacLaurin A144)
Classes in Session (click for details and locations)

3. [Foundations] Making Choices About Your Data (MacLaurin D109, Classroom)
4. [Foundations] DH For Department Chairs and Deans (Hickman 120, Classroom)
5. [Foundations] Introduction to Javascript and Data Visualization (Clearihue D132, Classroom)
6. [Foundations] Introduction to Computation for Literary Criticism (Clearihue A105, Lab)
7. Out-of-the-Box Text Analysis for the Digital Humanities (Human and Social Development A160, Lab)
8. Sounds and Digital Humanities (MacLaurin D111, Classroom)
9. Digital Humanities Pedagogy: Integration in the Curriculum (MacLaurin D016, Classroom)
10. Text Processing - Techniques & Traditions (McPherson Library A003, Classroom)
11. 3D Modelling for the Digital Humanities and Social Sciences (MacLaurin D010, Classroom)
12. Conceptualizing and Creating a Digital Edition (MacLaurin D103, Classroom)
13. Visualizing Information: Where Data Meets Design (MacLaurin D107, Classroom)
14. Introduction to Electronic Literature in DH: Research and Practice (MacLaurin D115, Classroom)
15. Race, Social Justice, and DH: Applied Theories and Methods (MacLaurin D105, Classroom)
16. XML Applications for Historical and Literary Research (Clearihue A103, Lab)
17. Processing Humanities Multimedia (Human and Social Development A150, Lab)
18. Digital Games as Tools for Scholarly Research, Communication and Pedagogy (MacLaurin D110, Classroom)
19. Web APIs with Python (Human and Social Development A170, Lab)
20. Ethical Data Visualization: Taming Treacherous Data (MacLaurin D101, Classroom)
21. Digital Publishing in the Humanities (Clearihue D131, Classroom)
22. Linked Open Data and the Semantic Web (Clearihue D130, Classroom)
23. Introduction to IIF: Sharing, Consuming, and Annotating the World’s Images (MacLaurin D114, Classroom)
24. Feminist Digital Humanities: Theoretical, Social, and Material Engagements (Cornett A229, Classroom)
25. The Frontend: Modern JavaScript & CSS Development (Clearihue A030, Lab)

Lunch break / Unconference Coordination Session (MacLaurin A144)
(Grab a sandwich and come on down!)

Undergraduate Meet-up, Brown-Bag (details via email)

Institute Panel: Perspectives on DH (or, #myDHis ...)
Chair: Alyssa Arbuckle (U Victoria)
(MacLaurin A144)

Milena Radzikowska (Mt Royal C): "Release the Kraken: Story-Driven Prototyping for the Digital Humanities."
Abstract: I have spent the last 15 years of my career designing text analysis tools for use by humanities scholars. In this brief presentation, I propose to share a concept-based approach to interface design for DH.

Emily Murphy (U Victoria): "#MyDHis Edgy."
Abstract: I will build upon—or, possibly, perform a misprision of—an essay by Polina Vinogradova; "#MyDHis messily, dusty, edgy, and radically inclusive!" Vinogradova evokes the mess and dust of the archives, the edges that connect nodes of a network, and the political impetus to think of cultural history and community together. I argue that these aspects of DH have a renewed importance as we think a moment of feminist historiography.

Margaret Konkol (Old Dominion U): "Prototyping Mina Loy’s Alphabet with a 3D Printer."
Abstract: This talk discusses the interpretive and methodological implications of using 3D printing technologies to prototype the archival diagrams of a proposed but never constructed plastic segmental alphabet letter kit—a game designed by modernist poet Mina Loy for F.A.O. Schwarz. Although intended as a toy for young children, "The Alphabet that Builds Itself," as a work of "object typography" articulates a theory of language as kinetic, geometric, recombinant, and open to mutation. Alphabetic segments extend into the x, y, and z coordinates in exponential iterations and conjoin with magnets. Combining elements of contemporaneous typefaces like Futura and Gill Sans, which represented modernity’s functional ideals and democratic principles of simplicity, these recombinant letters represent, as this talk argues, Loy’s unpublished modernist poem, an articulation of Loy’s concept of language as a physical fact in which substance, not just form, is semantic.

Lee Zickel (Case Western Reserve U): "Comfortably Trepid."
Abstract: #myDHis found outside the well-established, DH-friendly institutions, at an institution that is devoted predominantly to Medicine and Engineering. I, and with increasing frequency other DH practitioners and instructors, am not positioned in a DH Lab or Humanities Center, but in ITS. Part teacher, part technologist, part translator, I will briefly discuss my work supporting humanists and social scientists, particularly those who are new to or less comfortable with computational methodologies.

Dorothy Kim (Vassar C): "#MyDHis Antifascist."
Abstract: I’ve spent a lot of time in the last 12 months thinking about fascism, digital humanities, its long histories, and what it means to do DH work that centers social justice particularly in this global rise of late fascism. I will speak briefly about DH’s history, including the medieval history related to Busa but how that history really connects to data systems that created the Holocaust and also participated in the Cold War nuclear military complex.
Randa El Khatib (U Victoria): "Learning from the Iterative Process."
Abstract: #MyDHsIterative. In addition to the improvements that come with iterative projects, the iterative process itself is a fruitful area for scholarly inquiry. Within this iterative context, the various teams that I work with and I have been reflecting on and rethinking central DH practices, such as what it means to collaborate, prototype, remix, and implement DH values in our work. In this talk, I will present the various lessons learnt along the way.

Sarah Melton (Boston C): "#MyDHs...People."
Abstract: Taking seriously Miriam Posner’s exhortation to “commit to DH people, not DH projects,” I invite us to reflect on how people are the core of DH. In this brief talk, I will explore the intersections between DH, labor, and infrastructure.

Tuesday, 5 June 2018

5:00 to 6:00
Opening Reception (University Club)
We are grateful to Gale Cengage for its sponsorship.

9:00 to Noon
Classes in Session

12:15 to 1:15
Lunch break / Unconference
"Mystery" Lunches
▼ DHSI Lunchtime Workshop Session (click for workshop details and free registration for DHSI participants)
- 73. Introduction to ORCID (Digital Scholarship Commons, Classroom).

1:30 to 4:00
Classes in Session

▼ DHSI Colloquium Lightning Talk Session 1 (MacLaurin A144)
Chair: James O'Sullivan
- New Modes of DH and Archival Skills Acquisition in a Graduate Public History Course. Paulina Rousseau (Ryerson U)
- Walking a Transect: Exploring a Soundscape. John Barber (Washington State U)
- Centering the Edge Case: Designing Services for Humanities Data Research. Grace Afsari-Mamagani (New York U)
- Orwellian Vocabulary and the 21st-Century Politics. Ilgin Kizilgunesler (U Manitoba)
- Making Open Data from a Gray Archive. Sara Palmer (Emory U)

6:00 to 8:00
DHSI Newcomer's Beer-B-Q (Felicitas, Student Union Building)

Wednesday, 6 June 2018

9:00 to Noon
Classes in Session

Lunch break / Unconference
"Mystery" Lunches
▼ Brown Bag Lecture: Alexandra Branzan Albu (U Victoria): "Visual Recognition of Symbolic and Natural Patterns" (Digital Scholarship Commons, 3rd Floor McPherson Library)

Abstract: Image-based object recognition is a visual pattern recognition problem; one may characterize visual patterns as either symbolic or natural. Symbolic patterns evolved for human communication; they include but are not limited to text, forms, tables, graphics, engineering drawings etc. Symbolic patterns vary widely in terms of size, style, language, alphabet and fonts; however, literate humans can easily compensate for this variability and instantly recognize most symbolic patterns. On the other hand, natural patterns characterize images of physical structures; they often lack the intrinsic discriminability and structure of symbolic patterns, and vary widely in terms of pose, perspective, and lighting.

This lecture will explore similarities and differences in approaches designed for recognizing visual and symbolic patterns, and will address the following questions via examples.
- What are the distinctive characteristics of natural patterns? What dimensions of variability can we infer?
- What are the distinctive characteristics of symbolic patterns? What dimensions of variability can we infer?

Alexandra Branzan Albu is an Associate Professor with the Department of Electrical and Computer Engineering and cross-listed with Computer Science. Her research interests are related to image analysis, computer vision, and visual computing. She is actively pursuing outreach activities dedicated to increasing the women's presence in electrical engineering and computer science.

1:30 to 4:00
Classes in Session
Thursday, 7 June 2018

4:15 to 5:15

DHSI Colloquium Lightning Talk Session 2
Chair: James O’Sullivan

- Defining a Taxonomy of Abandonment for Online Digital Humanities Projects. Luis Meneses (Electronic Textual Cultures Lab, U Victoria) and Jonathan Martin (King’s College London)
- The Stories We Tell: Representing Gay and Lesbian History through Digital Technologies in the LGLC Project. Nadine Boulay (Simon Fraser University) and Ewan Matthews (Ryerson U)
- Italian Paleography in the Digital Domain. Isabella Magni (Newberry Library)
- Digital Humanities, A Question of Ethics. Negar Basiri (Louisiana State U)
- Writing Poetry in High School. Guadalupe Echegoyen (National Autonomous U Mexico)

Bring your DHSI nametag and enjoy your first tipple on us!

Thursday, 7 June 2018

9:00 to Noon
Classes in Session

12:15 to 1:15
UVIC Library/ETCL lunchtime talk: “A Humanities Application of 3D printing and Machine Translation in the ChessBard and Loss Sets” by Dr. Aaron Tucker
Digital Scholarship Commons, 3rd floor, Mearns Centre for Learning / McPherson Library
Bring your lunch and come on up!)

[Instructor lunch meeting]

1:30 to 4:00
Classes in Session

7:30 to 9:30
(Groovy?) Movie Night

Friday, 8 June 2018 [DHSI; DLFxDHSI Opening]

9:00 to Noon
DHSI Classes in Session

12:15 to 1:15
DHSI Lunch Reception / Course E-Exhibits (MacLaurin A100)

1:00 to 2:00
DLFxDHSI Registration (MacLaurin A100)

1:30 to 1:50
[DHSI] Remarks, A Week in Review (MacLaurin A144)

2:00 to 3:00
Joint Institute Lecture (DHSI and DLFxDHSI):
Bethany Nowviskie (CLIR DLF and U Virginia): “Reconstitute the World: Machine-reading Archives of Mass Extinction”
Chair: Lisa Goddard (U Victoria)
(MacLaurin A144)

Abstract: The basic constitution of our digital collections becomes vastly more important in the face of two understandings: first, that archives of modernity are archives of the sixth great mass extinction of life on our planet; and next, that we no longer steward cultural heritage for human readers alone. In the same way that we people are shaped by what we read, hear, and see, the machine readers that follow us into and perhaps beyond the Anthropocene have begun to learn from “unsupervised” encounters with our digital libraries. What will we preserve for the living generations and artificial intelligences that will come? What do we neglect, or even choose to extinguish? And from an elegiac archive, a library of endings, can we create forward-looking, speculative collections—collections from which to deep-dream new futures? The most extra/ordinary power we possess is the power to make poetry from records of the past. Could it be called on, one day, to reconstitute the world?
Saturday, 9 June 2018 [DLFxDHSI + DHSI Conference and Colloquium]

8:30 to 9:00  DLFxDHSI Registration (MacLaurin A100)

9:00 to 5:30  DLFxDHSI UnConference Sessions

9:00 to 4:00

* DHSI All Day Workshop Session  (click for workshop details and free registration for DHSI participants)

9:00 to 4:00

* 53. Building Your Academic Digital Identity  (MacLaurin D105, Classroom)

9:00 to 5:00  DHSI Colloquium Day Conference  (MacLaurin A144)

Welcome

People I: Documenting Online Lives. Chair: Molly Nebiolo (University of New York)

- Examining Gendered Harassment Online and in Silicon Valley. Andrea Flores (Utica College)
- This is Just to Say I Have <X> the <Y> in your <Z>: Modernist Memes in an Era of Public Apology. Shawna Ross (Texas A&M University)

Break

People II: Documenting Lives Online. Chair: Dheepa Sundaram (College of Wooster)

- Youtube Yoga and Ritual on Demand: The Virtual Economics of Hindu Soteriology. Dheepa Sundaram (College of Wooster)
- The Resemblage Project: Creativity and Digital Health Humanities in Canada. Andrea Charise (University of Toronto) and Stefan Krecsy (University of Toronto)

Lunch

Projects I: Building and Analyzing. Chair: Yannis Rammos (New York University)

- Building the ARTECHNE Database: New directions in Digital Art History. Marieke Hendriksen (Old Dominion University)
- The Ineffective Inquisition: The Holy Office’s Sphere of Influence in Early Modern New Spain. Kira Homo (Pennsylvania State University)

Break

Projects II: Mapping and Visualizing. Chair: Innocent Opara (Quinet Institute)

- Mapping Sarah Sophia Bank’s Numismatic Collection. Erica Hayes (North Carolina State University) and Kacie Wills (University of California, Riverside)
- Text Mining and Visualizing 18th Century American Correspondence. Ashley Sanders Garcia (University of California, Los Angeles)

Break

Practices: Digital Scholarship on Campus and in the Classroom. Chair: Alyssa Arhuckle (University of Victoria)
Concluding Remarks

Sunday, 10 June 2018 [SINM + DHSI Registration, Workshops]

8:30 to 9:00 Symposium on Indigenous New Media Registration (MacLaurin A100)

9:00 to 5:00 DHSI Registration (MacLaurin A100)

9:00 to 4:00

▼ SINM Sessions

- 63. Symposium on Indigenous New Media: Reading Group (Hickman 105, Classroom)
- 72. Symposium on Indigenous New Media: Indigitization (Hickman 120, Classroom)

Full details here

9:00 to 4:00

▼ DHSI All Day Workshop Sessions (click for workshop details and free registration for DHSI participants)

- 53. Building Your Academic Digital Identity (MacLaurin D105, Classroom)
- 54. An Introduction to the Archaeology of 1980s Computing (MacLaurin D114, Classroom)

9:00 to Noon

▼ DHSI AM Workshop Sessions (click for workshop details and free registration for DHSI participants)

- 55. Regular Expressions (MacLaurin D111, Classroom)
- 56. 3D Visualization for the Humanities (MacLaurin D010, Classroom)
- 58. DH Fieldwork Methods (MacLaurin D016, Classroom)
- 60. Pedagogy of the Digitally Oppressed: Inculcating De-/Anti-/Post-Colonial Digital Humanities (MacLaurin D107, Classroom)
- 61. Introduction to #GraphPoem. Digital Tools for Poetry Computational Analysis and Graph Theory Apps in Poetry (MacLaurin D101, Classroom)
- 62. Creating a CV for Digital Humanities Makers (MacLaurin D115, Classroom)

1:00 to 4:00

▼ DHSI PM Workshop Sessions (click for workshop details and free registration for DHSI participants)

- 64. Agent-Based Modelling in the Humanities (MacLaurin D111, Classroom)
- 65. Unleash Linux on MacOS (MacLaurin D010, Classroom)
- 66. DHSI Knits: History of Textiles and Technology (MacLaurin D016, Classroom)
- 67. Crowdsourcing as a Tool for Research and Public Engagement (MacLaurin D109, Classroom)
- 69. Web Annotation as Critical Humanities Practice (MacLaurin D103, Classroom)
- 70. Dynamic Ontologies for the Humanities (MacLaurin D107, Classroom)
- 71. Social Media Research in the Humanities (MacLaurin D101, Classroom)

4:10 to 5:00

▼ Joint Institute Lecture (DHSI and SINM):

David Gaertner (U British Columbia): "A Landless Territory?: CyberPowWow and the Politics of Indigenous New Media."
Chair: Deanna Reder (Simon Fraser U) (MacLaurin A144)

Abstract: Following the 1997 launch of Skawennati’s (Mohawk) CyberPowWow, digital space has become a vital new territory for the resurgence of Indigenous storytelling and cultural practice: "We have signed a new treaty," Cree artist Archer Pechawis wrote of this period, "and it is good. We have the right to hunt, fish, dance and make art at www.CyberPowWow.net, .org and .com for as long as the grass grows and the rivers flow." This talk will critically explore the theoretical, cultural, political-economic, and gendered dynamics underlying the histories and futures of Indigenous new media. Particular attention will be given in examining the ways in which new media and digital storytelling connect to and support key issues in the field of Indigenous studies, such as sovereignty, self-determination, decolonization, and land rights.

After the day, many will wander to Cadboro Bay and the pub at Smuggler's Cove OR the other direction to Shelbourne Plaza and Maude Hunter's Pub OR even into the city for a bite to eat.

Monday, 11 June 2018 [DHSI + SINM]
Your hosts for the week are Ray Siemens and Dan Sondheim.

7:45 to 8:15  DHSI Last-minute Registration (MacLaurin A100)

8:30 to 10:00  DHSI Welcome, Orientation, and Instructor Overview (MacLaurin A144)

9:00 to 4:00  SINM Sessions

▼  DHSI Classes in Session (click for details and locations)
- 29. [Foundations] Models for DH at Liberal Arts Colleges (& 4 yr Institutions) (MacLaurin D109, Classroom)
- 32. Stylometry with R: Computer-Assisted Analysis of Literary Texts (Clearihue A102, Lab)
- 33. Digital Storytelling (MacLaurin D111, Classroom)
- 34. Text Mapping as Modelling (Clearihue D131, Classroom)
- 35. Geographical Information Systems in the Digital Humanities (Clearihue A105, Lab)
- 36. Open Access and Open Social Scholarship (MacLaurin D110, Classroom)
- 37. Introduction to Machine Learning in the Digital Humanities (Cornett A229, Classroom)
- 38. Queer Digital Humanities: Intersections, Interrogations, Iterations (MacLaurin D119, Classroom)
- 41. Using Fedora Commons / Islandora (Human and Social Development A160, Lab)
- 42. Documenting Born Digital Creative and Scholarly Works for Access and Preservation (MacLaurin D115, Classroom)
- 43. Games for Digital Humanists (MacLaurin D016, Classroom & Human and Social Development A170, Lab)
- 44. XPath for Document Archeology and Project Management (Cornett A128, Classroom)
- 46. Surveillance and the Digital Humanities (MacLaurin D103, Classroom)
- 47. Text Analysis with Python and the Natural Language ToolKit (Clearihue A103, Lab)
- 48. Information Security for Digital Researchers (Clearihue D130, Classroom)
- 51. Critical Pedagogy and Digital Praxis in the Humanities (MacLaurin D105, Classroom)
- 52. Drupal for Digital Humanities Projects (MacLaurin D107, Classroom)

10:15 to Noon

Lunch break / Unconference Coordination Session (MacLaurin A144)

DHSI Undergraduate Meet-up, Brown-Bag (details via email)

DHSI Classes in Session

▼  Joint Institute Lecture (DHSI and SINM):
Jordan Abel (Simon Fraser U): "Indigeneity, Conceptualism, and the Borders of DH."
Chair: Michelle Brown (U Hawaii)
(MacLaurin A144)

Abstract: This talk brings together digital humanities discourses in computational textual analysis and Indigenous Literary Studies to analyze a corpus comprised of every book of Indigenous poetry published in Canada, extending from Pauline Johnson's 1895 book The White Wampum to Marilyn Dumont's 2015 book The Pemmican Eaters. While the main goal of this research project initially centered on the topic modeling of a corpus of Indigenous poetry, the project also addresses the systemic barriers that have prevented such work gaining traction, and likewise attempts to address the specific challenges that Indigenous writing (and in particular Indigenous poetry) present to current Digital Humanities methodologies.

