



STANFORD UNIVERSITY PRESS

Archivability

Stanford University Press shares with you the belief that digital academic arguments deserve treatment in the scholarly record equal to that of traditional print media, and we have a serious commitment to ensuring the longevity of your work through responsible approaches to sustainability and archiving. Your conscientious authoring and design choices are the first step toward sustaining the long-term fidelity of your work. But because we acknowledge that even the most sustainably designed web interfaces are susceptible to normal decay, we must anticipate the necessity of preserving your project's content so that readers and researchers can access it even after changing web environments begin to disrupt the project's dynamic and interactive features. To that end, we intend to take multiple approaches to archiving your content and preserving documentation of its intended presentation.

Unlike a typical monograph publication, for which the publisher implements the material architecture that contains the content, you, the author, hold the creative responsibility for your project's format, design, and execution. Nevertheless, as your publisher, we share the responsibility of the sustainability of your work and therefore recommend, encourage, and require certain technology and design standards. We have compiled these archivability guidelines, along with the enclosed style sheets to help you make informed decisions on the types of technological considerations that will impact your project's longevity, and its amenability to current web archiving standards and practices.

In addition to your use of these recommendations, it is important you complete the technical questionnaire so we may begin developing the best publishing plan for your project, including its most suitable archiving environment.

Archivability Spectrum

A digital project falls into a spectrum whose endpoints are usefully analogized as object and performance. Whereas a book can be considered an object, with its performative qualities limited to the audience's experience while reading the book, an interactive digital project shares certain qualities of ephemerality that characterize a performance. It only functions as an object insofar as the device displaying the project or facilitating the reader's experience with the material remains intact. Because technology changes so rapidly, these devices and standards are always in flux. Thus, depending on the complexity of the tools and applications needed to deliver the project in a reader's web browser, the project may be more or less susceptible to decay as technology evolves and renders certain tools and applications obsolete or as external data or media change locations. To help you determine where on the spectrum your project falls, and what further safeguards will most likely be required because of its location on the spectrum, we have divided the possible archivability states into three broad and non-exclusive categories: 1) archive ready, 2) archive amenable, and 3) archive resistant.

Type 1: Archive Ready

This type of project is extremely rare. Even though it functions as a performance, it is more analogous to an object than most digital projects. The look and feel, structure, and functionality can be preserved and should not be susceptible to decay as technologies evolve, or they can be reproduced in future environments with minimal resources and no loss of fidelity. Such a project uses standard file-and-folder systems and includes a sitemap, xml or otherwise, that provides fixed urls for each page and state. Each state is a static HTML page, and all data and media presented on those pages are hosted with the project and not queried from third-party sources. The project does not link to external websites or content other than that stored along with the project itself. This kind of project does not use proprietary software or applications, which would require updating and/or patching. There are few or no dynamic scripts that will glitch or fail when the coding standards or libraries change. In addition to satisfying all the style recommendations in the accompanying documents, it adheres to web archivability and accessible standards as published by the World Wide Web Consortium, Library of Congress, Stanford University Libraries, and the Web Accessibility Initiative:

- World Wide Web Consortium List of Standards: <https://www.w3.org/TR/>
- Library of Congress's Recommended Formats Statement: <https://www.loc.gov/preservation/resources/rfs/websites.html>
- Stanford's Archivability Standards: <http://library.stanford.edu/projects/web-archiving/archivability>
- Web Content Accessibility Guidelines: <https://www.w3.org/standards/webdesign/accessibility>

Additional Requirements:

Because this kind of project is made up of pages that can be automatically crawled and recorded as well as a file system containing all HTML pages and media objects, with no reliance on database protocols or third-party queries, few, if any, additional materials are necessary to aid in the archiving process. Nevertheless, as with all projects Stanford University Press publishes, documentation of the composition process as well as description (written as well as video-recorded and/or screen-captured) of its functionality and featured contents is encouraged. (See enclosed "Documentation" guide.)

Type 2: Archive Amenable

While the original functionality of this type of project will last longer than one that is archive resistant, certain features can be expected to break sooner than those of an archive-ready project. An archive-amenable project can still be archived, but the preparation stages will take more time and could require the author to provide alternative formats or paratext. The maintenance and preservation of the project's archive also requires more work and frequent attention. Additionally, the archived version will likely act and/or look differently than the initially released publication. This type of project might unnecessarily employ a database structure (Wordpress, Drupal, Scalar, etc.) that could impede the efficiency of navigating the site as well as put undue pressure on the server, slowing down the project's interactive elements. Consideration should be given to whether the user experience requires the framework offered by the database, or if that database functioned primarily for the authoring process and is not needed to deliver the published content. Oftentimes, for example, a site built in Wordpress does not actually need the scaffolding of that platform on the reader's end. In such cases it might be advantageous to scrape the content and restructure it using HTML(5) and file-and-folder systems, an architecture which puts far less strain on a server and requires fewer updates to security protocols. Other factors that characterize an archive-amenable, rather than archive-ready, project are javascript

that will require updates, web fonts that need to be downloaded each time a page loads, external links to websites that might disappear or change location, etc.

