



# How to Calculate Total Lifetime Costs of Enterprise Software Solutions

## Principle(s) Addressed

Design for Scale, Build for Sustainability, Be Collaborative, Use Open Standards, Open Data, Open Source & Open Innovation



## Overview

To ensure the success and sustainability of your initiative or program, weigh all potential considerations in selecting software, calculating and planning for technology costs beyond the initial deployment.

## Description

Digital development practitioners strive to select and implement sustainable, robust and cost-effective software solutions. One of the challenges faced in accomplishing this goal is that the total lifetime costs of implementation are often not accounted for accurately in a program budget. Technical experts who know how to identify and quantify ongoing costs related to usage, hosting and support may not be consulted during proposal writing. Inaccurate estimates can put a burden on program budgets or lead to choosing software that has prohibitive costs to scale or sustain [<http://digitalprinciples.org/design-for-scale/>] [<http://digitalprinciples.org/build-for-sustainability/>]. Further, digital development programs often aim to introduce tools and systems that can be transferred to a government partner for ongoing implementation and maintenance, so not accurately estimating costs transfers the long-term burden to the country, potentially leaving it with a solution that is ultimately abandoned because it is not affordable.

To evaluate the lifetime costs of a software accurately, the total cost of ownership (TCO) needs to be calculated. This calculation takes into account the costs of the software itself, as well as the necessary hardware, hosting and support. Understanding all the costs that will be incurred over the lifetime of a system can be tricky, as it is easy to overlook key considerations and costs that come later in the project lifecycle. It is useful to calculate TCO during the analysis and planning phase, as the resulting figure is one factor

“It is helpful to focus on the issues underlying the choice of a particular proprietary or open source solution, such as life cycle cost, availability of a robust support model, level of system interoperability, ease of ownership transition, ease of replacement and the effect on the local economy. Decisions should take full life cycle cost considerations, as well as support and maintenance implications, into account and consider impacts on local businesses and livelihoods.”

CAROL BOTHWELL  
CRS



# How to Calculate Total Lifetime Costs of Enterprise Software Solutions



used to compare and select software options. (For a discussion of other software-selection factors, see [How to Choose a Mobile Data Collection Platform \[http://digitalprinciples.org/be-data-driven/\]](http://digitalprinciples.org/be-data-driven/).) TCO also gives an accurate estimate of your program's budget needs to share with funders. If you are already implementing a software solution, calculating TCO can help you with ongoing budget planning and with communicating costs to partners who take over maintenance of the system.

TCO is only one of many considerations when selecting software. Other considerations include important contextual factors such as functionality, ease of use, local infrastructure, existing systems and the presence of a community of developers or consultants who can maintain the software and support users. For example, if there is no knowledge of the software locally, extra resources will be required to develop local expertise to maintain the system.

This guide outlines steps to help you calculate the TCO for a software solution. These recommendations are summarized from existing references, which can be found in the Resources section.

## Process

- 1. Understand the ecosystem.** Before estimating costs and selecting a software, it is important to understand the context in which you will implement. Even if you prefer software that you have previously selected, you should still reconsider the costs, as they may have changed over time or may be different in a different community. For example, in a country with a less competitive mobile market, you may find that costs for mobile data or SMS messaging are more expensive than in a country with a very competitive market. The costs of developers or other technologists to support maintenance of the system may also vary considerably. You may experience costs savings if a tool has already been implemented in the country, because experienced local developers and consultants have already been trained and are ready to provide support.
- 2. Understand cost drivers.** Pinpoint the types of costs you will incur to develop, initially deploy, scale and sustain the software you are implementing or considering [<http://digitalprinciples.org/design-for-scale/>] [<http://digitalprinciples.org/build-for-sustainability/>].

TCO is only one of many considerations when selecting software. Other considerations include important contextual factors such as functionality, ease of use, local infrastructure, existing systems and the presence of a community of developers or consultants who can maintain the software and support users.