5:00 to 6:00  Joint Reception: DHSI and SINM (University Club)

Tuesday, 12 June 2018

9:00 to Noon  Classes in Session

12:15 to 1:15  Lunch break / Unconference
"Mystery" Lunches
▼  DHSI Lunchtime Workshop Session (click for workshop details and free registration for DHSI participants)
- 73. Introduction to ORCID (Digital Scholarship Commons, Classroom).
### Wednesday, 13 June 2018

<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00 to 9:30</td>
<td>Classes in Session</td>
</tr>
<tr>
<td>9:30 to 10:00</td>
<td>Lunch break / Unconference</td>
</tr>
<tr>
<td>10:00 to 10:30</td>
<td>&quot;Mystery&quot; Lunches</td>
</tr>
<tr>
<td>10:30 to 12:00</td>
<td>Classes in Session</td>
</tr>
<tr>
<td>12:00 to 1:00</td>
<td>Lunch Reception / Course E-Exhibits</td>
</tr>
<tr>
<td>1:00 to 2:00</td>
<td>DHSI Colloquium Lightning Talk Session 4 (MacLaurin A144)</td>
</tr>
<tr>
<td>2:00 to 3:00</td>
<td>Faraway, so close: Has the political environment really changed in Ecuador? (Luis Meneses, Electronic Textual Cultures Lab, U Victoria)</td>
</tr>
<tr>
<td>3:00 to 4:00</td>
<td>Re-mixing Melville's Reading: Text Analysis of Marginalia with R and XSLT (Christopher Ohge, U London, School of Advanced Study) and Steven Olsen-Smith (Boise State U)</td>
</tr>
<tr>
<td>4:00 to 5:00</td>
<td>Developing Interactive and Open-Source OER: Inquiry-Based Music Theory (Evan Williamson, U Idaho)</td>
</tr>
<tr>
<td>5:00 to 6:00</td>
<td>Spatial Humanities and the Web of Everywhere (Ken Cooper, SUNY Geneseo)</td>
</tr>
<tr>
<td>6:00 to 7:00</td>
<td>&quot;Half Way There (yet again)!&quot; [An Informal, Self-Organized Birds of a Feather Get-Together] (Felicitas, Student Union Building)</td>
</tr>
</tbody>
</table>

### Thursday, 14 June 2018

<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00 to 9:30</td>
<td>Classes in Session</td>
</tr>
<tr>
<td>9:30 to 10:00</td>
<td>Lunch break / Unconference</td>
</tr>
<tr>
<td>10:00 to 10:30</td>
<td>&quot;Mystery&quot; Lunches</td>
</tr>
<tr>
<td>10:30 to 12:00</td>
<td>[Instructor lunch meeting]</td>
</tr>
<tr>
<td>12:00 to 1:00</td>
<td>Classes in Session</td>
</tr>
<tr>
<td>1:00 to 2:00</td>
<td>DHSI Colloquium Lightning Talk Session 5 (MacLaurin A144)</td>
</tr>
<tr>
<td>2:00 to 3:00</td>
<td>Composition not Inheritance: Imagining a Functional Digital Humanities (Andrew Pilsch, Texas A&amp;M U)</td>
</tr>
<tr>
<td>3:00 to 4:00</td>
<td>Plotting Our Trajectories: Navigating, Situating, and Re-Inventing Research Topoi with R (Sean McCullough, Texas Christian University) and Jongkeyong Kim (Texas Christian U)</td>
</tr>
<tr>
<td>4:00 to 5:00</td>
<td>Herb Simon and His Books. Avery Wiscomb (Carnegie Mellon U) and Daniel Evans (Carnegie Mellon U)</td>
</tr>
<tr>
<td>5:00 to 6:00</td>
<td>(De/Re)Defining &quot;The Digital&quot;: A Decolonial Approach to Digital Humanities (Ashley Caranto Morford, U Toronto) and Arun Jacob (McMaster U)</td>
</tr>
<tr>
<td>7:00 to 8:30</td>
<td>(Groovier?) Movie(r) Night (MacLaurin A144)</td>
</tr>
</tbody>
</table>

### Friday, 15 June 2018

<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00 to 9:30</td>
<td>Classes in Session</td>
</tr>
<tr>
<td>9:30 to 10:00</td>
<td>Lunch Reception / Course E-Exhibits (MacLaurin A100)</td>
</tr>
</tbody>
</table>
1:30 to 2:30

Institute Lecture: William Bowen (U Toronto Scarborough): "Discovery, Collaboration and Dissemination: Lessons Learned and Plans for the Future" (MacLaurin A144)

Abstract: Much has changed and continues to change in digital humanities since the formal establishment of Iter in the Fall of 1997. However, the mandate of the not-for-profit partnership to support "the advancement of learning in the study and teaching of Middle Ages and Renaissance (400–1700) through the development and distribution of online resources" continues to have relevance. This presentation explores the striking challenges faced by Iter and presents our current thinking on the realization of this mandate for the future through a platform with a focus on facilitating the discovery of the academic resources necessary to our work; creating an environment for collaboration, sharing and developing projects; and on enabling the distribution and publication of our scholarship.

2:40 to 3:00

Awards and Bursaries Recognition
Closing, DHSI in Review (MacLaurin A144)

Contact info: institut@uvic.ca P: 250-472-5401 F: 250-472-5681
DIGITISATION FUNDAMENTALS AND THEIR APPLICATION
Welcome!

You’re probably reading this coursepak because you’re registered in the "Digitisation Fundamentals and their Application" course at UVIC, part of the 2018 Digital Humanities Summer Institute.

To make the most of the practical, hands-on component of our time together at the DHSI, please bring with you any of your own projects which would benefit from exploration with the tools and techniques we'll be discussing. Of course, we will have activities for you if you have nothing prepared in this regard, but we, and likely you, would prefer that you could quickly apply digitisation skills to your own work!

Just so we’re all as adequately prepared as possible, here is a list of some things you might like to bring along to class. You certainly don't have to bring any or all of these things, but they'll come in handy:

- Headphones (earbuds will be fine)
- A USB microphone and/or audio recorder (if you have one)
- A USB memory stick, to store any work you create
- Any digital camera you have - this could be a phone, or a still or video camera
- Your own laptop, if you like, as much of the work we’ll do requires only free software tools.

At the beginning of this coursepak you will find a schedule. Each day has a list of topics we’ll discuss, and URLs for suggested “reading” material which you'll ideally have a chance to explore before class.

The remainder of the coursepak contains supplemental materials for each topic, and some tutorials to attempt, particularly if you are unfamiliar with the software tools!

Looking forward to seeing you soon!

Robin ([robin.davies@viu.ca](mailto:robin.davies@viu.ca)) and Calleigh dhsi.org
Course Schedule

Day One - Intro to Digitisation

Reading
- [http://digitalhumanities.org/companion](http://digitalhumanities.org/companion) - Conversion of Primary Sources
- [http://www.dlib.org/dlib/march09/holley/03holley.html](http://www.dlib.org/dlib/march09/holley/03holley.html) - OCR Challenges

AM
- Introductions
- Group/Individual Goals
- Course Overview
- Project Management
- Packaging/Publishing Options

PM
- Introduction to Digitisation
- Images
- Scanning/Photography
- OCR
- Meta Data

Day Two - Web

Reading
- [https://resilientwebdesign.com/chapter2/](https://resilientwebdesign.com/chapter2/) - A liberal attitude towards errors

AM
- World Wide Web and HTML

PM
- CSS Crash Course
- Responsive Web Design
Day Three - Audio

Reading

- [https://www.xiph.org/video/vid2.shtml](https://www.xiph.org/video/vid2.shtml) - Viewing - What is Digital Audio?

AM + PM

- Analog to Digital Conversion
- Audio Production
- Non-linear, Non-destructive Editing
- Audio on the Web

Day Four - Video

Reading


AM + PM

- Video Production
- What does it mean to use the moving image as your packaging mechanism?
- Video Codecs
- Video on the Web

Day Five - Code and Wrap-up

Reading

- [http://vimeo.com/48080736](http://vimeo.com/48080736) - Code Literacy (more listening than viewing!)

AM

- Teaching and Learning Code
- Group Project
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transferring Analog Audio Sources to the Computer</td>
<td>44</td>
</tr>
<tr>
<td>Audio Standards for Recording Audio Files</td>
<td>45</td>
</tr>
<tr>
<td>Archival Standards for Digitising Audio Files</td>
<td>45</td>
</tr>
<tr>
<td>Optimizing Your Computer for Recording Digital Audio</td>
<td>46</td>
</tr>
<tr>
<td>Resources for Digital Audio</td>
<td>47</td>
</tr>
<tr>
<td>Software Packages</td>
<td>47</td>
</tr>
<tr>
<td>Audio Standards Websites</td>
<td>48</td>
</tr>
<tr>
<td>Audio Tutorial 1 - Getting Started</td>
<td>49</td>
</tr>
<tr>
<td>Input</td>
<td>49</td>
</tr>
<tr>
<td>Recording Audio</td>
<td>49</td>
</tr>
<tr>
<td>&quot;Ripping&quot; Audio from CDs</td>
<td>52</td>
</tr>
<tr>
<td>Opening a File</td>
<td>53</td>
</tr>
<tr>
<td>Saving a Project</td>
<td>53</td>
</tr>
<tr>
<td>Exporting a File</td>
<td>54</td>
</tr>
<tr>
<td>Process</td>
<td>55</td>
</tr>
<tr>
<td>Selecting and the Editing Tool</td>
<td>55</td>
</tr>
<tr>
<td>Copying and Pasting</td>
<td>56</td>
</tr>
<tr>
<td>Cutting</td>
<td>57</td>
</tr>
<tr>
<td>Deleting</td>
<td>57</td>
</tr>
<tr>
<td>Undo, Redo, and Versioning</td>
<td>58</td>
</tr>
<tr>
<td>Output</td>
<td>58</td>
</tr>
<tr>
<td>Burning CDs or DVDs</td>
<td>58</td>
</tr>
<tr>
<td>Publishing an MP3</td>
<td>59</td>
</tr>
<tr>
<td>Audio Tutorial 2 - Advanced Audio Editing</td>
<td>60</td>
</tr>
<tr>
<td>Pasting</td>
<td>60</td>
</tr>
<tr>
<td>Mixing</td>
<td>62</td>
</tr>
<tr>
<td>Silencing Unwanted Sounds</td>
<td>63</td>
</tr>
<tr>
<td>SECTION 4: DIGITISING VIDEO</td>
<td>66</td>
</tr>
<tr>
<td>BEST PRACTICES FOR DIGITAL VIDEO</td>
<td>67</td>
</tr>
<tr>
<td>Digital Video</td>
<td>67</td>
</tr>
<tr>
<td>Shooting Video</td>
<td>67</td>
</tr>
<tr>
<td>Resources for Digital Video</td>
<td>70</td>
</tr>
<tr>
<td>(Some) Codecs Described</td>
<td>71</td>
</tr>
<tr>
<td>SECTION 5: Web Production</td>
<td>72</td>
</tr>
<tr>
<td>Introduction</td>
<td>73</td>
</tr>
<tr>
<td>The World Wide Web and Digitization</td>
<td>73</td>
</tr>
<tr>
<td>The Basics of HTML</td>
<td>74</td>
</tr>
<tr>
<td>The Building Blocks</td>
<td>74</td>
</tr>
<tr>
<td>Getting Started</td>
<td>75</td>
</tr>
<tr>
<td>Starting Template</td>
<td>76</td>
</tr>
<tr>
<td>Tag, You’re It</td>
<td>77</td>
</tr>
<tr>
<td>Angle Brackets</td>
<td>77</td>
</tr>
<tr>
<td>Nesting Tags</td>
<td>79</td>
</tr>
<tr>
<td>So Many Tags, So Little Time</td>
<td>80</td>
</tr>
</tbody>
</table>
Common Tags ................................................................. 81
The Framework of a Web Page .......................................... 82
Semantic HTML................................................................. 82
Validating Your HTML ..................................................... 83
Coding Exercise................................................................. 84
Cascading Style Sheets .................................................... 85
The Joy of Web Design ..................................................... 85
A Lesson In Targeting ...................................................... 86
Advanced Targeting .......................................................... 87
The Box Model ................................................................. 88
Adding CSS to your Website .............................................. 89
CSS Inspiration and Resources .......................................... 89
Building Websites ........................................................... 90
Web Development: Theory and Practice ............................. 91
Paper Prototyping ............................................................ 91
Using A Template ............................................................ 92
SECTION 6: WEB 2.0......................................................... 94
Rich User Experience ........................................................ 97
User Participation ............................................................. 98
Dynamic Content ............................................................. 98
Metadata ........................................................................... 98
Web Standards ................................................................. 99
Openness & Freedom ......................................................... 99
Collective Intelligence ........................................................ 100
Examples ........................................................................... 100
Summary ........................................................................... 104
SECTION 7: Putting it all Together ...................................... 105
Introduction ........................................................................ 105
Creating a Digital Edition ................................................ 107
Distributing Oral Histories ................................................ 112
Coordinating Access to Music Notation ............................... 119
Conclusion .......................................................................... 124
Works Cited ........................................................................ 125
SECTION 1: PROJECT PLANNING

Prepared by Michael Nixon, Robin Davies, Gerry Watson, Karin Armstrong
Planning for Digitisation Projects

A plan is essential for any digitisation project. A project plan will help you anticipate problems and needs, track your progress through the material, guide future development, and coordinate resources and staff. Your plan can be as flexible or rigid, detailed or vague as you like, but no matter what form it takes, in the long run making a plan will save you time, headaches, and resources.

Project planning can take many forms and encompass varying ranges of complexity. For our purposes, we'll keep it simple. Even if you are an experienced project manager, there are a few things to keep in mind when planning for any digitisation project.

**Making the plan**

There are a number of software packages that can help you build your project plan, track resources, and offer a host of other tools and solutions for large project management. While these software tools can be useful, their effectiveness extends only so far as the vision and detail you supply. First, establish broad goals. Aim for an idea of what you want the final product to look like; then you can begin filling in the details. Once you have addressed some of the concerns below, you will be able to consciously plan your project. Remember, you don't have to focus on every single detail the first time through. It is sometimes best to think through your project at different scales, using an iterative approach.

For example, begin by thinking of major goals and the final product, and then break the project into major sections or phases; each phase can then be broken down into smaller goals, and those goals into individual tasks.

This is also a good point to get your team involved in the growth of the plan, as they will have insights into how the work will or should progress. This approach allows your team to take ownership of their specific tasks, and take part in setting deadlines. Extending the decision-making process to your team takes the pressure
off of you to micro-manage the project, and allows them some control over, and responsibility for, their own work.

The plan can take many forms and serve multiple functions. The important thing to remember is to make a plan, revise it as you need to, and try to stick with it. This simple process will help organize your thoughts and give you a clear framework for progress in meeting your project’s unique needs.

**What will the final project look like?**

First, decide what you want to accomplish in general terms. Do you want an online exhibition or portfolio of art, literature, cartography, or historical materials? Is the end product of your digitisation going to be some form of print? Do you want to build a research tool as a software package or an online archive for public use? Is the project for private or institutional use only? All these questions guide what the product should be, and lead to vastly different requirements in terms of digitisation, resources, equipment, staff, storage, and a host of other minutiae that should at least be acknowledged before you begin. Even with a vision and a plan, things will inevitably go wrong, but at least you can be prepared for contingencies, which will enable you to react more appropriately and quickly.
Assessing your Digitisation Project

A vital part of any digitisation project is project assessment. Some initial decisions will streamline the process, help you decide how to proceed, and most effectively dictate the use of available resources. Here are some questions to consider:

What is your presentation medium?

Will your final project include audio, images, text, video, HTML, interactive 3D models, or some combination of these? Making distinctions between applicable media will help to define and focus your learning task. With each medium, one must gain relevant vocabulary and knowledge, and learn applicable skills.

For example, if your project involves scanning images for online use, it is important to know that you may be limited in your choice of file types (JPG, GIF, and PNG). You’ll also need to understand pixels, compression, and bandwidth.

This knowledge will be integral in the day-to-day work of your project, and will also prove useful in the initial stages of your project as you make staffing and equipment decisions. The equipment you choose will dictate the standards you set. Whether you use equipment or facilities already in place at your institution, or secure funding to purchase dedicated machines, you should educate yourself in the limits and benefits each option represents. When purchasing new equipment, do not rely on what the salesperson says. Research the scanner, computer system, camera, and printer that you will come to depend on. Only you truly understand what you want out of these machines. If the funding is available, it is best to purchase a machine that does more than what you need. At the outset of your project, it can be tough to foresee all your needs, so choose a piece of equipment that does what you need, but is also capable of more advanced functions should they become necessary.