Additional Requirements:

When we perform the technical review of your project, we will compile a formal list of further requirements to aid in the pre-publication archiving process. Although it's difficult to anticipate the form certain archive-amenable projects will take, we can at the very least suggest that you be prepared to either edit platform and code base or to build in the time it could take for SUP to perform such actions in the production phase. You should also strive to collect cached datasets from any third-party sources your project queries and gather and save all media elements that are currently hosted separately from your project (e.g. Vimeo, Soundcloud, Imgur objects, etc.)

Type 3: Archive Resistant

Because most authors utilize dynamic authoring tools and cutting-edge technologies, a majority of projects fall under this category. Like a performance, it is possible to record an instance of this kind of project, but preserving the original experience beyond five years is a significant challenge with current archiving technologies. This type of project is likely designed using custom code that might not be accessible to someone who may be required to update it in the future for it to work properly in evolving web environments. While the technologies needed to render the project are current now, they will certainly change in the near future. The best approach to preserving such projects is emulation, a still inconsistent and imperfect system that requires stacks of multiple technologies such as specific browser versions, operating systems, and code libraries, all of which must work in conjunction with any third-party data that is likely to become unavailable or incompatible with the old environments being emulated. Until a stable, reliable solution is implemented that can recall today's technologies in the future, we must at the very least safeguard against the loss of content by creating an archived version of the project that necessarily omits the more experimental and less stable features. Such features include script-based animations and visualizations, network and database visualizations, and embedded APIs that query third-party sources. Also problematic are links to external websites that will very likely become broken as content moves or is renamed and as security protocols update the urls used to locate that web content. While we don't restrict your use of these tools and formats, we cannot recommend them, especially when they are the only method of delivering content, and we cannot guarantee their long-term (or even short-term, in some cases) fidelity beyond the initial release of your work. Likewise, if you choose to employ such technologies, you acknowledge their ephemerality and understand that though your content will be archived, the fidelity of the initial user experience will likely be suspended until more efficient and sustainable emulation technology is developed. Although this type of project is dynamic and employs design features that may be creative, impressive, and engaging at the time of publication, these same features will most likely also render the site shorter-lived in its intended form, and even upon its initial release slow or inoperable on older machines.

Additional Requirements:

In order to convey to future audiences the project's original functionality, it will need to be documented and supplemented with paratext, such as video presentation or screencast by the author or SUP that thoroughly describes the project's purpose, scope, functionality, and key features. Such a project will also likely require any range of the following: a thorough written description of the project's development and the principles that guided the technology choices, a visual blueprint of the project's organization and structure, if feasible a complete static HTML version of the site that contains placeholders (e.g. jpg versions of interactive visualizations) for any moving parts that are expected to

degrade over time, and access information for a public or shared repository of any custom-built programs or applications.

Resources

Design:

For more information about sustainable web design, we recommend you read these more detailed resources:

- Library of Congress's Recommendation Formats Statement: <https://www.loc.gov/preservation/resources/rfs/websites.html>
- Stanford University Libraries' Archivability Recommendations: <http://library.stanford.edu/projects/web-archiving/archivability>
- An article on the advantages of static websites over database-driven sites: <https://www.smashingmagazine.com/2015/11/modern-static-website-generators-next-big-thing/>

Delivery and Access:

To test your project's code and accessibility compliance, use the following tools:

- Web Accessibility Evaluation Tool (WAVE): <http://wave.webaim.org/>
- W3C's Markup Validation Service: <http://validator.w3.org/>
- W3C's CSS Validation Service: <http://validator.w3.org/>

Web-Archiving:

For more information on web-archive format and emulation solutions, read the following:

- Library of Congress's Description of Web ARChive File Format: <https://www.loc.gov/preservation/digital/formats/fdd/fdd000236.shtml>
- Bibliothèque nationale de France's Description of WARC format: <http://bibnum.bnf.fr/WARC/>
- BWFLA's Description of Emulation: <http://bw-fla.uni-freiburg.de/>

WARC in Action:

Use these tools to glean the functionality of your project as it would be rendered in WARC format. Note that a web-archived version of the project is not the same as its initial fully-operative published version as the crawlers that capture the pages do not have access to the scripts behind those pages. This is why conscientious design, and design that avoids third-party querying, is so important.

- Web Recorder: <http://webrecorder.io>
(First, record the url and browse the site for a while. When finished, play it back and note what works and what doesn't.)
- Wayback Machine: <https://archive.org/web/>
(If your project currently has a public url and has been crawled by Internet Archive, see what it looks like as a crawled website.)