Software Selection Process and Criteria

Developed by the University Center for Social and Urban Research, University of Pittsburgh, 2009

Evaluating a wide range of emerging options requires the development of software selection criteria to ensure that products are the best fit for the varied needs of the users and are in alignment with a sustainable business model. This document borrows from a host of resources that outline model software selection processes and selection criteria. In general, the selection process includes the following steps. This document was adapted from a 2009 version prepared for the Pittsburgh Neighborhood and Community Information System.

Decision-making process for selecting software
1. Assess user needs
2. Prioritize Selection Criteria
3. Create list of vendors/options
4. Request Information
5. Evaluate responses
6. Create short list of vendors/products
7. Make final decision

The remainder of this document only focuses on selection criteria (#2 in the above list) that could be employed in the selection process.

Selection Criteria
The following selection criteria can serve as a starting point for any software assessment. Criteria are organized into a set of six broad categories.

Capital (startup) costs
- How much does the product cost?
- How easy is it to install the product?
- What are costs for any additional hardware, software, or server space needed to operate the product?
- What degree of installation support is available from the vendor, and is there any additional charge for this support?
- Are there any training or consulting costs in helping project staff become familiar with the system?
- What is the learning curve for project staff in adapting to the new system (length of time)?
- What changes are needed to make the standard product fit the needs of the project/users (fit-for-purpose)?
- What cost would be incurred to customize the standard product?
- Are programmers widely available with specialized knowledge or familiarity with the product?
- Is source code provided so customizations or modifications can be made without large costs?
- Do customizations hamper upgrading to future software releases?
- How many users can be served by the software, and what is the cost for scaling the project up to meet increased demand?

Operating costs
- What level of maintenance is needed to operate the system in an ongoing basis?
- What are the annual maintenance charges for the software, and are they reasonable?
- What are the costs in adding or updating data used by the software?
- Can our server administrators work with the product?

Functions/features
- How well does it meet the needs of the user community?
- How well does it meet the needs of the project team?
- Does it accept and use common file formats (or is a conversion necessary)?
  - Are file formats proprietary?
  - How easily is data importable/exportable?
  - How well does it work with existing software currently in use?
- How difficult is it to administer security and permissions?
- Is support for metadata offered?
- Is there an administrative module available allowing for easy site management?
- Is data secure?
- Does the software allow compliance with terms of data sharing agreements?
  - How/where is data stored?
  - Who hosts the data, and where are servers located?
  - Who owns the software?

Reliability
- Is the technology mature, or has the technology been proven to meet the needs of similar clients?
- What major changes/upgrades are planned?
- How long has the product been in existence, and how many versions of the software have been released?

Usability
- Are help features/documentation offered to the users through the software?
- What degree of training is required for end users (how intuitive is the software)?

Vendor
- Who else is using the technology (how large is the user community - especially important with open source)?
- How stable is the vendor?
  - Length of time in business?
  - Company size/revenue?
  - Number of customers (local and national), including recent change?
- Does the vendor have a proven track record?
- Does the vendor have the ability to provide support to users?
  - What types of support are offered?
- Is the company providing the technology likely to be around for the long-haul?
  - What are the implications if it is not?