If you know you want to scan in archival images at 400 dpi, it might be a good idea to buy a scanner that can go far beyond this and that can also scan slides and microfilm, in case you need access to those media. Similarly, don’t buy a printer that produces a minimum quality document when someday you may need it to produce
high quality advertisements for your project. Make purchases based on immediate needs, but with an eye to solving future problems as well. Though not marked on the box, hardware has a shelf life and expiry date as does any reactant.

**What is your long-term plan for this project?**

Decide what your final output will be: web display, an archive, digital projection, storage at home, long-term pedagogy, and so on. What are the procedural implications of this projected outcome? Knowing how you will access and present your final data will go a long way in deciding your digitisation process.

For example, if you are planning to archive your data for long-term storage, be certain to choose the best storage format possible, one which will ensure long-term viability and versatility. Generally, this means large, *lossless* file formats like TIFF (for images) and AIFF (for audio). Or, for a project destined for web or classroom use, choose a smaller, portable, perhaps *lossy* format such as JPG (for images) and MP3 (for audio).

**How will you coordinate and track your decisions?**

Project management is critical to the success of larger initiatives. It is recommend, for example, to maintain an up-to-date project summary document which outlines the chronology of decisions made and work completed helping to keep the project on-task and moving forward. Other valuable management practices include establishing consistent naming conventions for documents - title, location, date captured, index - building a well-organized directory structure, and keeping track of metadata.

**How will you store or present your data?**

Storage: burning data to CD or DVD is still an inexpensive and convenient choice, providing a spacious, transportable and safe storage solution. A USB flash storage device is also a possibility. Safer still would be a portable or internal hard disk, or a third party cloud server. Ideally, you should try to establish several locations for
data storage, with an established and consistent backup process. Overall, your best bet is to diversify. With larger archival projects, long-term planning will involve establishing a safe, permanent location for your master data.

Presentation: most commonly, a project will be displayed on the web or in a classroom. This often requires versatile file formats and reduced storage space. In this case, aim for clarity and convenience rather than posterity.

**The Eight Basic Principles of Project Management**

When writing your project plan and while your project is underway, remember these simple principles of project management:

No major project is ever installed on time, within budget, with the same staff that started it. Yours will likely not be the first.

Projects progress rapidly until they become 90 percent complete; they then remain 90 percent complete forever.

One advantage of fuzzy project objectives is that they let you avoid the embarrassment of estimating the corresponding costs.

When things are going well, something will go wrong. When things just can't get any worse, they will. When things appear to be going better, you have overlooked something.

If project content is allowed to change freely, the rate of change will exceed the rate of progress.

No system is ever completely debugged; attempts to debug a new system inevitably introduce new bugs that are even harder to find.

A carelessly planned project will take three times longer to complete than you expected; a carefully planned project will only take twice as long.

Project teams detest project reports, because these reports vividly manifest their lack of progress.

(http://www.daclarke.org/Humour/projects.html)
SECTION 2: DIGITISING IMAGES

Prepared by Robin Davies, Cara Leitch, Gerry Watson, Karin Armstrong, Anne Correia, Trish Irish, and Graham Lyons
**Best Practices for Digitising Images**

When beginning a digital imaging project, the first goal is to determine how your images will be used. Images designed for the web, for print, or for archival purposes are all very different. Below are a few basic guidelines for any project, and some specific pointers for particular kinds of projects. Some basic concepts are discussed only briefly here; please refer to the online links for more information.

**Basic Rules for Digitising Images**

**Rule 1:** Always save a master image in a *lossless* format.

This means RAW, TIFF, or PNG should be set as your default image format when capturing or scanning images. While your final project may require JPGs, every time you alter and save a JPG, information is lost and quality decreases. Your image quality will be much better if you edit your master image as a TIFF, and convert to JPG only in the final stages of your project.

**Rule 2:** Start with a high quality image, and then decrease the resolution (and perhaps increase the compression) to suit your project.

You cannot take a small JPG and turn it into a high-quality print. But you can take a high-quality printable image and turn it into a small JPG. It’s far easier to scan at a higher resolution than you think you need, and then manipulate the image for your project. Target a higher level of image quality for your scans than you may need for your final project. You might even notice some finer details!

**Rule 3: Manage your images.**

It’s hard to remember the difference between imgJune.jpg and picture_5.png. Make the effort to name your images meaningfully from the beginning. You may also want to add an index number for easy reference in large projects. Take the time to consider how to organize images for your project. A great image is useless if you can’t find it.
**Common Terms for Imaging**

**CMYK**

Like RGB (see below), CMYK, which is an acronym for Cyan, Magenta, Yellow, and Key (black), represents the ratio and relative intensity of colours in an image, specifically for print display. Most colour printers use this scheme. The use of Black here means the more expensive inks can be used sparingly to obtain dark colours.

**RGB**

Red-Green-Blue colour attempts to represent - for a computer monitor or data projector - a colour in numerical form. By defining a level of intensity (regardless of how that intensity is measured) for red, green, and blue, one can create a full colour palette for digital display.

**HSB**

The Hue, Saturation, and Brightness model is perhaps the most human way to describe colour. Hue refers to the value of the colour, as might be chosen from a colour wheel or rainbow. Saturation is the intensity of the colour; colours with low saturation will appear gray, faded. Brightness measures the amount of "light" shining on the colour; very low brightness settings make all colours look black.

**Grayscale**

Grayscale is a mode of displaying images with no colour, in shades of gray. Removing the colour information from an image can make for useful photographic effects, and can decrease the file size of the image. We often erroneously call grayscale images black and white images. In addition, when someone asks for a black and white image, they often are really looking for a grayscale image tinted with a sepia colour.
Pixel

The term pixel is a computer abbreviation for picture element, the smallest element of a digital image or digital display.

dpi

Dots per inch is a term often used to quantify the (maximum) resolution capabilities of printers, scanners and monitors. The more dots per inch, the higher the resolution: 600 dpi would mean 600 x 600 = 360000 dots per square inch. It is important to understand this term when converting from analog to digital, or from digital to analog. Within the digital system (monitors, web display, digital projection), the term "inches" is irrelevant, but we must keep in mind how the "dots" on the printed page will translate to "pixels" on the video screen. In general, remember that a higher dpi means a larger image on the video screen or in the web browser, and a larger the file size for storage.

ppi

Pixels per inch. This term is directly related to dpi - in fact, the two terms are often used interchangeably. Pixels per inch refers to the relationship between video display (digital) and print display (analog). While dpi indicates the number of ink dots for every inch of physical space on the printed page, ppi denotes the equivalent level of detail on a video screen, the number of pixels for each inch of display. However, it is important to remember that the physical display size of an image on a video screen is partly dependent on the resolution of the screen. A 400 pixel-wide picture displays across half of a retro 800x600 display, but only about a quarter of an HD television. The higher the ppi, the larger the image will appear on a monitor. For example, if you were to scan a 4x6 photo with different dpi settings, your image size would display differently on the same monitor; a 4x6 photo scanned at 600 ppi/dpi would display a digital image twice as large as the same photo scanned at 300 ppi/dpi.
**Pixelation**

When you enlarge a small image, you may, depending on the resolution, encounter pixelation. This term describes the effect of individual (usually square) pixels being visible to the eye. This problem often occurs when digital images are archived incorrectly, stored with a *lossy* file type like JPG or GIF, or when an image is frequently resampled or colour-altered. You’ll likely run into this problem if you try to appropriate a web image for use in a print situation.

**Scaling or Resampling**

Resampling refers to changing the size of an image for digital display purposes (monitor, digital projector, or the web). Resampling alters an image significantly, engaging various compression algorithms to fit an image to certain dimensions, such as 800x600. Because each pixel is changed in order to fit new parameters, you might encounter a loss of quality, often because the resampling process adds or removes data from the image through interpolation.

**Resolution**

The term resolution is applied to digital images, print objects, and monitor displays; it reflects, in general, the “granularity” of an image. Resolution is expressed in dpi for print media, in ppi for visual display, and in pixel dimensions for monitor resolution: 1280x720.

**Common Image File Types**

The list below includes only some of the many image file formats available.

**BMP**

Bitmap. This is used for bitmap graphics, usually on the Windows OS. BMP is a lossless file format, so files can be very large.
**GIF**

Graphics Interchange Format. This is a very common format on the web, usually for logos or illustrations with few colours. Because of the way this format handles colour variation, photographic colour transitions are often jagged. The GIF format provides for multiple frames, hence the "animated GIF".

**JPG**

Joint Photographic Experts Group is the standard file format for compressed photographic files. When you create a JPG or convert an image from another format to JPG format, you are asked to specify the quality of the image you want. Since the highest quality results in the largest file, you can make a trade-off between image quality and file size. However, the JPG format is *lossy*, meaning that every time you save a file, some information is lost through compression. JPEG loses its E when seen as a file extension - lossy compression!

**PNG**

Portable Network Graphics. Since the PNG compression is fully lossless, and since it supports up to 48-bit colour or 16-bit grayscale, restoring and resaving an image will not degrade its quality, unlike standard JPG.

**PSD**

The Photoshop proprietary format, containing information about layers and other details specific to that software. This format is high quality, and useful if you want to be able to change part of a complex graphic, but the file size is large and not supported on the web, or in many imaging programs beside Photoshop. There are some free software tools which can understand some elements of the PSD files.

**RAW**

A completely uncompressed capture produced by a digital camera with all filters and compression tools disengaged. For professional photographers, this file format maintains the highest fidelity to the photographic subject, and retains a high image quality.
quality after manipulation in Photoshop or other software. Drawbacks of this format include substantial file sizes and a lack of compatibility across platforms and media.

**TIFF**

Initially an acronym for Tagged Image File Format, the TIFF is the most widely supported lossless image format. This is the format to use for any high quality master image, and is especially relevant for archiving. TIFF files can be quite large.

**Vector Images**

All the above bitmap image formats store information about every single pixel within an image. This is excellent, in particular, for photorealistic images where resolution and high-definition are essential. However, many animation and publishing software tools use vector-based formats. In a vector image, information is stored which will allow the user’s system to actually *redraw* the image. To store an image of a square, for instance, a vector system would write a "recipe" for recreating the image, making note of the location, width, height, and colour of the square, instead of storing the blue data individually for each pixel within the square. Software tools such as Adobe’s Animate and Illustrator create vector images.

**Imaging for the Web**

Putting digital images on the web is often the main goal of a project. However, wherever possible, you should always treat web images as the final product of your project, not as the initial starting point (see Rule 1 above).

In order to be easily viewed on the web, many images are served in the JPG format, which is lossy and compressed. Your goal is to balance image quality with image size. Quality depends on your audience’s systems, and the purpose of the image. A logo should be very small, while a manuscript page must be quite large.

When you are preparing images for the web, disregard all measurements that come from print. Don’t look at the scale in inches or even in dpi. Use pixels instead, since
this is how a web browser measures an image. Typically, scale your image to a maximum of 1280x720 pixels, the size of an average computer screen.

For web imaging, here are three software tools to consider: Photoshop, Affinity Photo, and Graphic Converter. Photoshop is the industry standard; it is excellent for fine detail manipulation and image creation. Affinity is similar to Photoshop, though avoids a subscription licensing system. Graphic Converter is a Mac-only and less expensive program, and is very useful for preparing large folders of images using batch conversion.

If you have copyright concerns about your images, you may wish to use Adobe Acrobat’s PDF format, which can make it less easy for a user to save or modify an image. In addition, the PDF format allows a user to zoom in on detail, and offers a better web-to-print experience than other compressed formats.

**Imaging for Print**

In order to print a digital image, you need to ensure that you have a high-resolution image. There are two factors to consider: the physical print size and the resolution. A printer has a maximum dpi, meaning how many dots per inch it can print. If your project is using a single printer, it is worth finding out its maximum dpi and targeting that quality. However, if you are preparing images for general printing, the standards below offer a good compromise:

- Enlarge your image to a suitable print size. For snapshots, a 4x6 inch image is sufficient. Some projects may require a full-page size of 8x10 inches.

- Set your resolution to 300 dpi. This will give a good quality print on almost any printer. Some older printers have a lower maximum dpi, and will not be able to take advantage of your higher resolution.

- Choose a format for your image. This choice will depend largely upon your audience. If your project is a CD, a TIFF or PNG image is perfect.
This will allow your audience to print multiple images on a page, and arrange or edit the images. If you are serving a printable image on the web, a PDF or high-quality JPG offers good compression and quality. Keep in mind that many cameras produce JPG images, so upgrading to a lossless format for printing won't add any quality.

**Imaging for Archival Purposes**

To ensure the longevity of your archival project, the following needs must be addressed:

**Metadata**

For an image to remain useful, it must have information attached to it, such as the place, the date, the subject, and the original medium. Ideally, this information should be searchable and easily accessible. See the online links for more information.

**Storage Medium and File Formats**

Every year, new media are developed and new computer hardware is sold. Eventually, older media can be hard to access. The best way of ensuring that your images remain accessible is to use an open standard format instead of a proprietary format. In addition, a review process should be in place to ensure that all images are migrated to new media and/or formats when needed. Save image files in an uncompressed format like TIFF if possible. In addition, and in general for digitisation projects, don’t be too quick to throw away technology perceived to be obsolete. For example, there are many audio recordings found only on vinyl, and turntables are required to appreciate these recordings. The same can be said for images stored as slides.
Quality

Archival images must be of a very high quality; the purpose is to ensure that these images are available and can be easily adapted to many contexts without having to return each time to the physical format (microfilm, slides, paper, etc.). The highest quality can be obtained by scanning at the highest possible resolution.
Resources for Digital Imaging

General Info

Moving Theory into Practice: Digital Imaging Tutorial

http://preservationtutorial.library.cornell.edu/tutorial/contents.html

Capture Your Collections: Planning and Implementing Digitization Projects


These tutorials will guide you through the process of digitisation; the site includes information about best practices in digitisation project management.

Scantips.com

http://www.scantips.com

Developed by Wayne Fulton as an on-line support tool for his independently-published handbook on scanning, this site provides insightful beginner-level information on scanning and the basics of digital imaging. Check out, in particular, his meditation on image resolution.

A History of Colour Models


Printing Resolution Calculator

http://www.scantips.com/calc.html

An on-line tool developed by Wayne Fulton that helps you to decide on scanning resolution, bridging the divide between digital and analog display, and taking into account issues such as cropping and margins in its calculations.
Metadata Resources

DCMI
http://www.dublincore.org/

Dublin Core Metadata Initiative. An open forum to develop the Dublin core metadata standards, which are used by the Canadian Government.

MDC
http://www.loc.gov/standards/mdc/elements/

Library of Congress Metadata resources.
Imaging Tutorial 1: Flatbed Scanning

1. Scanners often come with custom software to facilitate the scanning of images into the computer. Sometimes these software tools are separate applications in the computer, but sometimes they exist as plug-ins in standard image editing tools such as Photoshop.

2. Regardless, the software which communicates with the scanner will likely provide you with control over resolution, scanning area, colour or brightness settings, and the type of media you're scanning (photos, drawings, slides). Often, you'll need to access the software's advanced options to control many of these settings. The scanning software might look like this:

![Advanced Properties dialog box]

3. Perhaps the most important variable to control is the resolution, which could be set at 400 dpi (or more), as shown above, for high-quality work. The software shown above provides control over brightness and contrast, but it is wisest to capture the image as is, and leave adjustments for later.
4. The last step before you scan your image is often a "preview" mode. This process may take a few minutes if the scanner needs to warm up, but when it is finished you will be able to see a preview of the content currently on the scanner, and the area to be scanned enclosed in the dashed lines. You can resize the scanning area by clicking on the red boxes in each corner. Some tools will provide the capability to select numerous areas of the same content, and save those areas as separate images.

5. Depending on the settings, the scan may take quite some time. Scanning at a very high resolution forces the scanner to look very carefully at every part of the content. When the scanner is finished, the image will appear in the editing software, such as Photoshop, or it may be saved to disc in which case you'll open it manually. The first thing you should do before editing is to save a high-quality copy of the image, using a format such as TIFF, and a meaningful name.
6. You now have a master copy of your image that meets archival standards. This copy should always be left alone and unmodified. Whenever you want to make a new image from the master, the first step is to save it in a new location or with a new file name to avoid over-writing the master image file.
Imaging Tutorial 2: Digital Photography

Sometimes the nature of the material to be digitised is not conducive to flatbed scanning. Consider oversize materials like maps, which if sent out to be digitised professionally can be costly. You may also have old manuscripts or fragile archival materials which require gentler treatment. In such cases, it may be useful to try digital photography techniques, which can be effective in the right environment. The right combination of camera, settings, and lighting can yield surprisingly good results. Some things to remember when using this method:

✓ Try to control lighting conditions as much as possible.

✓ If your document is fragile, use archival techniques in handling and storage.

✓ Be patient. Digitising a couple hundred pages of a book in this manner can take hours when done properly.

✓ Experiment with different combinations of cameras, lighting, resolution, and file formats until you find one that works.

✓ The settings to take a good image of one document may not be perfect for a different document. Make sure you do a trial run for each new document you want to digitise.

✓ Depending on the material you are photographing, you may be able to see text or images from the other side of the page. Place a piece of black poster board or construction paper underneath your page to minimize how much of the “phantom” text or image appears in your photo. If your page curls or is otherwise difficult to lay flat, lay a sheet of clear plastic or glass overtop.
The following steps will guide you through the process of photographing content:

1. Mount or position the camera at an appropriate distance from the document. A stable camera will produce far better images in low lighting situations. Be sure you can see everything you need to in the camera view screen.

2. Set the camera to an appropriate resolution. Many cameras have the ability to take pictures with different quality settings. Be sure the camera is will be using a high enough amount of pixels to represent your image.

3. If the material you are photographing is fragile or sensitive to light, turn the flash off and rely on available light. Bring in lamps as required.

4. If necessary, manipulate the zoom lens until you have targeted the document properly. It’s often best to move the camera instead of zooming, particularly if you’re working with a handheld (not fixed or mounted) camera.

5. Take the picture.

Once you have an image on the camera, you will need to transfer it to your computer. The simplest way to do this is connecting your camera with a USB cable.