# How to Calculate Total Lifetime Costs of Enterprise Software Solutions



FIGURE 1

CATEGORY	COST DRIVER	CONSIDERATIONS
<b>Management and Staffing</b>	Salary and travel expenses	What is the level of effort for program management staff associated with training, vendor relationship management and other meetings? Does staff capacity already exist on your team, or will you need to hire and train new resources to handle the program? Is there an opportunity to build capacity with existing staff at a lower cost than hiring new staff?
<b>Development and Setup</b>	Software licensing cost per environment and per user	What is the licensing model? For example, is it open source or proprietary? What are the licensing costs, and how will these change with scale? Is there a flat fee per number of users or individual fees per user or device? Is there a platform fee or costs to add users?
	Software customization, including adding additional languages	If you are working with software vendor, what are the costs to add features now or in the future? If the software is open source, is there a responsive, established user community that will provide support and help add features at no cost? Do you have skilled, available technical staff who can customize the software? What is the level of effort for technical staff to customize the software? What are the costs to contract with a consultant who is skilled and familiar with the software code to do the customizations? What are the costs to translate terms and develop the software in additional languages, if needed?
	Software installation and configuration	What is the level of effort for staff to install and configure the software? If you are replacing an existing system, consider the time needed to uninstall a previous system and to transfer data from the old system to the new system.
	Interoperability with other systems	What is the cost to interoperate with existing tools and systems? What efforts will be needed to ensure that the system complies with relevant standards, including open standards?
	Hardware	Do you need servers? Do users need devices?
<b>Deployment</b>	Cost and availability of data connectivity and power	What is the cost for the internet bandwidth or mobile data needed for the software to operate properly? Will you need to equip your office with generators to ensure that the system remains running during power outages? Do you need solar chargers, car chargers or spare batteries for reliable device charging?
	Training	Is there a fee for initial training? Are there travel and other logistical costs associated with training? Do you need to create new training and capacity building materials? How long are the trainings? How many people need to be trained? How frequently will you offer training to new users as the tool scales?
<b>Operations</b>	Voice and data services (mobile data plan, internet, number of text messages)	How many text messages and voice minutes will be used? How much mobile data will be needed for each user? Can you negotiate a below-market rate with a mobile network operator?
	Hardware maintenance, ongoing administration, and replacement rate of hardware	How often will you replace hardware (typical replacement rate is approximately 20 to 25 percent per year)? What are typical maintenance costs? How many staff members are needed for ongoing administration of hardware? What are their costs related to travel to project sites?
	Subscriptions	Is there a subscription fee? Are there costs to receive software updates or to access specific features?
	Software maintenance (fixing bugs, adding features, maintaining customizations)	Will you need to pay new license fees when you update to new software versions? Will volunteers from the open source community be able to do maintenance, or will you have to hire a developer? Consider that some updates may require additional development and testing. Will you get support from a vendor or program staff? Consider the budget implications of operations support for system crashes or to address software performance issues.
	Transfer of ownership	How much staff time will be needed to transition ownership to the government or another entity? What capacity building will this require? Will licensing costs change due to an increase in number of users? Will the new owner need to procure new hardware?
	Refresher training and additional training activities	What is the staff attrition rate? How frequently will you provide refresher training? What other training activities and materials will you offer?

# How to Calculate Total Lifetime Costs of Enterprise Software Solutions



TCO is typically calculated for three to five years, the average project lifecycle, but this timespan could vary based on your tool or context. Work with your team and potential software vendors to identify the relevant costs. Consult with other digital practitioners to get advice on other costs to consider [<http://digitalprinciples.org/be-collaborative/>].

The illustrative list of cost drivers in Figure 1 is adapted from existing TCO guides.<sup>1,2</sup> Not all costs will be relevant for every project or tool. For example, open source and proprietary software will have different licensing costs.

## CREATE BUDGET MATRICES TIPS AND RESOURCES

**TIP:** Consider how costs will vary as your initiative moves from pilot to scale to sustainability. Important variables include the cost of adding and supporting new users, in addition to preparation and training for transferring system maintenance to local partners.

<sup>1</sup>World Health Organization, PATH. *Planning an Information Systems Project: A Toolkit for Public Health Managers*. Geneva, Seattle: WHO, PATH; 2013. Available at [https://www.path.org/publications/files/TS\\_opt\\_ict\\_toolkit.pdf](https://www.path.org/publications/files/TS_opt_ict_toolkit.pdf).

<sup>2</sup>Cotton, B. "Free As In Puppy: The Hidden Costs Of Free Software" [blog post]. Opensource.com. 2017 Feb 13. Available at <https://opensource.com/article/17/2/hidden-costs-free-software>.

