



Guide and Instructions: Total Cost of Ownership Calculator for Child Welfare Information Systems

Introduction

Child welfare information systems are critical tools to help support management and tracking of child welfare programs and effective delivery of services to children and families. The goals for a child welfare information system should be to improve processes, boost operational efficiency, achieve end-user satisfaction, and provide child welfare staff with tools that support the services provided to children and families. In addition to those goals, data collection and tracking are needed to inform practice, aggregate data for Federal reporting, and support measurement of the effectiveness of child welfare programs. As States/Tribes/territories contemplate how their data and operational needs can be met by a child welfare information system and consider options available in the market, it is important to calculate and consider the total cost of ownership of such a system.

Total cost of ownership (TCO) is a comprehensive assessment and financial estimate of the expenses (direct and indirect costs) associated with purchasing, deploying, using, and retiring a product, an information system, or a piece of equipment. A TCO analysis can gauge the purchase price of an asset—in this case, a child welfare information system—and it can also gauge ongoing operational costs over a multi-year period. The TCO includes hardware and software acquisition, management and support, communications, end-user expenses, training and the cost of moving data from the old system to the new system. Additional detail on the types of costs that should be considered in the TCO is provided on subsequent pages.

A TCO should be performed at the start of a software planning process and should be performed by individuals who are familiar with the various costs associated with the information system. A strong TCO plan, like any other planning document, results from ongoing partnership between IT staff, program leadership, and other stakeholders. Collaboration early in the process can help the agency identify the type of child welfare information system that best meets the unique business needs of the jurisdiction. The completed TCO, or portions of it, can be shared with and used by IT managers, budget and finance managers, program managers, funders and agency leadership teams.

The following section describes commonly used approaches to system development, key phases of system development, provides an overview of information technology options available and describes differences in cloud vs. on premises hosting. They are included in this TCO guidance because choices that a jurisdiction makes on development approach, selected solution and hosting option will drive system costs and each phase of the system development lifecycle has costs that should be included in the TCO.

Agile vs. Waterfall Development Approaches

Two of the most common methods of software development are Waterfall development and Agile development.

Waterfall is a traditional software development methodology whereby development is done in a sequence with one phase being completed before another phase begins. At a high level, Waterfall

projects allocate extensive periods of time for analysis/requirements gathering, design, build, testing and user acceptance testing (UAT), before deploying the system. Given the sequential nature of the Waterfall methodology, upfront project planning, thorough project documentation, and ongoing project management are all crucial to meet development and deployment timelines. Using the Waterfall approach, a project can take several months or years to complete, and the client may not see the end product until the project is complete. The Waterfall approach works best when clients can articulate a clear vision of what the final product/information system should be.

In contrast to Waterfall, **Agile** projects have sprints or iterations, which are shorter in duration (sprints/iterations can vary from 2 weeks to 2 months) during which predetermined features are developed and delivered. Unlike Waterfall, Agile projects can accommodate changes/rewrites to the design after the initial analysis and requirements phase, allowing for updates to be accommodated prior to project completion. Upfront project planning and ongoing project management are all crucial to the success of an Agile development. Agile projects can have one or more iterations and deliver the complete product at the end of the final iteration. The Agile approach works best when there is not a clear vision of what the final product/information system should be and when skilled developers who can adapt to changes quickly are available to work on the project.

A. High-Level Overview of the System Development Life Cycle

System Development Lifecycle (SDLC)—A logical process used to develop an information system, including requirements definition, validation, training, and user ownership (source: webster-dictionary.org). The SDLC has different phases that track stages of system development. The phases are defined by the following activities:

1. **Analyze:** During this phase, project managers, stakeholders, and subject matter experts gather information about relevant business requirements. Meetings with managers, stakeholders, and users are held in order to determine business requirements such as: Who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? After requirement gathering is complete, the requirements are analyzed for their validity and the possibility/feasibility of incorporating them in the system to be developed is also studied. Finally, a requirement specification document is created that serves as the guide for the next phase.
2. **Design:** In this phase, the system and software design is prepared from the requirement specifications that were analyzed in the first phase. System design helps to specify hardware and system requirements and define overall system architecture. The system design specifications (as documented) serve as input for the next phase of development (Build/Code). During the Design phase, identified testers develop a test strategy which defines what to test and how to test the system.
3. **Build/Code:** Upon receipt of system design documents, the work to build the system is often divided into modules/units and actual coding is started. This is the longest phase of the software development life cycle.
4. **Test:** After the Build phase is complete, developed system code is tested against the requirements to make sure that the system/product is actually solving the needs addressed and gathered during the Analyze (requirements) phase. During the Test phase, all types of functional testing, such as unit testing, integration testing, system testing, and acceptance testing are done, as well as non-functional testing.
5. **Deploy:** After successful testing, the system/product is delivered to/deployed by the client. As soon as the system/product is given to the client, the client will first do beta testing. A beta test is

the second phase of software testing in which a sampling of the intended audience tries the system out. If any changes are required or if any bugs are detected, then users report it to the build/coding team. Once changes are made or the bugs are fixed, then the final deployment occurs.