To transfer images from the digital camera via a USB cable:

6. Plug the USB cable into the computer and also into the appropriate port on the camera.

7. Turn the camera on. You may need to switch the camera to viewing mode as opposed to shooting mode.

8. Software on the computer may open automatically and detect items to be imported. If not, start the appropriate software. Some cameras come with software which needs to be installed on the computer before you connect the camera. Many software tools, such as Apple’s iPhoto, understand how to speak to numerous camera models.
9. When importing pictures, give your "roll" or folder of imported images a name and a description. This will help you to keep track of large amounts of content. Most photo software will organize your imported images by date.

10. Decide whether you want to delete the originals from the camera. If you have taken a lot of photos, you may want to erase them to free up memory for your next batch of photography.

Many cameras make use of removable media for storage, such as an SD card; some computers have ports for these or other specific media. Some cameras and computers allow data to be transferred wirelessly, and/or uploaded directly to online hosting sites such as YouTube or Flickr, simplifying this step of the digitisation process.

You can perform some basic photo editing in software such as iPhoto, or the software which comes with your camera, (cropping, red-eye reduction,) but it is sometimes necessary to export your photos and edit them in a fully-featured program like Photoshop.

When exporting photos:

11. Select the photo (or photos) you wish to export from the capture software.

12. Remember to select a high-quality format such as TIFF, give your file a meaningful name, and leave your image full-size. Many capture tools provide export settings for the web, email, print, or other targets. See if you can find the "expert" or "manual" settings to specify the quality you need.
Imaging Tutorial 3: OCR Scanning

Optical Character Recognition is a process that can be applied to digital images of text documents in order to extract the text from the image. The process operates on the principle that a high-contrast image of black text on a white background, with a standard font, can be analyzed to discern characters, building text files for the document. The first step is to acquire appropriate images of the document that you want digitised via scanning or photography as described in previous tutorials.

Once the acquisition is complete, you will have a number of images open in Photoshop (or a similar editor). You will need to save these individually as TIFFs to an appropriate folder with appropriate file names. You now have the images you need to begin OCR.

Now that you have a clean batch of images prepared, you can begin the OCR process. A detailed description of this process can be found here:

http://www.dlib.org/dlib/march09/holley/03holley.html
Cropping and Rotating

Images acquired through scanning or photography will often require cropping and rotation adjustments.

1. Cropping capabilities are most easily accessed via the Crop tool.

2. Select the portion of the image you would like to keep by clicking and dragging your mouse across the image.

3. The cropping box which results can be rotated and resized. Press return on the keyboard once the changes are correct. You can always undo and retry the operation if it doesn’t look quite right.

Adjusting Your Image – Colour and Contrast

Photoshop features many ways to adjust the light balance and colour contrast of an image, procedures used most often to increase the clarity of an image. You will find many options and features for adjusting your image under the Image->Adjustments menu.
The *Levels* tool is a versatile way to adjust the colour and contrast of your image, correcting exposure problems. Its advantages include visually displaying a histogram of the levels of colour and light in your image data, (allowing you to make informed decisions beyond the naked eye,) and immediately *previewing* the effects of your choices.

**Using Levels to Adjust Brightness and Contrast**

Photoshop creates a histogram for the image, charting its tonal levels.

![Levels tool interface](image)

The example above features an underexposed image, which means it is darker than one might like. In particular, it is difficult to discern details such as facial expressions. In order to correct for these deficiencies, we need to rebalance the colour distribution.
By moving the centre slider left, (establishing a new mid-range,) and reducing the total colour range, (by moving the right slider,) we are able to lighten the image considerably. This gives us much more facial detail, for example. To compensate for the foggy overall composition, we can turn now to the *Brightness/Contrast* tool.

In this case, increasing the contrast and decreasing the brightness gives a sharp, crisp image, with much more detail than the original.
Using the Levels Tool to Improve Colour Balance

The Levels tool can also help with colour balance. Click on the pull-down menu above the histogram. This allows you to adjust - with the three adjustment sliders - the range and centre of Red, Green, and Blue, respectively. Make sure you adjust the Brightness/Contrast in combination with the levels tool, or you will likely wash out all the vibrant colours in your image. The Hue/Saturation feature can be used to achieve similar results. The best way to understand these tools is to explore and experiment a little!
Imaging Tutorial 5: Using Layers in Photoshop

Using layers in Photoshop is like placing an overhead transparency over an image. You can write on the transparency, or print an image on the transparency, place it over your original image and it will look like one image.

1. Start by browsing the web for a photo of uVic or Victoria. Try to find a high resolution image - Google will help you with this. Open the image in Photoshop; this will be your background image. Next we'll layer some additional elements on top of the background.

2. In the Layers toolbar, click the new icon. You may want to give the layer a useful name. Be sure this new layer is selected. Choose the marquee tool and draw a rectangle over the image where you might like to place some text.

3. Click on the background colour for the document, and use the magnifying glass tool from the Colors box to select a colour from the background image.
4. Use the *Paint Bucket* tool to fill the rectangle with the colour you’ve chosen.

5. To see how layers work, click on the eye icon next to the layer you are working with. Poof! Your rectangle is gone. Click on the eye to see your layer again. Try adjusting the Opacity setting for your layer to make the box slightly see-through.

6. Select the *Text* tool, and draw a box inside the rectangle you just created. Add some text inside the image, adjusting font and size to be sure the text fits inside the rectangle. Note that using the *Text* tool created an additional layer in your document.

7. Google DHSI, and find a GIF of the tall dark being. Drag or save the logo from the browser window right to the Desktop. You may need to right-click on the image to do this. Open this image in Photoshop. Select the image using Edit->Select All, and copy it. Go back to the postcard image, and paste the logo on top. Note that this creates a new layer as well.
8. Note that, in the image above, the logo has been resized, and edited slightly to fit better with the rest of the image. In addition, the Blend Mode of the logo has been set to *Multiply*. Normally, layers have a Blend Mode of Normal, as seen in the Layers panel, which means layers on top completely cover layers below. This pop-up menu has a whole list of possible Blend Modes, which help layers merge their pixels with the pixels belonging to the layer below. The *Multiply* setting has the effect of making the resulting pixels always darker, and making the content on top slightly transparent.

9. Once you're satisfied with your modified image, choose the *Save As* command to produce a new image for distribution. You may, of course, choose different settings and formats depending on the destination of your media.
SECTION 3: DIGITISING AUDIO

Prepared by Robin Davies, Allison Benner, and Brian Millward
Best Practices for Digital Audio

Whether you are starting with a vinyl record, a cassette tape, or a fresh recording of your own voice, the following guidelines will help ensure your digital audio is suitable for your intended purpose. We will use software called Audacity to demonstrate. Audacity, though free, performs quite similar operations to industry standard software such as Avid’s Pro Tools, or Apple’s Logic.

There are four main factors you should to keep in mind when digitising audio files/media:

- The most common/accessible audio file formats
- How to connect analog audio sources to the computer
- Audio recording standards: sound quality
- Optimizing your computer for recording digital audio

Basic Rules for Digitising Audio Files

The first step in any digital audio project is to determine how the audio will be used. For example, a digital audio recording for streaming on the web has very different requirements than a recording for a CD or Blu-ray. We will begin by reviewing the basic rules for digitising audio files.

Rule 1: Always save a master audio file in an uncompressed format. We encountered this quality issue when dealing with images as well. As with images, when you compress an audio file, you affect its quality in the process of compression. As a result, when you reformat an audio file from one compressed format to another, there will be greater quality differences than if you convert an uncompressed audio file into a compressed format.

Rule 2: Start with the highest quality audio file/media possible and then make the proper adjustments necessary for your project. Start with an original vinyl or reel-to-reel recording, as opposed to a poorly created MP3 file.
Rule 3: Manage your audio files. It is important to use a file naming convention to ensure that files are correctly labeled, and to avoid accidentally erasing or copying over a file when editing your audio files. Also, high quality audio files may take up a substantial amount of storage space. Be prepared for this.

Common Audio File Types

Many audio projects begin with an analog audio source which needs to be converted into a digital audio file. You will have to select a file type that is best suited to your needs. Each file format has a slightly different purpose. WAV and AIF file types are uncompressed digital audio; they are best suited for creating audio CDs, Blu-Ray soundtracks, or an archival master file. Compressed audio formats - MP3, M4A, WMA, OGG - are best suited for computer and web-based presentations.

Apple AIFF (.aif)

The Audio Interchange File Format is the standard for high quality audio files on Apple computers. It supports audio data of various sampling rates, bit depths, and in mono or stereo (or more!). If you want to transfer an audio file from a Mac to a PC, the AIF file format works, because the format can be read by PCs using standard audio software like Audacity.

Microsoft Wave (.wav)

The WAV format is a Microsoft Windows-based audio file format that stores sounds as waveforms. The wav format also supports audio data of various sampling rates, bit depths, and in mono or stereo. If you are working with Windows-based PCs, you will probably use this format most of the time, because it is the most widely supported audio file format. If you want to transfer an audio file from PC to a Mac, the WAV file format works because Macs with standard audio software like iTunes can read it.
**MPEG Audio (.mp3, .mpg, .mpeg)**

MP3 is the audio file format which was the catalyst for the arrival of iPods and the fall of the traditional music industry. Officially known as MPEG-1 Audio Layer 3, MP3 is a form of lossy compression. MP3 music files can be one tenth to one twelfth the file size of the CD format WAV or AIFF file, while sounding exactly the same *to most ears*. This compression is achieved by filtering out elements of the audio not necessarily detected by the human ear using a technique called *perceptual encoding*. Many hardware audio and video players understand the MP3 format, and other file formats more commonly found only on computer systems, but the MP3 is most commonly found online and in portable music players. Audacity can export MP3 files using the LAME MP3 encoder, or you can use software such as iTunes to make MP3s for you.

**Windows Media File (.wma)**

The Windows Media File format is a proprietary compression format used for streaming audio files over the Internet. .wma files require a special media player. For more information about the wma format and to download a free media player, go to [https://support.microsoft.com/en-ca/help/14209/get-windows-media-player](https://support.microsoft.com/en-ca/help/14209/get-windows-media-player)

**Ogg Vorbis (.ogg)**

This compressed audio format was designed as a free alternative to the MP3. Audacity can import and export this format. Ogg Vorbis files tend to take up a little less disk space than MP3s for similar compression quality, and Ogg Vorbis is free from patents and licensing restrictions. Ogg Vorbis files are not as commonly used as MP3 files, though they are used by some newer implementations of HTML.
FLAC (.flac)

FLAC stands for Free Lossless Audio Codec. This is a relatively new lossless, data-compressed audio format. Audacity can import and export this format, and many players support this file format. The ponomusic.com service uses this format. For more information on FLAC, and to download a free Windows installer, go to http://flac.sourceforge.net/

Transferring Analog Audio Sources to the Computer

Even though there are many different types of analog sources (tape players, turntables, amplified voice, and radio), the way they are connected to your computer is straightforward. With both PCs and Macs, the line from the amplified sound source that connects to your computer must be plugged into a port on your computer. You may find a number of audio ports on the computer:

Microphone – this is where you would connect your microphone. You might also use a USB microphone. This port is specially designed to boost the quieter signals which are usually produced by microphones.

Input (Line-in) – this where you would connect your stereo, mixer, or the headphone output of another computer or portable audio device.

Output – this is where you would connect your speakers or a line to a stereo system or other audio recording device.

Alternatively, particularly with newer machines containing fewer ports, you may need an external digital audio interface or recorder which will likely connect to your computer via a USB port. Recorders made by zoom.co.jp are currently widely used, and different models cater to different situations. These kinds of recorders have removable media which your computer may be able to access directly.
**Audio Standards for Recording Audio Files**

To achieve the highest quality audio files, you need to record the audio at the highest sampling and bit rate possible. The sampling rate defines how often the computer records the sound, and is measured in terms of samples per second (Hz or kHz). The larger the sampling rate, the higher the frequency of sound that can be captured, and the better the sound quality. However, the depth of each sample also affects the quality of sound. Just like the sampling rate, the higher the bit depth, the better the quality of sound you can record. The basic guidelines to ensure CD quality audio for music or speech are to sample at 16-bit, 44.1kHz, stereo.

**Archival Standards for Digitising Audio Files**

The questions to take into consideration in the creation and preservation of digital audio for archival purposes are similar to those for the digitisation of text and images: quality, stability, longevity, and accessibility. It is now possible to sample audio at rates of up to 192kHz, as the PonoMusic service advertises, but are the storage needs created by the use of such a high sampling rate worth the information stored? You may choose to make a large collection of audio files accessible to the public in a compressed format such as MP3 (though, of course, you have an uncompressed master file!). Perhaps you have an auditory preference for a new, lossless file format, but is that file format widely used and supported? Will others be able to appreciate the file?

Standards for the creation and preservation of digital audio remain open to debate. However, the most widely used uncompressed file format used for archival master files is the WAV file, sampled at a minimum of 44.1kHz, 16-bit (though 48 kHz or higher, and 24-bit or higher are recommended). For accessibility purposes, the most common compressed file format is MP3.
Optimizing Your Computer for Recording Digital Audio

Most audio recording and editing software is disk-based - you record and edit directly onto the computer’s hard-drive. This allows you to edit large amounts of data and to retain extensive edit history, the ability to undo and redo, also known as non-destructive editing. However, this also means that your hard drive needs to have sufficient temporary storage and free space to store large amounts of data. In addition, a higher amount of RAM in the computer will allow for faster processing and editing, because RAM is much faster than accessing the HD. Therefore, to ensure better performance and to avoid problems associated with incontiguous files, you should run your operating system’s defragmentation program before starting to record audio, close unnecessary applications, and consider using a separate HD.
Resources for Digital Audio

**Software Packages**

Audacity is free, open source software for sound recording and editing, available for many operating systems.

http://audacity.sourceforge.net/

Cakewalk audio software offers a wide range of software packages for professional and amateur musicians.

www.cakewalk.com

Dart software produces digital audio recording and restoration software for all levels of users.

www.dartpro.com

Waves creates digital audio recording and restoration tools.

www.waves.com

Adobe Audition (formerly Cool Edit) offers advanced audio mixing, editing, and effects processing capabilities.

http://www.adobe.com/ca/products/audition.html

No list of digital audio resources would be complete without a mention of Digidesign's Pro Tools software (which is now managed by Avid), the industry standard for recording and production.

http://www.avid.com/US/products/family/Pro-Tools
Audio Standards Websites

http://mpeg.chiariglione.org/standards

No discussion of Moving Pictures standards would be complete without discussing audio!

http://www.vorbis.com/

This website is dedicated to the Ogg Vorbis audio compression format. Here you can find information about Ogg Vorbis and download the encoder. Ogg Vorbis is comparable to other formats used to store and play digital music, such as MP3, but it is completely free, open, and unpatented.

http://flac.sourceforge.net/

This website provides information on the FLAC (Free Lossless Audio Codec) audio compression format. You can also download a free Windows installer for FLAC on this website.
Audio Tutorial 1 – Getting Started

We will use Audacity for this tutorial. Please keep in mind that each of the functions covered here are applicable to other audio editing software. The goal of this tutorial is to demonstrate how to acquire audio files, how to perform basic editing, and finally how to burn audio files to CD. In order to help you understand the many different functions within the program, we have broken the tutorial into three main categories: Input, Processing, and Output. We will begin by looking at all the different ways to input audio media into the software.

**Input**

To illustrate the input mode, we begin by connecting a microphone to the computer. There are several types of microphones available. You will need to make sure that the microphone you want to use is designed for a computer.

Once you have verified which type of computer microphone you have, plug it directly into the computer’s microphone jack, or plug it into a USB port. You may also be able to use the computer’s internal microphone. Once you have connected the microphone, open your audio editing software and begin recording. For an example using Audacity, refer to the section on recording audio within this tutorial.

In order to digitise vinyl records and audio cassette tapes, you will first need to connect the audio device to your computer with an audio cable. Refer to the Best Practices for Digital Audio for greater detail on this point.

Once you have connected your analog audio device to your computer, you will need to open your audio editing program. From your audio editing program you can begin to record.

**Recording Audio**

Before recording audio in Audacity, it is important to save a new project and to set the preferences for any files you will record or import in your project.
Create a project

Under the File menu, select Save Project As... and choose a location and filename for your project. Remember that tools such as Audacity make non-destructive changes to your source material, by creating new audio files when you make modifications to existing ones.

Set your preferences

1. Open the Preferences dialog box (from the Edit or Audacity menu).

2. Under the Audio I/O tab, check that you have selected the correct input and output devices. If you are recording from an analog device, select your sound card. If you want to make a stereo recording, select the appropriate number of Channels.
3. Next, under the **Quality** tab, select the sampling rate and the bit depth for your project. Any files you record within the project will automatically have these settings.

![Audacity Preferences](image)

4. Finally, check that the gain (input level) on the mixer of your soundcard is set to an appropriate level. Depending on your soundcard drivers, you can do this using the volume control settings on your computer, via a Control Panel or System Preferences.

![Wave In](image)

To adjust the gain within Audacity, use the onscreen sliders. The control on the left, marked by a speaker icon, controls your output/playback level, while the control on the right, marked by the microphone, controls your input/recording level.
Record

5. Audacity, like most other audio software, uses a multitrack tape deck metaphor for an interface. Click on the red Record button to begin recording.

6. Click on the blue Pause button if you need to pause the recording, particularly if you’re giving a vocal performance. Any blank sections in the recording still take up space on the HD, and will likely have to be removed later. Note that when the Pause button is pressed it’s impossible to use many of the other features of Audacity.

7. When you are finished recording, press the Stop button. When you stop recording, the start position will reset to the beginning of your audio track.

8. To review your recording, press the green Play button.

JUST FOR FUN:
Ever wanted to hear yourself singing in harmony with yourself? You can do this in Audacity. Record yourself singing the melody line. Then, go to the Quality tab under Preferences in the Edit menu. Select Play other tracks while recording new one. Press Record and sing the harmony in sync with the melody. Press Play and see if you like what you hear. If you do, export the file as a wav; the two parts will automatically be mixed into a single sound file.