**3. Create budget matrices to summarize or compare costs.** Use your identified costs to create a detailed budget for each software solution, and then summarize the costs in a budget matrix (see Figure 2). If you are comparing costs across software solutions, use the comparison chart (see Figure 3). Online tools are also available to summarize your costs and compare TCO of different software, such as the following:

- Total Cost of Ownership Model developed by Dimagi to help with budgeting for mobile tools [<http://sites.dimagi.com/totalcostownership>].
- Costing Utility Analysis tool created by NetHope for calculating costs related to electronic payment tools (such as mobile money) and comparing them to costs associated with cash-based payment systems [<http://solutionscenter.nethope.org/products/view/653>]

**FIGURE 2: SUMMARY BUDGET MATRIX FOR AN INDIVIDUAL SOFTWARE APPLICATION**

BUDGETING CATEGORY	YEAR 0 (PILOT)	YEAR 1	YEAR 2	YEAR 3	YEAR 4	FIVE-YEAR TOTAL
Management and staffing						
Development and setup						
Deployment						
Operations						
Total						

**FIGURE 3: COMPARISON CHART FOR SOFTWARE**

BUDGETING CATEGORY	SOFTWARE 1: FIVE-YEAR TOTAL	SOFTWARE 2: FIVE-YEAR TOTAL	SOFTWARE 3: FIVE-YEAR TOTAL
Management and staffing			
Development and setup			
Deployment			
Operations			
Total			

**4. Use your TCO for planning, decision making and learning.** TCO is one selection factor that you can consider when deciding between several software options. You may also want to develop comparison charts that rank other factors important to your initiative and context, such as whether a local user and support community exists, whether the software adheres to open standards, whether it is available in the local language and so on. While cost alone should not be seen as the definitive deciding factor, TCO is important for planning and for determining if a tool will fit in your budget. Once you have established your TCO, share it with your funder or other stakeholders, so they can understand the cost drivers and the true cost of implementation. Some funders may be interested in analysis, such as the total cost per user, which can be calculated once you have estimated the TCO.

During the course of the initiative, monitor your budget to determine if your TCO estimates were accurate. If you find that certain costs were significantly higher or lower than estimated or if you encounter new costs, recalculate your TCO and realign your budget as appropriate.

In your evaluation phase, compare your estimated TCO with the actual costs. Look at where you over- or underestimated, and see what lessons you can learn for future TCO calculations [<http://digitalprinciples.org/be-data-driven/>]. Consider sharing your TCO process, the costs you identified and any mistakes you made in estimation with other digital development practitioners [<http://digitalprinciples.org/be-collaborative/>].

# How to Calculate Total Lifetime Costs of Enterprise Software Solutions



## Outcomes

These outcomes are illustrative and have been collected from digital development organizations that have followed the steps outlined in this guide.


- Better understanding of lifetime software costs, allowing for more accurate budget estimates.
- Ability to communicate true funding needs and show due diligence in software selection to donors, partners, host-country governments or other stakeholders.
- Deployment of software that meets local needs and can be scaled and sustained.

## Common Missteps

- **Identifying only the short-term costs.** Make sure that you estimate the costs for the entire life of the initiative, not just for the initial deployment and pilot phases. Plan for long-term costs, such as staff turnover, refresher training, support services and software updates.
- **Not engaging individuals with the needed technical knowledge during budgeting.** During your budgeting process, which often occurs while developing a funding proposal, make sure that you engage someone with the expertise needed to develop an accurate and thorough technology budget. If you do not have that expertise on your staff, seek out a consultant with that expertise.
- **Misunderstanding the technical needs of the initiative.** Take the time to design with the user to avoid the risk of choosing a tool without “must-have” features [<http://digitalprinciples.org/design-with-the-user/>]. Make sure that you account for the time and budget needed to carry out sufficient user research and to continue to solicit user feedback throughout the project lifecycle.
- **Not considering costs unique to the ecosystem.** A previous implementation of a tool in another context will help you understand some costs but not the full picture. Create your TCO with a strong understanding of the current ecosystem and the new, increased or reduced costs that you will experience in this specific context.

## RESOURCES

- Budgeting for a Project: CommCare Total Cost of Ownership (TCO) Model, Dimagi. <https://confluence.dimagi.com/display/commcarepublic/Budgeting+for+a+Project>
- Costing Utility Analysis tool, NetHope Solutions Center. <http://solutionscenter.nethope.org/products/view/653>
- Free as in Puppy: The Hidden Costs of Free Software, Opensource.com. <https://opensource.com/article/17/2/hidden-costs-free-software>
- Planning an Information Systems Project: A Toolkit for Public Health Managers, PATH and World Health Organization. [https://www.path.org/publications/files/TS\\_opt\\_ict\\_toolkit.pdf](https://www.path.org/publications/files/TS_opt_ict_toolkit.pdf)



# How to Calculate Total Lifetime Costs of Enterprise Software Solutions



- **Failing to manage and control costs.** Proper planning, training and oversight can help cost estimates remain accurate. Plan for ways to ensure that the software maintenance, hardware replacement and training costs remain within budget, which could include providing hardware to staff with protective cases and communicating a clear usage and damage policy.