"Ripping" Audio from CDs

Many audio editing software packages can import directly from an audio CD. However, when using Audacity, you will need to use iTunes or other CD ripping software program to extract CD tracks into a format that Audacity can read, such as WAV or AIFF.
Opening a File

9. To open a file, you can drag and drop the audio file into the Audacity window, or you can go to the Project menu, select Import Audio, and choose the file you want to open. Audacity can import many file formats.

10. The audio file will display as a waveform in an audio track. If you have made a mono recording, there will only be one window. If you have recorded in stereo, the top window corresponds to the left channel, and the bottom window corresponds to the right channel.

Saving a Project

While editing all the files that are part of your project, you should frequently (as with any other computer task) save your work, in this case with the Save Project function under the File menu. Note that most of the audio data for an Audacity project is not stored in the project file itself, the .aup file, but in a separate folder with the same name as the project plus the _data suffix. When you open the project file, all the audio content associated with the project will automatically open as well. However, until you export sound files that you have recorded, they will not be readable by other players or software programs, so don’t lose track of the .aup and _data folder!
Exporting a File

To save a master copy of the file that is readable by other software programs, you need to export the sound file from Audacity.

Audacity lets you export a file in a variety of formats, including wav, aif, mp3, ogg, and flac. As stated in the best practices, you should first export your project as an uncompressed master by selecting the Export as WAV or Export as AIFF under the File menu. After you have created a master audio file, you can export the project in a compressed format using the Export as MP3 or Export as OGG options under the File menu. When you export a file, Audacity will ask you to specify where you want the file to be stored.

TIP:
If you want to export only a part of your project in a format readable by other programs, select the desired portion, go to the File menu, and use the Export Selection as WAV, Export Selection as AIFF, Export as MP3, or Export as OGG options.
Process

This part of the tutorial will show you how to perform basic editing functions on your audio files. Although editing is usually viewed as a complex process, the following tutorial has been structured to look at each editing function separately. We will also review the benevolent *Undo* command. The most important thing to remember when editing is that most software packages follow the convention of “select then do”.

**Selecting and the Editing Tool**

The easiest way to select a portion of a file for editing is to use your mouse. Simply click and drag in the desired direction until the portion of the sound file you want to select is shaded. If you want to extend the shading beyond your original selection, press the Shift key and drag the selection to the right or left of its original location.

If you are making a selection within a stereo recording, both the left and right audio channels are selected by default. If you want to control selections in the left and right channels independently, go to the drop-down menu under the arrow immediately to the left of the audio track, and select *Split Stereo Track*. 
Copying and Pasting

Just like in a word processor, the copy function allows you to copy a selection without modifying the original file. To demonstrate this, we will quickly review the step-by-step process.

1. Open a project containing an audio file.
2. Create a selection. (To listen to the selection, press the space bar.)
3. From the Edit menu, choose Copy or click on the Copy button.
4. Once you have copied the selection it will remain on the clipboard until you copy another selection. The next step is to paste it into a new location within the project.
5. Move your cursor to where you would like to insert the copied clip.
6. From the Edit menu, select Paste. The selected clip is now inserted at your cursor.
7. Press play to hear the changes.
**Cutting**

As with the copy function, cutting is similar to a word processing program. The Cutting function allows you to cut a selection out of the audio file by removing it and placing it on the clipboard. To demonstrate this, we will quickly review the step-by-step process.

8. Create a selection.

9. From the **Edit** menu, choose **Cut**.

Use the **Paste** command to place it at the end of the audio file.

**Deleting**

Deleting a selection permanently removes it from the project. To delete a selection, press the **Delete** Key on the keyboard.
**Undo, Redo, and Versioning**

You can undo any edit operation by choosing **Undo** in the **Edit** menu. You can redo any undone edit by choosing **Redo** in the **Edit** menu. Some audio editing software offers an editing **History**, similar to the history you may have seen in Photoshop. This allows you to see a list of the processes you have performed while editing, and jump back to a specific location in your work. Audacity will let you step back numerous times. However, it's often beneficial, particularly with a large project, to save separate versions of your work when you reach significant milestones. Choose the **Save Project As...** command, and make a copy of the file, or duplicate your file on the Desktop. Your computer will automatically put a new date on your work, so you can always go back and see what you were working on several days or weeks ago. Often, projects will move in directions that hindsight dictates were incorrect. This technique of **versioning** allows you to evaluate your progress.

**Output**

**Burning CDs or DVDs**

Many audio editing software programs allow you to burn an audio CD. Audacity does not include this feature, but software tools such as iTunes, Nero, or Toast make burning easy.

1. Insert a blank or **unclosed** CD or DVD into your computer.

2. With some burning software, you may be prompted to choose the type of disc you want to burn. If you only want to play your disc in a computer, (or some media systems, such as DVD players) you may choose to burn a **data** disc. However, if you want your CD to play in a conventional CD player, you need to burn an audio or music CD.

3. Next, add or delete the tracks/files you want to burn to your media, and preview how the tracks sound back to back. iTunes accomplishes this by treating the files destined for a CD like any other playlist.
4. Before you burn the disc, decide whether or not you will want to burn more tracks or sessions to this disc. It's very possible you'll only be burning a few tracks or files at a time. If you close the disc right away, it's like renting a large moving truck and only putting one chair inside. The rest of the space is wasted.

5. Don't forget to label your media once it's burned!

**Publishing an MP3**

Audacity is able to create mp3 files, though not *right out of the box* on account of licensing issues; see Audacity documentation for details. You may find it easier to use iTunes or a web-based tool such as audio.online-convert.com to do this work. Use your master .aif or .wav as a starting point; choose encoding settings which produce a file that provides the sound quality you require within the size limits necessary for your delivery method.
Audio Tutorial 2 - Advanced Audio Editing

The following Tutorial will cover advanced audio processing tools. Some features, such as pasting, mixing, and noise reduction, can be performed within Audacity. If you require more advanced, automated editing functions, you may wish to investigate a more specialized audio editing package, such as Sound Forge or Audition.

Pasting

1. If you have already saved an Audacity project file (.aup), open it. Any sound files you previously saved in the project will automatically open, and you may proceed to step 3 below. If you are starting afresh, create an empty project.

2. Assuming you are starting from an empty project, import two sound files into the project, either by dragging them into the Audacity window, or by using the Import Audio function under the Project menu.

3. Select a portion of one of the audio files.

4. Use the Copy function under the Edit menu, or the Copy icon on the toolbar at the top right-hand portion of the screen.
5. Place your cursor at the location in the other audio file where you want the selected portion to appear. Select **Paste** from the **Edit** menu, or press the Paste icon on the toolbar.

6. The selected portion will now be inserted at the location you selected. If you are satisfied with the result of your paste, choose **Save Project** under the **File** menu. If you are not, press **Undo** under the **Edit** menu. Keep in mind that within an Audacity project file, you can still undo a change even after you have saved it.
Mixing

If you are working with several audio files within an Audacity project, one way to approach your audio editing is via the Cut, Copy, and Paste functions described above. Alternatively, you can use the mixing features within Audacity to work with these files. If you want any of the sound files to overlap, for example by combining two parts of a harmony, or combining a vocal line with background sound effects that you have recorded separately, mixing is essential.

Suppose that you have two sound files. The first is a recording of one person talking, and the second is a recording of a different person talking. You want to combine these files such that the first person’s utterance slightly overlaps the second. Mixing can help you do this.

1. Open or save an Audacity project and record or import the sound files you want to mix.

2. Activate the Time Shift tool by pressing on the button in the upper-left-hand corner of the Audacity window. This tool will allow you to move each audio track around within its own space, allowing you to adjust the position of each sound file relative to any others in your project. In the example below, the beginning of the second sound file has been shifted to the right, and it slightly overlaps with the first sound file.

TIP:
If you have more than one sound file open within an Audacity project, and you press Play, both files will play simultaneously. To listen to one of the files without the interference of the other, press Mute on the window immediately to the left of the audio track. If you want to hear the files playing together again, simply press Mute again on the sound file you previously silenced.
3. When you are pleased with the way the two sound files sound relative to each other, save the project and export the project as a wav. The exported wav file will be a mix of the two sound files.

**TIP:**
If you are working with several audio files in a project, and you want to see them all at once on the screen, select Fit Vertically under the View menu.

**Silencing Unwanted Sounds**

If your sound file contains an unwanted noise such as loud breathing, the sound of a table leg squeaking, or something falling on the floor, and if this sound occurs in isolation, you can either cut it out completely or replace this portion of the sound file with silence.

To use this feature, use your mouse to select the unwanted sound. Next, under the **Generate** menu, select **Silence**. A small window will appear specifying the length of time of your selection. Press **Generate Silence**, and your selection will be replaced with silence.
If you have performed this operation, you may find that the result is an unnatural drop in ambient noise. To minimize this effect, select a portion of the file before the silence, go to the Effects menu, and select the \textbf{Fade out} function. Then select a portion of the file after the silence, go to the Effects menu, and select the \textbf{Fade in} function. In your selections of the portions to fade in and fade out, keep in mind the general principle that you should fade in quickly, and fade out slowly. This is the way most real-world sounds occur.

\textbf{Click Removal}

If your recording contains unwanted clicks and pops, you can use the \textbf{Click Removal} feature in Audacity to remove them. To take advantage of this feature, follow the steps below.

1. Open a file that you want to edit for clicks and pops. Using your mouse, select the portion of the file that you want "cleaned."

2. Under the Effects menu, select the \textbf{Click Removal} function. A window will appear that will allow you to preview what your sound file will sound like following click removal, based on the default settings on the sliding threshold scales. Press \textbf{Preview} to see if you like the result. If you detect no noticeable difference, or if the result sounds distorted, adjust the threshold accordingly. When you are satisfied with the result, press \textbf{Remove clicks}.
Noise Reduction

Audacity has a **Noise reduction** feature within the **Effects** menu. While this feature is not always effective in removing noise without introducing artefacts or distortion, it can work well with some types of noise, such as fans, tape noise, or hums.

1. Open a sound file you want to rid of background noise. Using your mouse, select a portion of the file that contains an example of the background noise *on its own*. Often you can find a good example at the beginning of the sound file.

2. Under the **Effects** menu, select the **Noise removal** function. A window will appear outlining a two-step process. In the first step, Audacity will use the noise selection to create a noise *profile*. To activate this feature, press **Get Noise Profile**.

3. Press **Preview** to hear the results of the noise removal based on the sample you have provided and the noise removal level indicated by the sliding scale in the middle of the Noise Removal window. If you find the result distorted, try sliding the gauge towards less noise removal. If you are satisfied with the anticipated result, press **Remove Noise**.

These features which remove clicks and noise are handy. However, the best way to deal with noise is to have a clean recording in the first place. If you have control over the sound quality and recording environment, do your best to keep things quiet and tidy from the outset.
SECTION 4: DIGITISING VIDEO

Prepared by Robin Davies, Allison Benner, Derek Finstad and Laurel Fulford
BEST PRACTICES FOR DIGITAL VIDEO

**Digital Video**

Digitising video involves three steps: capturing the video (transferring video from a video camera or another source to the computer), manipulating the video, and exporting the video to a file that can be viewed in a potentially wide array of situations.

Digital video is data displayed as pixels which constitute the individual frames of a video sequence - moving pictures. The quality of the video is determined by two main factors: frame rate and bits per pixel. A high frame rate produces a smooth moving image, while a low frame rate produces a choppy moving image. The number of bits per pixel determines the colour quality.

While uncompressed file formats and lossless compression methods are often used for digital audio, lossy compression is the norm in digital video, even for archival purposes, because the large amount of information contained in digital video files makes lossless compression impractical. The most widely used compression for video is MPEG, which is available in MPEG-1, MPEG-2, and MPEG-4 schemas.

**Shooting Video**

Here is a short list of things to keep in mind when shooting video. Some of the information provided is exclusively for shooting video for the web.

- Use a tripod, especially if you plan to use a zoom lens.

- Use close-ups if your video will live online. Close-ups are better for the often small format used by mobile devices, YouTube, and similar sites.

- Frame the shot; leave a little breathing, looking, and talking space around your subject.

- Don’t move the camera too quickly. Avoid the Blair Witch effect.
Backgrounds should not be busy or similar to your subject.

Backgrounds should not be exceedingly light or dark; some cameras will compensate for these backgrounds by lightening or darkening your subject.

Make sure the timecode is disabled in the camera, unless you want the timestamp to appear in your video permanently. It is possible to erase this timestamp in your final product, but it is very tedious.

Leave extra time at the beginning and end of each shot; this time can be removed when you edit the video, but it is difficult to add.

Avoid jarring *jump cuts*. A jump cut results when there is a gap in the flow of action, often introduced during the editing process. To avoid this, make sure you have footage that can be inserted to create *continuity*.

Minimize changes and movement from frame to frame. This is an internet trick, since video compression often functions by storing only information that is changing. More *complicated* video will take longer to compress, and will likely be a larger file.

Bright whites, yellows, and blues (and other hot colors or glares) are not as attractive as darker tones in video for the Internet. Try to make sure your subject is wearing dark, solid colors, as opposed to crazy patterns.

Set your audio levels as close as you can to what you want in your end product. Aim for as *hot* a signal as you can get without *clipping* or *distorting* the sound. Do some tests in advance to ensure the audio setup is functioning correctly. It’s *extremely challenging* to compensate for missing or bad audio after the shoot!

Save camera tricks for the editing stage. Then you won’t have to include them if you change your mind.
When shooting a close-up of someone speaking, remember that excessive hand gestures can be distracting. Try to frame your subject's head and shoulders. If the subject is standing, provide a chair for steadiness. If the subject is seated, encourage a straight back, thus simulating a standing posture. Otherwise, the subject may look slouched.
Resources for Digital Video

http://video.online-convert.com/
This is an all-purpose, online media convertor for audio, video, images, text files, and more!

http://www.mediacollege.com/
This site provides information and tutorials on digital audio and video recording and processing.

Check here for information about the current video compression standards used by Blu-ray Discs, YouTube, and others.

http://websitehelpers.com/video/
Though it looks a little antique, this is a great discussion about web video formats.

https://www.youtube.com/takezero
Though they haven’t been very active recently, Take Zero delve into all aspects of video production.
(Some) Codecs Described

H.264

You may come across media encoded in H.261 and H.263 formats. H.264 is the latest in this line of codecs, and uses a process similar to MPEG4 to reduce the data size of videos. This codec is used for viewing videos over the Internet, as the file size and data rate are small. However, as the compression system is quite complex, a fast CPU is required to decode the data. As a result, a machine with a fast internet connection may be able to acquire the data quickly, but be unable to decode the data to display it.

DV-NTSC

Using DV, which some camcorders still use, results in very clear footage with large file sizes. NTSC is the SD standard for television in North America. Your storage is limited to the number of tapes you can purchase.

Animation

This codec will compress your animation files but will not damage the visual effect. The codec works best on images with large areas of solid colour, as found in animations. File sizes are still quite large.

Flash Video

This .flv codec was traditionally used online by sites such as YouTube. There are many free players for .flv content, and some software will still export in this format. The .flv format belongs to Adobe (formerly Macromedia).
SECTION 5: WEB PRODUCTION

Prepared by Calleigh Lim, based on material prepared by Michael Nixon, Robin Davies Cara Leitch and Gerry Watson
**Introduction**

This section will provide an overview of techniques and technologies associated with web production at a fundamental level, and examine how different types of digital materials might be incorporated into a simple website. The section that follows will cover advanced topics spanning the creation of web templates, and modern design techniques.

Most DH projects incorporate web based components and technologies. It is important to note only know how these technologies work, but also their associated affordances and limitations in terms of online presentation and interactivity.

**The World Wide Web and Digitization**

The World Wide Web refers to the collection of inter-related documents and resources that are made up of computers, servers, and other equipment and allow the communication of information across the globe. Hailed by some as “the Gutenberg press of our time,” the web presents a number of interesting opportunities and challenges for scholarly projects and digitization.

While it can be tempting to present your information “as is,” there are major benefits to starting your web project by planning a proper website. You should begin by outlining your **information architecture**, how your information will be organized and how users will navigate your site (this can be done using flash cards or post-it notes). The next step is to design the site such that it’s not only aesthetically pleasing, but also accessible, intuitive, and scalable. For projects in the digital humanities where communication of information and scholarly collaboration are so important, a well-designed and accessible website can prove to be an essential tool.
**The Basics of HTML**

HTML, or *Hypertext Markup Language*, is the foundational building block of all website and web applications. It is a coding language that tells browsers what is to be included on a web page, how it is to be displayed, where to reference external resources like images, and where to send people when they move elsewhere. As you will see, it is a little more complicated than this, but this is a good way to begin thinking of HTML. The modern form of HTML is HTML5, a markup standard put forth by the World Wide Web Consortium.

Read more about HTML5: [https://www.w3schools.com/html/html5_intro.asp](https://www.w3schools.com/html/html5_intro.asp)

**The Building Blocks**

When building a website, there are two layers of production worth keeping in mind: the **Content Layer**, and the **Presentation Layer**.

**The Content Layer** comprises all the material that needs to be presented on screen, from the text of a link, to blocks of text, videos, or images. Some of this content is held internally in the website markup, such as text and links, and some of it is opened by the browser as it loads, such as images and videos. The markup, written in HTML (often referred to as code, tags, or elements), surrounds the content and tells a browser more about what a piece of content means, its structure, and how to display it.

**The Presentation Layer** deals with Cascading Style Sheets and a website’s aesthetics and layout, and will be touched upon further into this section.
**Getting Started**

The first step to building a webpage is to equip ourselves with the right tools, a good text editor and a modern web browser. Previously, web production was done by hand, with code written in a simple text editor. These days, most web developers use more sophisticated text editors, while others fall back on integrated design environments that help fill in pieces of code. For this course, we will focus on the basics by using a moderately sophisticated text editor.

Atom Text Editor: [https://atom.io/](https://atom.io/)

In addition to a capable text editor, a modern browser is essential to the web production process. Browsers such as a Google Chrome and Mozilla Firefox feature a 'developer' mode or tool that aids with debugging. On Chrome, for instance, you can Right-Click an element on a webpage and click inspect to view the markup, content, and styles applied to it.

Google Chrome: [https://www.google.ca/chrome/](https://www.google.ca/chrome/)

Starting Template

To begin building a webpage, we begin with a proper starting template – this is sometimes known as a boilerplate. You can find many boilerplate examples online, but here's a basic one:

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Your Website Template</title>
  </head>
  <body>
    <!-- Your content goes here -->
  </body>
</html>
```

Web pages use the DOCTYPE in order to specify the version of HTML that they rely on for validation. Meta is used to tell the browser metadata about the website.

Once you copy and paste this template, you can begin to make changes where it says “Your content goes here”. You can then save it on your computer as a starting point. Save your document as plain text, with the file extension .html.

Congratulations! You've just made and saved a web page that we will use for the exercises to follow.
**Tag, You're It**

A tag, sometimes called an element, is markup that surrounds or wraps a piece of content, giving the browser specific instructions about how to display the content and what it is. Tags are the parts of code that make web programming scarier than it actually is. All tags have this basic syntax:

```html
<tag attribute="value"> content </tag>
```

The **tag** is the instruction that tells the browser to present the **content** in a certain way.

The attribute modifies the tag in a certain way, which is specified by the value. All tags have a certain set of attributes in common, but many have attributes specific to themselves. Each attribute has certain values or ranges of values that can further specify the instruction. To illustrate this point, let’s examine the following line.

```html
<div class="greeting">Hello World!</div>
```

In this example, **div** is the name of the tag, while **class** is the attribute with a **value** of “greeting”.

**Angle Brackets**

At this time, you may be wondering what the angle brackets are for. When a browser reads your HTML files, it looks for angle brackets to signal pieces of code – it tells the browser which snippets are instructions, where those instructions end, and what they apply to.
You may have noticed above that the example was wrapped by the tag in this format:

```html
<div class="greeting">Hello World!</div>
```

The first instance of the tag, the **opening tag**, along with its attributes and values, is enclosed in angle brackets so that the browser can recognize that there is an instruction to be applied to the content, which begins after the first closing angle bracket. After the content, there is a **closing tag** enclosed with angle brackets once again, but preceded by a forward slash. The closing tag tells the browser where to stop applying a piece of instruction. Without the slash, the browser won't know where to end a tag, and will assume that the closing tag is really the start of a new piece of instruction.

One way to think of it is as follows:

```html
<begin instruction>insert content here</end instruction>
```

Even experienced web developers sometimes forget an angle bracket, forward slash, or closing tag, which can cause problems with your markup and web page. If problems occur, make sure to check that your syntax is correct – that every tag opens and closes in an appropriate manner. This is complicated during visual inspection by the fact that web browsers tend to show content even when a faulty page is loaded – they often “guess” with varying results!

To facilitate debugging, most good code editors recognize HTML and will colour code and even close (balance) your tags accordingly (e.g. Coda). Alternatively, use a browser’s developer tools to examine your live code for mistakes.
**Nesting Tags**

It is common practice to nest multiple elements within one section of content. Imagine that we want to take the phrase “Hello World!” and make it a paragraph that is emphasized, while rendering only the world “World” in bold. The code would look something like this.

```html
<p><em>Hello <strong>World</strong>!</em></p>
```

While this looks daunting, it can be easy enough to following by breaking it down. First, we’ll want to add some whitespace (i.e. line breaks and tabs) for better human readability.

```html
<p>
  <em>>Hello <strong>World</strong>!</em>
</p>
```

As you can better see, the `<strong>` element is housed in the `<em>` element, which in turn is housed in the `<p>` element. The `<strong>` element is, as such, a direct child or descendant of the `<em>` element, while the `<p>` element is the parent or ancestor of its child elements. Let’s break this down piece by piece:

“Hello World!” is wrapped in one `<p>` tag to tell the browser that this is a paragraph: it will likely be given vertical space away from other text to demarcate it. There is an `<em>` tag to make the text emphasized (usually in italics). However, only the content “World” is wrapped in the `<strong>` tag, and will appear bold.
Notice that each tag is opened and closed in the same order. These are properly nested tags. Be careful always to close your tags in the proper order. Sometimes you can get away with improperly nested tags, but it can cause major display problems, is confusing to read, and most importantly, isn’t HTML best practice.

So Many Tags, So Little Time...

This is a good time to mention that there are MANY different tags that you can use to do a variety of things, from organizing the internal structure of your web page, to linking to other sites, changing the format and style of your text, building tables and web forms, and so on. If you are new to web production, it can be overwhelming and frustrating to work with all these tags – think of it as the vocab and grammar of a whole new language! If you want to learn more about tags, visit the links below.

HTML Tags: https://www.w3schools.com/tags/default.asp

## Common Tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
</table>
| headings | There are six levels of headings, from `<h1>` to `<h6>`. The higher the level of a heading (i.e. `<h1>`), the larger the heading will appear. Headings should be planned and implemented hierarchically (i.e. `<h3>` follows `<h2>`, and `<h1>` should be at the very top). Examples:  

```
<h1>This is a top-level biggest heading</h1>
<h3>This is smaller than the h1</h3>
```
| paragraphs | The `<p>` tag denotes textual content that's presented as a paragraph and should be treated as such. It may also be used as a general catch-all for textual content. A new set of `<p></p>` tags denotes a new paragraph. |
| phrases | Phrase tags denote text that has additional meaning. There are several types of phrase tags, each with their own meaning and style. These tags should be used meaningfully, and not imply for their look. Examples:  

```
<strong>This text is bold.</strong>
<em>This text is italic.</em>
<code>This text appears in monospace font.</code>
```
| lists | There are two main types of listed, ordered or unordered, denoted by `<ol>` and `<ul>` respectively. By default, each list is displayed indented with numbers for `<ol>` and bullets for `<ul>`. Individual list items are denoted by an `<li>` Example:  

```
<ul>
  <li>List Item 1</li>
  <li>List Item 2</li>
  <li>List Item 3</li>
</ul>
```
| images | Images are denoted by an `<img>` tag and do NOT require a closing tag. They do, however, require a `src` attribute with a value that specifies the location of the image. `title` and `alt` attributes are optional, but recommended – they provide textual information associated with the image should the file not load. Example:  

```
<img src="images/puppy.jpg" title="Puppy" alt="A small puppy.">
```
| links | Hyperlink elements may be created using the `<a>` tag, and require a `href` attribute with a value that specifies the link destination. `<a>` tags may be used to link internally within a website, or externally to an off-site resource. Example:  

```
<a href="https://destination.ca/">This is an external link.</a>
```

These are just some examples of common tags for you to try out. We will learn more as we go along. Refer to the W3Schools or Mozilla reference above for more information.
The Framework of a Web Page

Now that you know basic HTML, we can take a closer look at the framework of a web page. There are two main parts of a web page that need to be understood, the `<head>` and the `<body>`. The `<head>` contains code that gives the browser and anyone looking at your source file information about the web page and content. The `<body>` contains all the content, structure, and formatting for the website, which has been our focus thus far.

The `<head>` is NOT displayed when a browser loads. It holds information such as the `<meta>` tag, which describes metadata associated with the HTML document. You should also see a `<title>` tag, which denotes the title bar of your web page as well as bookmarks. To change the title of your web page, simply edit the text wrapped by the `<title>` tag and refresh your web browser to view how the title bar of the window has changed.

The `<body>` section of your website houses all the content you want to display on screen when a web page is loaded. This section can be as long or short as the content you have, and is often split up into subsidiary and child sections.

Semantic HTML

Before HTML5, web developers fell back on using `<div>` and `<span>` tags as generic catch-call containers for any kind of content. Typically, they’re used to create the overarching structure or layout of your content. For example:

```html
<div class="section">
  <div class="section-inner">
    <h1>Your header here</h1>
    <span>Your content here</span>
  </div>
</div>
```
With the standards set forth by HTML5, there is now a stronger emphasis on meaning over aesthetic in terms of your code design. This means that the tags you use should semantically make sense to the browser. To facilitate this, new tags were recently introduced including `<header>`, `<footer>`, and `<section>`. According to HTML5, the snippet above may be restructured as:

```html
<section>
  <article>
    <h1>Your header here</h1>
    <p>Your content here</p>
  </article>
</section>
```

Coding semantically enables browsers to parse the significance of any piece of content. For more information on how to use specific semantic structural tags, view the flowchart at the link below.


**Validating Your HTML**

The best way to ensure a stable foundation for your web page is to design it to certain standards. This means choosing a set of rules against which you can check your site for compliance. In HTML, the rulebooks come in the form of Document Type Definitions, or DTDs. A DTD is a digital rulebook that specifies how tags may be used. By choosing a DTD, you declare that your site plays by a certain set of rules, making it more consistent and reliable. The DTD you choose will dictate certain aspects of your site, but also give you a way to validate your code.
Notice the very first line of the HTML template starts with `<!DOCTYPE>`. This line tells your browser which standard the web page should comply with. For now, let’s validate index.html:

- First, make sure your file has been saved
- In your browser, load: [https://validator.w3.org/](https://validator.w3.org/)
- Upload your file and click “Check”
- You can click on the errors to learn about issues with your file
- Fix any errors you just created and save index.html – you now have valid HTML!

**Coding Exercise**

Before we move on to aesthetics, let’s test your newly learned knowledge on HTML! Here are some snippets for you to examine – see if you can find the errors without the help of a HTML validator:

```html
<p>The quick brown fox jumps over the lazy dog.<p>

<h>The Times Newspaper</h>

<img src="puppy.jpg"></img>

<ul>
    List Item
    List Item
    List Item
</ol>

<a>Click here to learn more!</a>
```
Cascading Style Sheets

One of the goals of standards-compliant web development is to separate content from aesthetics. This means that your HTML file should only contain content and information about the content, while the aesthetic of the website, the Presentation Layer, is handled by Cascading Style Sheets. Cascading Style Sheets, or CSS, involve a separate coding language altogether. However, CSS interacts with HTML to tell your browser how certain elements should appear.

The Joy of Web Design

CSS is fairly simple. That is why many people enjoy the styling aspect of web development. Before we put our creative muscles to use, we first need to understand how CSS works. Here’s an example of a CSS snippet:

target {
    property: value;
    property: value;
}

The target is what you want to change. Everything between the curly braces denotes what to do with the aforementioned target. Between the curly brackets, you will include pairs of properties and values. A property is basically an instruction, similar to an HTML attribute in that it specifies what aspect of the target to change (e.g. color, font family, size, etc.). The value specifies what is to be done with the property. These property-value combinations are sometimes called declarations.
For example, if you want to change the color of text in `<p>` elements to red, you’d use the following snippet:

```css
p {
    color: red;
}
```

This may seem a little strange at first, but once you get the hang of CSS and learn all the ways it can affect your website, you will begin to understand just how powerful CSS can be.

**A Lesson In Targeting**

Targeting is often a tricky subject, and can sometimes trip up even the most experienced of web developers especially with projects that incorporate hundreds upon thousands of lines of HTML code. Thankfully, there are a variety of ways to target your HTML elements. At a fundamental level, there are three methods:

- You can target using a tag, example: `p { color: red; }
- You can target using an id selector, example: `#paragraph { color: red; }`
- You can target using a class selector, example: `.paragraph { color:red; }

**id** and **class** are attributes that is included within HTML opening tags. They’re essentially names for your tags (that you create) so that your CSS can better find specific elements to target and change. Functionally, ids and classes are quite similar. However, an id is a unique name, and specifically refers to a single tag and, by extension, the content governed by that tag. Classes, are catchalls, and may be used multiple times across a HTML page.
Here are examples of classes and ids in action:

<div id="section">This div element has an id of 'section'.</div>

<h1 class="primary-heading">This h1 element has a class of 'primary-heading'.</h1>

<section id="about" class="full-column">This section element has both an id and a class.</section>

**Advanced Targeting**

Basic targeting rules aside, CSS allows for certain advanced targeting techniques that make our lives even simpler! For instance, multiple elements, classes, and ids may be targeted simply by spacing each target with a comma, as follows:

```
h1, h2, h3, p {
    color: red;
    font-size: 24px;
}
```

Stacking these selectors allows us to style multiple elements at the same time. In the example above, the declaration denotes that all <h1>, <h2>, <h3>, and <p> tags will have a red font color, and have a font size of 24 pixels.
In CSS, there is often an emphasis on specificity, especially when dealing with nested parent and child tags. To target a tag within a tag, separate each target with a single space. Examples:

```css
div#header h1 {
    this targets a <h1> element nested within a div element with the id 'header'
}

nav ul li a {
    this targets <a> elements nested within <li> items of an <ul> unordered list within a <nav> element
}
```

**The Box Model**

A key fact to understanding CSS and building scalable and aesthetically pleasing webpages is mastery over the CSS Box Model. The Box Model simply refers to the way in which browsers render HTML elements. Effectively, all HTML elements exist as blocks/boxes, and these blocks/boxes each have three main layers that wrap around them: **padding**, **border**, and **margin**. These layers may be denoted in your CSS declarations using their associated properties.

Understanding the Box Model and these three layers is essential to creating complex web layouts as they allow us to define the amount of 'breathing room' that separates our content chunks.

For more on the Box Model: [https://www.w3schools.com/css/css_boxmodel.asp](https://www.w3schools.com/css/css_boxmodel.asp)
Adding CSS to your Website

You now have some grounding in CSS, and how it interacts with your HTML code. Here’s the clincher: how do you let the browser know that it should use CSS to style your site? This is done with a simple instruction line that you should include in the <head> section of your web page:

```html
<link rel="stylesheet" href="css/style.css">
```

Take notice of the href attribute as it shows the location of the CSS file to retrieve relative to the HTML web page. This line essentially tells the browser to use the instructions in the CSS as it loads the rest of the page. There are certainly other ways of including CSS in your web pages (e.g. inline styles). However, keeping your CSS and HTML files separate is highly recommended.

CSS Inspiration and Resources

- Stack Overflow: [https://stackoverflow.com/](https://stackoverflow.com/)
- CSS Tricks: [https://css-tricks.com/](https://css-tricks.com/)
- W3Schools CSS Reference: [https://www.w3schools.com/cssref/default.asp](https://www.w3schools.com/cssref/default.asp)
**Building Websites**

All websites start with a single page. What do you do, then, if you have two major topics and want to give them equal emphasis? You make a new page and link it with a special tag. This is the easiest way to divide your content and navigate to and from independent sections. You can repeat this step as many times as necessary. Before you do so, however, it is advisable to plan your website's information architecture to better grasp your site’s user flow and how the pages will work together.

Just like any digitization project, your digitization website requires some planning before you go any further. Some useful questions worth considering in your planning process include:

- How do you wish for your content to be presented?
- Will your content emphasize textual or multimedia elements?
- What kind of aesthetic would you like your website to have?
- How accessible will your user-flow be?
- As you digitize more material, will your site be able to grow and accommodate it?
- Will you be easily able to update both the content and the style of your site if need be?
- Will your site be built with future functions and features in mind?
Web Development: Theory and Practice

At its core, web development should first begin with the planning of your content. Following this, you can decide how you want your content to look, and how your users will interact with said content. You can think of your web design’s aesthetic on a micro or macro level, from the styling of individual links and buttons, or the larger scale of overall look, whitespace, page flow, and layout. The best-looking and easy-to-navigate websites pay close attention and aim to strike a balance between these and other facets.

Paper Prototyping

As with all grand plans, our planning process should begin at the drawing board, or at least on piece of paper. There are two main goals you should try and aspire to at this stage: outlining your information architecture, and sketching wireframes for your webpages.

Defining your information architecture early is crucial to as it will determine much about how your content is laid out, how a website’s hierarchy of pages will be organized, and how easy your website will be to navigate. This can be done using a nested list in a word document, or by using visual aids such as mind maps, flow charts, or post-it notes and flash cards. As a rule-of-thumb, content should be housed and organized no more than two or three clicks away from your home page. Sketching wireframes, on the other hand, gives you a chance to visualize your website’s major divisions and sections, such as your main content area, the header, the footer, and navigation. Think about where you want each element to appear, and how they will fit together.
It is important to catch and retain a viewer's interest. When a first-time viewer visits your site, he or she should be able to tell at a glance who is responsible for the site, what the website is about, and what sort of information it contains. Be sure to consider the following:

- The name of your project/institution: Think of this as the title of a book. Displaying the name of your project or organization with an eye-catching graphic will give the viewer a visual to associate with your name.

- An interesting overview of the purpose of the site or the goals of your organization: This is your site's subtitle. It should be informative enough to answer the question: What is this site about? However, it should also be short enough to read at a glance.

- A simple, catchy, and informative navigation bar: This is your site's table of contents. Keep the information simple and concise so that the viewer will know where he or she is going when they click on a link.

There are all things to remember when sketching out your sites. Keep track of sites you like (or dislike) and don't be afraid to use your browser's developer tools to look 'under the hood' of your favorite pages – looking at another developer's HTML/CSS is a great way to learn!

**Using A Template**

Once you've decided upon your content and layout, you should then set about the construction of a boilerplate (see above). A boilerplate/template is like a cookie cutter for websites. You don't need to build a new page from scratch. Instead, simply
keep a 'clean' version of your boilerplate on hand, and re-use it whenever you build new websites and web pages.

Effectively, all you have to build is one skeleton page that meets your every need. Whenever you want to create a new page or site, simply fall back on the saved boilerplate (copy and paste it) with a new file name and any new content.
SECTION 6: WEB 2.0

Prepared by Michael Nixon
While the term remains somewhat controversial due to being criticized as a vague buzzword, “Web 2.0” still retains value as a description of the latest trends to influence businesses and their use of the internet. In 2001, the so-called dot-com bubble burst, and many businesses with shaky profit models collapsed. Based on the observation that many web-based companies survived and had important elements in common, O'Reilly started a conference series in 2004 to highlight the characteristics. Rather than describing an updated technical specification, which 2.0 usually indicates, Web 2.0 focuses on specific characteristics that define many successful web ventures:

- rich user experience
- user participation
- dynamic content
- metadata
- web standards
- openness & freedom
- collective intelligence

To help you understand this movement, each of these elements will be described in terms of their contribution to the web you use, along with examples and ways your own work can incorporate them.
Tim O'Reilly described a hierarchy of four levels for websites:

4. applications that work equally well offline or online (e.g. Google Maps)

5. applications that operate offline but gain features online (e.g. Google Docs or iTunes)

6. applications that can operate offline but gain advantages from going online (e.g. Flickr and its community-contributed tags)

7. applications that exist only on the Internet (e.g. eBay, del.icio.us, Wikipedia)

He summarized the “core competencies” of web 2.0 business as follows:

8. Services, not packaged software, with cost-effective scalability

9. Control over unique, hard-to-recreate data sources that get richer as more people use them

10. Trusting users as co-developers

11. Harnessing collective intelligence

12. Leveraging the long tail through customer self-service

13. Software above the level of a single device

14. Lightweight user interfaces, development models, AND business models

One of the reasons for an educator to understand these trends comes from Prensky's identification of generation-Y (born in the 1980's and early 1990's) as “digital natives”, in contrast to “digital immigrants” who were born earlier. He
argues that continual immersion in contemporary technologies changes the way digital natives learn. Rather than simply shortening their attention span, it changes the way they prefer to learn to methods that include hypermediation, random access, and play. According to educational marketers, understanding how students access information directs us to make it available in more applicable ways.

**Rich User Experience**

In order to keep visitors invested in a site, delivering a complete experience similar to a desktop application became crucial. Browsers could deliver “applets” using Java and DHTML in the early 1990’s, but these were clunky, being slow to load and poorly integrated into the browsing experience.

*Google’s Gmail and Google Maps are the canonical examples of web-based applications with rich user interfaces. The combination of technologies used in them was named “AJAX” in an essay by Jesse James Garrett and includes:*

- standards-based presentation using HTML and CSS;
- dynamic display and interaction using the Document Object Model;
- data interchange and manipulation using XML and XSLT;
- asynchronous data retrieval using XMLHttpRequest;
- and JavaScript binding everything together.

This explains the rise in importance of standards-based design, openness, and the importance of several particular technologies (notably XML and Javascript).

Primarily, this impacts you in the range of web applications you have access to, and the ease with which they can be used.
**User Participation**

One key thing that users of a site can add is their own unique data. Sites succeed that involve their users who in turn add value to it. Whether this involves breath-taking photos such as Flickr’s content or the amateur video hosted by YouTube, visitors now seem comfortable adding their own content. Another way users are involved is in testing. Given that many web applications stay in a perpetual “beta” state (again, Google provides the most obvious examples), new features can be released to a large audience to test their reception.

Taking advantage of the large number of users who aren’t in the conventional majority has been referred to as harnessing the “long tail” of the Internet. Small sites make up the bulk of the internet’s content; narrow niches make up the bulk of internet’s the possible applications.

The focus on user participation means that researchers and students can often taken advantage of a wide range of contributed data in a particular subject. Based on the community’s value of freedom, this is often available for reuse. As well, you can in turn share some of your own content.

**Dynamic Content**

In order to keep sites up to date, you used to have to access the pages marked up using HTML and add more content, then save and up the pages. The growth of database-backed sites has led to the ability to edit content “on the fly” through a form, often without using HTML yourself. Sites that do this are generically called “content management systems” and include blogs, wikis, and all sorts of other ways to edit content in an easy way.

**Metadata**

As the web grows more data-centric and various sources release their information through the use of APIs (programming interfaces that control how you can access the data), it’s grown more important to accurately describe your data. This is
especially true because the most frequent browsers of your web site are search engines which try to use metadata to determine the overall importance of your content. This growing trend means that formal metadata schemas such as Dublin Core have grown in popularity.

**Web Standards**

The web has changed a lot since the 1990’s when Internet Explorer dominated the market and frames were the latest innovation in browsing technology. The arrival of a competing browser whose adherence to specifications made Internet Explorer look dated was one factor that helped things change. Now that the web has so many “moving parts” in the form of interacting technologies, it’s become important that web programmers and designers work to standards. For page coding, this means HTML 5, and CSS, the style sheet language for visual effects. For client-side interaction, this means unobtrusive Javascript packaged in libraries, working in the background and showing up as little as possible in the actual html file.

On the one hand, the free-for-all that was creating websites in the 1990’s made the barrier to entry really low. On the other, many of the interactions that occur now can only happen because we keep to standards-based design.

The Web Standards Project [http://www.webstandards.org/] is one group that advocates for simple and accessible web technologies.

**Openness & Freedom**

People used to be reluctant to share ideas in order to keep an edge over competitors. Now people and companies tend to loosen hold over these resources because they reap more benefits in doing so. By allowing others to share ideas, their products are able to gain significant improvement and scrutiny through collaboration. This was equally true for code specifications or collections of data. Now in order to reach more users, sites are making their information and tools available for general use. One particular motivator of this was the rise of the “free and open-source software” movement, which holds that software should be
developed for the good of the community and released for general adaptation and modification. This spirit of production is line with most research endeavours, and means that the production of knowledge and community enrichment is a generally appreciated goal.

Besides making tools and data freely available, another outcome has been the ability to migrate data in and out of sites. By not creating an environment that traps users' data in an attempt to keep them on the site, sites gain respect and the ease with which users can return with their data means that they're likely to have them do so.

**Collective Intelligence**

The rise in using user-created content has led people to try use this force to accomplish goals. For example, Amazon has a “Mechanical Turk” project that recruits people to perform “human intelligence tasks” like group photos. Less commercially, this refers to the fact that a group sharing knowledge has a larger combined knowledge base. This means that many of the community that have formed on the web tend to collectively know a lot about the subject that binds them, and can provide an excellent resource. This takes a different and more persistent form that the latent potential of a mailing list, owing to the ongoing conversations that are stored in many of the new media forms.

**Examples**

**Social Networks**

A social network service is a web site that builds an online community of people who share interests by offering ways of sharing information about themselves and explores that of others. These sites often provide ways of interacting such as e-mail and instant messaging. The most popular services are strongly themed and often focus on classmates and friendships; examples include MySpace, Facebook and Twitter. Social network services rely on user participation, since all their content is user created. They are the culmination of almost all the Web 2.0 trends. These
services provide a means of becoming connected to colleague and student alike, but beware the privacy concerns that arise from such open access.

**Mashups**

A Mashup is a web application that combines data from multiple sources into a single integrated tool. It is made possible through access to open APIs and freely available data sources. To obtain content, people use web feeds (e.g. RSS) or information from third party databases such as Amazon, eBay, Flickr or YouTube. This first occurred when people used Google Maps to plot interesting data like real estate or 911 calls.

The advent of mashups has led to a “remix culture” which thrives on the recombination of cultural artifacts.

Creating a mashup is an ideal way to take advantage of existing tools; whether you have your own source of data, or you’d just like to see how a third party data set could be visualized, mashups let you publicize that intersection.

**Blogs**

The idea of a web log or online diary has been around since before Web 2.0, of course. What’s new is how commonly they’re used and the maturity of the platform. All blog software typically offers chronological linking and sorting, categorization, and is easily searchable. The popularity of blogs means that people with insights from our research areas or favourite hobbies are writing about it, and what’s more, able to receive comments and interact in a public space. Blogs have really replaced the static “personal home page” with a dynamic, social space. Of course, they need to be kept updated like anything else.

Popular examples of no-cost blogging sites are Google’s Blogger [http://blogger.com] and WordPress [http://wordpress.com]. Blogs can be used in the classroom to provide a quick feedback loop between creative writing and feedback, or to maintain discussions outside the classroom. Teachers can even maintain class blogs to keep students updated on homework and assignments.
Feeds

Feeds became popular around 2003 when they were primarily associated with blogs. By subscribing to an RSS (the syndication format) feed, you could keep up to date on headlines from a blog. Now feeds are being used to syndicate the format from all sorts of media, from pod- and vid-casts to social networks (e.g. Facebook, Twitter) and search agents. You can use a feed reader (a special application, either offline or embedded in a web page) to manage your feeds and monitor your favourite sites.

Wikis

By now, everyone’s familiar with the most popular and – in the educational world – notorious wiki, Wikipedia. While the online encyclopedia that’s edited by the community introduced us to the format of easily linked and marked up web pages, they’ve expanded to be used for a wide variety of uses. The primary focus of wiki is collaboration, and harnessing the so-called wisdom of crowds. A wiki is a type of free on-line writing space that allows users to add, modify and update its pages. If something is missing or incorrect in a wiki and permissions allow you to edit the wiki, you can easily add your thoughts or make changes to the wiki. It is essentially a fully editable web site.

Many classes use wikis to manage student or project collaboration. One of the important decisions to make is whether you’re creating a canonical reference that will be built upon by successive classes or whether you’ll create a new one for each class to chronicle their learning journey. There are many wiki alternatives for download, and it’s worth checking whether your institution hosts a specific one that you could test out.

Folksonomy

This term, coined by Thomas Vander Wal, refers to a new form of organic categorization that comes from internet users who encounter new information. Users add a keyword or descriptive phrase, and from then on, all users can find the
item using the same keyword. This spreads the categorization workload around to all the participants in the community, and creates a flat hierarchy that is more representative once a consensus model emerges based on the group interaction. On the other hand, misspellings can challenge the system with orphaned content, and lazy tagging can make matters worse.

**PodCasts**

Podcasting is very similar to a radio broadcast with the main differences being transmission via the Internet and producing pre-recorded content. Podcasts can be enhanced through the use of images, video as well as links to web site content. Broadcasting over the internet offers students and teachers access to a worldwide audience, with authentic feedback from around the world.

Usually, the multimedia files are uploaded to a blog via a podcasting service and posted to the internet. This content can then be syndicated through RSS feeds or via online services such as the iTunes Store and downloaded or streamed by the listener.

**Creative Commons**

The wide variety of work available on the internet can create a lot of confusion around ownership rights. It's hard to tell what's “public domain” and when the copyright holder should be contacted before use. The ease of downloading means that that is the simple route is often the default one. In the spirit of openness, the Creative Commons ([http://www.creativecommons.org/](http://www.creativecommons.org)) – a non-profit group devoted to expanding the range of available creative work – created some clear and easy to use licenses based on copyright. These licenses clearly the ways in which a work (be it an image, video, song, or writing) can be reused. By using a Creative Commons license, you can clarify how your work can be reused, whether freely remixed without attribution or only with proper attribution.
Summary

The Web 2.0 movement and the trends involved have served to make more tools freely available to teachers, researchers, and learners. These tools can help them collaborate with their peers, visualize information in new ways, or delve into niche communities. The biggest challenge can be keeping track of the most important tools and information. Web feeds help on this front, especially once you develop a list of canonical news sources that keep you up to date on subjects of interest. Hopefully this section has helped you understand what's going on with Web 2.0!
SECTION 7: Putting it all Together

This final section, *Digitization Fundamentals*, by Robin Davies and Michael Nixon, is taken from Routledge’s *Doing Digital Humanities* (2016), Edited by Constance Crompton, Richard J. Lane, and Ray Siemens.

**Introduction**

What constitutes a Digital Humanities (DH) project? First, there is material to be digitized, including various manuscripts, images, and audio/video recordings. This material may be fragile or rare and require special handling, and there may be medium-specific artifacts that need digitizing as well (e.g. marginalia, dates on the backs of photos). Some of these can be considered essential elements, while others are recorded as metadata, and stored in a subsidiary format. Next, digitized material will often be incorporated into a processing or analysis system. There are a wide variety of these, ranging from tools for text analysis to visualization to dynamic programmatic transformation, and they tend to allow what Moretti calls distant reading (Moretti). Finally, some combination of the initial plainly digitized materials and their new dynamic formats need to be published, typically on the Web.

Given this broad description of the DH project, it’s easy to see how the field continues to grow and draw attention as a means for scholarly engagement. Kathleen Fitzpatrick states: “For me [DH] has to do with the work that gets done at the crossroads of digital media and traditional humanistic study. [...] On the one hand, it’s bringing the tools and techniques of digital media to bear on traditional humanistic questions. But it’s also bringing humanistic modes of inquiry to bear on digital media” (Lopez, Rowland, and Fitzpatrick).
As we think about the nature of DH in relation to new analytical tools, it is also useful to draw attention to David Berry’s concept of three “waves”. He identifies the first wave as one occupied with acts of digitization and structuring material for classification and access (Berry). The second starts to draw on new methods of interpretation and interaction, especially for interrogating “born-digital” work. Finally, he suggests a third wave may lie ahead of us, bringing with it shifts in knowing, a “way of thinking about how medial changes produce epistemic changes” (Berry 4). This helps clarify the scope of DH projects and the developing nature of what constitutes a contribution to the field.

In this chapter, we will focus on several kinds of tools, leveraging foundational digitization practices for transferring humanities data into a digital medium, and modeling it for sustainable computational research. In particular, we will take a scenario-based approach to reveal important foundational considerations behind the choices researchers make as they work through a DH project involving the digitization of artifacts.

The three scenarios that we will be describing are: creating a digital edition, distributing oral histories, and coordinating access to music notation. First, we will look at how to provide access to scholarly, edited texts that have been digitized, with reference to the Internet Shakespeare Editions (ISE, http://internetshakespeare.uvic.ca/). Next, we will look at challenges digitizing audio files and related material, with reference to the Memorial University of Newfoundland Folklore and Language Archive (MUNFLA, http://www.mun.ca/folklore/munfla/). Lastly, we will examine a scenario where researchers have to innovate in their use of file formats for multimedia data. There we will look at the Single Interface for Music Score Searching and Analysis project (SIMSSA: http://simssa.ca).
We will use the lenses of *Goal*, *Process*, and *Sharing* to examine these projects. *Goal* allows us to focus on ensuring hidden priorities and constraints of technology are a good match to the practitioner’s motivations and intended contribution. *Process* takes us to the core practice of converting analog materials into a digital format, while reminding us of implicit trade-offs; it points at commonly adopted best practices. Finally, *Sharing* increases awareness of the complete digitization life cycle to further the goals of preservation and publication. This helps us consider how dissemination can best be accomplished.

**Creating a Digital Edition**

In this section, we will discuss some critical issues involved in digitizing text. Earlier, we noted Berry’s observation that the first wave of DH projects is primarily concerned with providing digital access to material, and one of the mainstays of this is the digital edition. Such editions clearly highlight the importance of considering your goal, however. Patrick Sahle provides a working definition of a scholarly edition as “the critical representation of historical documents” (Sahle), including both representation by data and by media. Notably, he also points out that if an edition is to be scholarly it must go beyond being simply a digital facsimile. The notion of a requisite different paradigm is useful, and helps connect to the values highlighted in Berry’s second wave; new modes of access and even “reading” are facilitated.

Thus, it is important to understand how you might leverage the existing infrastructure of digital tools for textual analysis, and what requirements such interoperability might impose on your digitization process. You will need to decide how your digitized materials can be used with students to facilitate teaching, or by researchers conducting various kinds of scholarly work. By looking around at existing tools, you could determine whether you’re reasonably able to produce digital material appropriate for use with them. While we all are likely to want to
support a broad range of uses, it’s also valuable to decide on a reasonable scope to avoid continually increasing the amount of work to be undertaken. So, if you are planning to create a digital scholarly edition, you will find yourself concerned with a range of possible technical operations and a need to make choices about all of them. Always reflect back on your goal to determine whether the addition of a particular element is actually desirable.

An example of such a possibility is an early consideration that you will need to make: whether to have the original text transcribed. Transcription is a laborious process that will have to be managed for quality control—as an example, the UK 1911 Census project had a goal of 98.5% accuracy due to the complexities of deciphering the many handwriting samples involved (“About Census”). However, transcription will directly produce ‘born-digital’ text for your use.

You might choose direct digital imaging using a scanner or digital camera. This could be the case when there are details unique to a particular manuscript that you wish to capture as images, such as those the US National Archives and Records Administration (NARA) considers to determine if an artifact merits preservation: aesthetic or artistic quality, unique physical features, details for establishing authenticity, and serving as an important exemplar of the work (National Archives and Records Administration). These digital images can be integrated with text to allow researchers to spot any details that would otherwise elude textual description.

Recommending digital imaging hardware is outside the scope of this chapter, and it is increasingly the case that contemporary equipment is broadly capable, so instead determine the kind and quality of images that can be produced, and compare them to relevant minimum standards. Additionally, don’t overlook the power of what can be accomplished with existing hardware in constrained scenarios, such as using a
digital camera (perhaps even a mobile device in certain cases!) to capture works incompatible with a flatbed scanner.

Essential is a high quality digital master image that represents all significant information from the source document. Creating this master image is important for the goals of preservation and access, and can prove economical in the long run. It aids preservation by serving—accompanied by appropriate documentation and metadata—as a surrogate sufficiently detailed to eliminate the need to view the original. This does imply careful archival storage of the digital image, allowing for retrieval, quality monitoring, and support for the foreseeable range of future applications. The master image takes on increased value to the degree that the one you’re making can be considered the canonical or most reliable archive version, although it is critical for your personal project regardless.

There are a few technical details relevant to the Process stage worth covering here. These include pixels and resolution, bit depth, and file formats. Digital images consist of a grid of small squares, known as picture elements (pixels). These are mapped to the display device. Resolution refers to the level of detail of an image: it is measured in pixels for computers and digital cameras; in pixels per inch (ppi) for scanners; in dots per inch (dpi) for printers. Bit depth refers to the number of available tones or colours for each pixel, with a higher number providing a larger palette. For example, 8-bit colour has a range of 256 colours since that is the maximum size of the integer that can be stored to represent it.

There have been many file formats created to manage picture information, but we will only mention the most relevant. RAW is a format you may find on your digital camera, and really stands for a variety of proprietary formats that contain all available image sensor data. While these files should generally be retained, images must be converted to a universally understood format before sharing. TIFF is
flexible and platform-independent with broad support by numerous image-processing applications. Master images should be archived in this format. The PNG format is a compressible, free open-source successor to the GIF format and supports true colour. Their compression does not discard information: it is lossless. PNG is recommended as a presentation format for images. Finally, the JPEG format is ideal for sharing photos online due to its excellent compressibility. However, it is *lossy*, and will irretrievably compromise information and image quality. JPEG files should be treated as essentially read-only and produced from higher-quality images for sharing on the web as an alternative to PNG files.

Following on this information, here is a general overview of best practices for digital conversion. Digital cameras should capture 24-bit colour depth (1 bit for black & white text) TIFF (or optionally RAW) files. Flatbed scanners should capture 24-bit colour (1-bit for flat text, 256-bit monochrome for shaded print), 600+ ppi TIFF files. These recommendations, along with more regarding slides and photographs, are contained in the online *Capture your Collections* guide (Canadian Heritage Information Network). Various other institutions (e.g. NARA, the Library of Congress) establish their own best practices (Cornell University Library/Research Department) which include variations such as 8-bit gray colour depth at 300 dpi for capturing images of text, and maximum resolutions such as a maximum width of 3000 pixels for photos.

Once you have captured the master image of your texts, you must now turn that digital image into text. The Optical Character Recognition (OCR) process takes digital images and attempts to recognize typographic details in them. While the free software that came with your scanner often suffices for capturing images, you may find it doesn’t provide OCR capability or lacks features like training to improve recognition accuracy on a corpus. We have had success on projects with software such as Adobe Acrobat Pro, Abbyy FineReader, and even the basic free-ocr.com.
Before performing OCR, ensure master images of text are high-contrast and not skewed; both properties can be managed to a degree in photo editing software, although ensuring good source images directly is best.

However, an OCR system can make only educated guesses about what printed materials are trying to represent. Rose Holley discusses this: “True accuracy, i.e., whether a character is actually correct, can only be determined by an independent arbiter, a human. This can be done by proofreading articles or pages, or by manually re-keying the entire article or page and comparing the output to the OCR output. These methods are very time consuming” (Holley). Therefore, one should ensure that one’s chosen OCR system is both accurate and capable of showing uncertain text in a reasonable way.

Once reliable digital text has been obtained, it can be put to a number of uses. The ISE follows a standard approach. The text was transferred to an XML file and marked up with appropriate elements based on the Text Encoding Initiative (TEI) standard. Then, HTML files are procedurally produced using the XSLT process on the original XML files (Siemens et al.).

Michael Best describes how the ISE take advantage of various Web technologies, by enabling users to choose a desired level of annotation, language, and ultimately how each work will be accessed (Best 31). Readers can read linearly or by searching text or annotation using automated tools; they can also read a variety of integrated essays, from performance histories to the editor’s own critical viewpoint. Best also explores possibilities such as using animated graphics to switch between alternate meanings of an archaic word, as in the case of the weyard/wayward/weird women in *Macbeth* (Best 33).
Distributing Oral Histories

The Memorial University of Newfoundland Folklore and Language Archive (MUNFLA) serves here as a point of departure for discussing the digitization part of the Process stage, using the audio medium as an example. “The archive comprises extensive collections of Newfoundland and Labrador folksongs and music […], folk narratives of many kinds, oral history, folk customs, beliefs and practices, childlore and descriptions of material culture” (MUNFLA). A large portion of this archive is analog audio media, which will need to be digitized to enable easy access. While this process can happen on demand, it is time-consuming; a well-tended, fully digital collection would be more portable and durable than fragile and decaying analog tape. Since MUNFLA’s inception, digitization has occurred in phases, funding permitting, "to preserve the materials and to make them more widely available to researchers". Digitizing "involves struggling with technical parameters (levels of resolution for both pictures and sound files that give as clear a sound or image as possible with as low a bit rate as possible)" (Inkpen).

An audio Analog to Digital Converter (ADC) is the hardware responsible for digitization, and the ADC deserves some attention here; it produces, at the sampling rate, a series of pulses that are shaped/modulated by the microphone’s electric signal (Figure 1). In a computer, the end result of the ADC process is typically Waveform Audio File format (WAVE) or Audio Interchange File Format (AIFF) content.
CD quality audio—still a benchmark, even as the recording industry leaves that format behind—uses a sampling rate of 44.1kHz. This means the audio waveform is sampled 44,100 times per second, permitting the signal to contain frequencies up to 20kHz, the theoretical upper-end of the range of human hearing. Readers interested in the origin of the 44.1kHz rate should consult Michael Unser’s discussion of the Nyquist–Shannon sampling theorem (Unser).

Using progressively lower sampling rates, to reduce the amount of data stored in the digital facsimile, will remove higher frequencies from the audio signal. Subjectively, these higher frequencies are often described as the bright or crisp parts of an audio signal, and their presence helps with intelligibility and clarity. Signals missing these high frequencies can sound dull or muffled, and it is harder to separate the different sound sources in the audio. In an effort to more faithfully represent the analog waveform, the recording industry often makes use of higher sampling rates—such as 48kHz, 96kHz, or 192kHz—the merits of which are fiercely debated. GRAMMY audio engineering winner Andrew Scheps, for example, acknowledges "diminishing
returns [...] building audio hardware” to deal with higher sampling rates (Scheps 15:53).

The bit depth determines the possible value of each sample. The CD uses 16-bit audio, meaning the sample can be set to one of $2^{16}$, or 65536 possible values. Bit depth is directly connected to the way the volume of a signal is measured. Part of the appeal of an analog system is that quantities change smoothly over a range of possible values; digital systems can be jarringly jagged. Consider an example from daily life: a light switch—digital in nature—provides only two states, on or off, whereas a rheostat—analog in nature—facilitates a gradual, conceivably unnoticeable, change in light level. Thus, reducing the bit depth of a digital signal brings what is already approximation closer and closer to dichotomy.

While it may be permissible for the ADC to omit elements of the analog signal when making a digital facsimile, the conversion process must not introduce digital artefacts that misrepresent the analog original. Two examples of this kind of misrepresentation are (1) aliasing, which happens when the sample rate is insufficient to capture high frequency detail in a signal (Figure 2), and the ADC makes an erroneous approximation, and (2) quantization distortion, which introduces errors in a digital signal resulting from simplifications made measuring the loudness of the audio signal (Figure 3).
To understand further the concept of aliasing, consider this visual example. If a wheel is spinning clockwise at a speed of one revolution every ten seconds, and a camera documenting the wheel's revolution shoots one frame every nine seconds, someone viewing the resulting sequence of frames would say the wheel is actually...
turning counter-clockwise. In this case, the camera’s samples provide an incorrect record of what is happening in the real world.

In photography, a camera records the amount of red, green, or blue light reaching a specific location on a sensing chip. As discussed earlier, if a camera uses eight bits to represent the amount of red light, there are only 256 possible values for red; the camera must then choose the closest colour match for each pixel. These choices, known as quantizations, introduce errors: distortions of reality. Therefore, when digitizing, variables such as sampling rate and bit depth should be set as high as is possible and necessary. The digital result is then faithful to the analog content, and contains detailed enough data to facilitate prescribed analyses.

The Moving Picture Experts Group 1 or 2, Layer III format (MP3), popularized by portable music players and made notorious by online file sharing disputes, is the de facto standard for digital music distribution. The biggest advantage of an MP3 over the archive WAVE is its relatively small size. MP3 files can be one-tenth or one-twelfth the size of their archive sources. A smaller file size means less data has to be stored and streamed online, and facilitates the lower bit rate desired by distributors such as MUNFLA. Most listeners will not perceive any difference between a WAVE and its MP3 counterpart, but clearly significant changes have been made to the data. The MP3’s COnpression and DECompression (codec) system, which encodes and decodes audio data to achieve this reduction, capitalizes on the characteristics of human hearing to achieve efficient lossy compression.

Jonathan Sterne has written extensively on the topic of perceptual encoding, and points to telephony research at the root of the MP3 format: "There was something to be gained by employing the ear, and specifically its perceptual limitations as outlined in the theory of masking and critical bands, as the basis for digital sound
reproduction in limited-bandwidth environments” (Sterne 112). Briefly, however, let us consider three ways a codec could eliminate some data.

First, in a stereo recording there are two separate channels of audio information meant for playback on a device with two speakers. However a great deal of what we hear from television and radio has exactly the same content on both channels. Consider a CD recording of an audiobook: it must have two channels, to conform to the CD specification, but the performer’s voice is actually recorded in mono, so exactly the same information is stored on both channels. One easy method then, to simplify in a lossy version, is to eliminate that redundant information, and simply re-duplicate it on playback. Another option for oral histories would be to make a blanket conversion of content to mono, assuming nothing important is conveyed by the signal being stereo. Some applications, such as Hollywood blockbuster surround sound, may require more than two channels, and even a simple oral history recording might contain useful ambient sound captured in stereo.

A second area of opportunity for producing lossy audio lies in masking. Audio from the real world fights for our ears’ attention. The sound of a passing ambulance’s siren will mask (drown out) a quiet conversation happening on the sidewalk. Although a microphone will notice small changes in sound pressure created by the latter sound, a lossy compressor can safely remove any record of those changes, as the human ear won’t notice.

Thirdly, our ears are more sensitive to some frequencies, which should be championed, and some frequencies do not need to be included at all, such as those outside the range of human hearing. A codec can remove very high and low frequency sounds, particularly if other material is masking those sounds. In these ways, the codec’s algorithms calculate whether a particular sound will be audible and relevant to the listener, or else it will be removed from the audio file.
Thus content captured into a WAVE file during acquisition is simplified for delivery. Distribution of content, in the Sharing stage, will likely see MUNFLA employing MP3 files, or those produced to a standard such as the Advanced Audio Codec (AAC) or Ogg Vorbis (OGG), which feature similar kinds of perceptual-based data economizations. These files will likely be distributed via a web-based system, easing the burden on archivists making CD copies of digital audio content for individuals.

Copyright is an additional Sharing consideration. The archive's website states: "All copies made by MUNFLA are for user research and private study. Permission for use of archival materials is for single use only" (MUNFLA). Making users aware of their rights can help to ensure compliance as a good will measure. Offering a license that provides more specificity about usage rights is even better, and the Creative Commons licensing system ("About Us") is an excellent example of a flexible method for enabling this process.

A good template for controlling access to content for research purposes is the website freesound.org, which exists to share user-generated audio content, although users must register a free account. The Freesound team is proud of its tracking capabilities: "All the data about downloads, ratings and comments is available with corresponding timestamps since 2005" (Font, Roma, and Serra 412). Similarly, in an effort to make users somewhat responsible for their use of the archive content, MUNFLA could automatically respond to each request with a custom download link that doesn’t expire and inherently maintains a record of any data access.

Finally, the content owner may also take steps to thwart digital piracy by either directly or indirectly prohibiting copying. A full discussion of Digital Rights/Restrictions Management (DRM)-enabled formats and associated issues is beyond the scope of this chapter, but archives such as MUNFLA may wish to take
steps to prevent unauthorized duplication. Consider the strategy of the Pragmatic Bookshelf publishing imprint (pragprog.com): ebooks purchased from the site are DRM-free, meaning the content can be freely copied and reformatted. However, these purchased PDFs sport a custom-watermarked footnote on every single page. This footnote contains the words "Prepared exclusively for", followed by the downloader's name, in an effort to avoid sharing between people, as opposed to copying for a single person's use in multiple contexts. A watermark of this nature may be an acceptable feature of text-based material online or off, and even video media. However, aural watermarking poses numerous challenges. Listeners do not want the audio quality to be affected by extraneous noises, tones, or advertisements, and a watermark could also hinder analysis via digital tools. One designer describes an algorithm's non-intrusive function: "With default parameters, the proposed watermarking algorithm demonstrates practically undistinguishable (sic) watermarking which is transparent to an average listener with audio equipment of any quality on majority of audio content" (Radzishevsky).

**Coordinating Access to Music Notation**

DH projects of any size, particularly when collaborative in nature, need to make decisions about file formats to ensure data's portability throughout the project lifecycle, and longevity once the work is complete. Choosing appropriate formats is essential early in the Process stage, for content acquisition, and later in the Sharing stage, for content delivery.

This section will examine the capture and storage of musical notation, specifically addressing translation of musical scores from physical objects through scanning, and the resulting data's representation in a digital file format. While choice of file format is a consideration in any project, printed music is chosen here as a medium with a less-established digitization practice.
In some cases, a well-known file format may be appropriate, such as a TIFF (acquisition) and JPEG (delivery) solution for photographs, or a WAVE (acquisition) and MP3 (delivery) solution for oral histories.

Other projects’ goals may require custom, less prevalent formats to make data accessible as intended. SIMSSA provides such an example: SIMSSA’s goal “is to teach computers to recognize the musical symbols in these images and assemble the data on a single website, making it a comprehensive search and analysis system for online musical scores” (Fujinaga, Hankinson, and Cumming 100). Though the researchers of SIMSSA "have chosen the MEI (Music Encoding Initiative) format" (102), a very recent (2013) standard, to hold musical data, they also note certain aspects of the music recognition and storage process will require "an avenue of research that has never before been explored" (102). Let us consider the data on which SIMSSA plans to operate.

Early composers and transcribers of musical performances considered the many facets of musical data - pitch, duration, dynamics, timbre - and determined which components of music were essential to record, in an effort to store the music, making it transportable and shareable. Thomas Forrest Kelly shows how frequently a storage format is a product of its time, developed to meet current goals, not necessarily future demands. "It’s not that the system is inefficient (although for some purposes it is); it’s just that it developed over time with specific music in mind. As music changed, the musical notation changed too, but not by completely reinventing itself. Instead certain aspects of the existing system were developed or given new significance as people wanted to record additional aspects of music” (Kelly 4).
In this way the format itself becomes a kludge; the original structure is stretched and repurposed to accommodate additional data. Modern digital formats for a variety of media exhibit the same kind of purpose-specific mutation. For example, the various versions of Hypertext Markup Language (HTML) document users’ desires to eschew standards and make a format capable of new, perhaps exciting things. Ben Henick writes: "The virtues of interoperability do not, however, harmonize easily with the hot desire for bells, whistles, and pretty things often felt by artists and marketers. [Web] Browser vendors cannot ignore the imperative to innovate, and the market usually works on a shorter life cycle than the standards acceptance process" (Henick 43).

Audio, too, has only some consistency; the manner in which music is stored in an MP3 file is very clearly defined by a number of standards. However, the completely unstandardized structure of meta ID3 tags—housing genre, artist, ratings, and other data— appended to MP3 files, shows how software such as Apple's iTunes can leverage and extend an existing, "informal standard" (Nilsson).

It is sensible, then, that creators and users of digital content would aim to streamline the storage of content for easy distribution. One entertainment software giant had this kind of forethought, creating the Interchange File Format (IFF): "By proposing a standard format structure, Electronic Arts wanted to avoid a situation where hordes of programmers would wantonly create whatever file formats were most expedient for their specific applications." (Wallace 24). Similarly, the history of computing is full of examples where a company created a format to address a current need and it gained enough popular usage to remain the de facto standard. The aforementioned AIFF is one example of how the IFF concept can produce files accessible to software from the past, present, and future (Morrison). On the consumer side of file formats, it’s important to understand how comprehensive and inclusive the design process
was, in order to choose a format that's most likely to stay relevant and usable for a long time.

Returning to SIMSSA, and its goal of a "single interface" for many users, it is clear why the project requires the interoperability of the MEI format. The MEI aims for longevity in addition to the affordances of digital exploration. "Because of its emphasis on comprehensibility and software independence, the data format defined by the schema may also serve an archival function" ("Music Encoding Initiative").

Fundamentally, MEI stores musical information as text, which separates the data from its presentation. MEI uses XML as a foundation, leveraging its combination of elements and attributes. Thus, it doesn't matter what form of written notation was employed to originally store a note on paper; the digital translation is something like this:

\[<\text{note }\text{pname}="c" \text{oct}="4" \text{dur}="4" />\]

In the above example, a quarter note C in the fourth octave is created.

Needless to say, it is hard to imagine performing music from this markup kind of score. However outdated and inefficient conventional music notation is, Kelly reminds us we "still use their system, and it allows us to record sound for the future even today, a millennium later" (Kelly 4). Though Middle-Age era notation systems have been able to stand the test of time, they are clearly not easily deciphered by modern computers at the centre of big data initiatives.

When scanning text documents, OCR converts a sequence of glyphs to a digital representation. Printed musical information carries with it the same kinds of visual challenges as text in this regard: ink and paper quality will often affect the interpretation, and the software will have a harder time dealing with the
inconsistent marks made by hand as opposed to press. Holley suggests a number of best practices to deal with these challenges; for example: "Obtain best source possible (marked, mouldy, faded source, characters not in sharp focus or skewed on page negatively affects identification of characters)" (Holley).

But music, though it may contain text, has a much wider variety of symbols, and potential arrangements of those symbols. Optical Music Recognition (OMR) software thus extends greatly the OCR groundwork. "Since existing OMR software is less reliable and less comprehensive than OCR software, our first challenge is to develop reliable OMR software that can process many forms of music notation" (Fujinaga, Hankinson, and Cumming 100).

An important aid in this challenge of training machines lies in the possibility of collaborative computing, whereby users knowledgeable about content take part in the digitization process, most notably verification of recognition systems' accuracy. Adopting this strategy, users of SIMSSA "will correct the OMR for music sources they care about" (100), which reminds us a data set is only as comprehensive as interested parties' investments.

Though it hasn't yet been created, a tool accessible via the Web will be part of SIMSSA’s Sharing stage, allowing users to interact with existing data and provide data of their own. In addition, users could choose to render MEI documents from the SIMSSA archive in other formats. For example, the Portable Document Format (PDF)—introduced in 1993 by Adobe Systems Incorporated—would be sensible for users wanting to choose a musical font and generate conventional notation to view on screen or in print. If computers or other hardware are used for automated performance of MEI content, a Musical Instrument Digital Interface (MIDI) file—introduced by a collective of industry forces in 1983 (Chadabe)—could be
employed. Researchers wanting to do custom analysis of musical data could develop tools to manipulate the MEI content directly.

**Conclusion**

In this chapter, we proposed a three-stage model for thinking about projects: Goal, Process, and Sharing. Thinking and working iteratively through these allows consideration of important Goals and how they will inform other stages of the project. For example, Sharing images via website implies that multiple kinds of files will be created during the Process stage: archive-quality files (acquisition), plus lower-resolution files (delivery). We also explored the idea that DH encompasses a wide range of work, and showed how contemporary DH projects span the three waves envisioned by David Berry, by looking at issues involved in creating a digital edition, distributing oral histories, and coordinating access to music notation.
Works Cited