6. **Maintain:** Once clients start using the developed system, bugs, issues, and other problems come up that need to be resolved; in addition, system components need to be updated from time to time. The Maintain phase includes performing activities to keep system performance at peak levels, updating components to meet new standards, and replacing/retiring hardware as needed.

SDLC, Waterfall & Agile Definitions Modified from: International Software Testing Qualifications Board Certification Exam Study Materials

In addition to the defined SDLC above, the Department of Health and Human Services has developed an Enterprise Performance Life Cycle (EPLC) framework to enhance Information Technology governance. More information on the EPLC can be found at the links below:

<http://www.hhs.gov/ocio/eplc/>

http://www.hhs.gov/ocio/eplc/Enterprise%20Performance%20Lifecycle%20Artifacts/eplc_artifacts.html

B. High-Level Overview of IT Options or Solutions (COTS, SAAS, Custom)

The descriptions provided below provide high-level information on several IT options, each of which has unique considerations that will affect the resources required to plan for, implement and maintain an information system.

A **COTS** (commercial off-the-shelf) product is one that is used "as is." COTS products are designed to be easily installed and to interoperate with existing system components. Almost all software bought by the average computer user fits into the COTS category: operating systems, office product suites, word processing, and email programs are among the myriad examples. One of the major advantages of COTS software, which is mass produced, is its relatively low cost.

SAAS (Software as a Service) is a software licensing and delivery model in which software is licensed on a subscription basis, applications are hosted by a vendor or service provider, and made available to customers over a network, typically the Internet. It is sometimes referred to as "on-demand software."

Note: Program Instruction 13-01 (ACF-OA-13-01) (available at <http://www.acf.hhs.gov/programs/cb/resource/acf-oa-pi1301>) provides guidance on requesting a waiver of APD rules limiting the use of FFP for COTS products and SAAS applications, including those developed for any services programs covered by the Social Security Act.

Custom – Software that is built to the exact needs of a client. If properly designed, a custom solution should address current business needs and be scalable to expand and support future business demands. The upfront costs are more expensive than COTS or SAAS software, but clients who build a custom solution normally avoid license fees, seat fees, and upgrade fees that are common with off-the-shelf solutions.

C. Cloud vs. on Premises Computing

Child welfare information systems require infrastructure to operate. The servers, platforms, and tools that are put in place to run the application, store data, and provide backup and recovery can operate on premises or operate in the cloud. On premises normally requires the setup and operation of a data center either within the organization (example, in a state's Department of Human Services/Department of

Children and Families office) or in a partner organization (a jurisdiction’s Information Technology Department, for example), requiring an investment of capital upfront. Cloud solutions are normally accessed through the Internet and clients paying a fee for a third-party entity to provide and maintain the infrastructure offsite. Fees for cloud computing can be based on “pay as you go” or based on demand/usage of the systems. To use a simple analogy, computing on premises vs. cloud is akin to buying vs. renting.

Additional Comparisons

Feature	On Premises	Cloud
Located within the organization/jurisdiction	Yes	No
Expense type	Capital—establish a data center	Operating—pay fees to a cloud vendor
Data connectivity and speed	Controlled in-house	Controlled by cloud vendor
Software or hardware upgrades	Installed in-house	Installed at vendor site
Implementation time	Longer	Shorter

Additional Information and Key Considerations for States/Tribes/Territories

Jurisdictions are increasingly focused on operational efficiencies and leveraging technology to realize those efficiencies. Agencies with such a focus should recognize that business processes may need to be modified or reengineered, with input gathered from multiple users and stakeholders, in advance of technology changes. As such, business process redesign should be planned for and scheduled in advance of the information system project’s commencement, or as part of the early phase of design/requirements gathering.

In addition, Federal guidance for information system upgrades or replacements strongly encourages the development of new systems that promote data sharing and interoperability and include components, features, and modules that can be reused by the jurisdiction. Open source development, wherein software source code is provided to users who may modify and share updated code with others is an area where jurisdictions may benefit from collaboration among multiple users and serve to advance data sharing. While open source code can be obtained at no charge, associated costs such as maintenance and support should be considered in the planning process.

Before undertaking an information system acquisition or development effort, the jurisdiction needs to secure funding via Planning or Implementation Advanced Planning Documents (P/IAPDs), operating budgets, capital requests, or other sources. In addition, Requests for Proposals/Requests for Responses/Requests for Quotes must also be written and issued, and responses evaluated by internal staff. The costs of securing funding and selecting vendor(s) for a child welfare information system are not part of this TCO analysis; those tasks are assumed to have already been completed. However, depending on the chosen development approach (Waterfall vs. Agile) a jurisdiction may need to write multiple Requests for Proposals and the level of staff resources dedicated to the acquisition and development effort may fluctuate from high to low intensity throughout these phases. A TCO analysis based on one approach and one jurisdiction’s specific procurement requirements is expected to look very different from another jurisdiction’s. For more specific information on how these decisions and procurement processes might affect a TCO analysis, please contact your Center for [States Liaison](#).

Some projects may require Independent Verification and Validation (IV&V) and/or project management office (PMO) vendors as part of a large system implementation. These costs may have been incurred during Design, Development and Implementation (DDI) phases of the project, not necessarily maintenance and operation (M&O).

Included in the TCO Analysis	Not Included in the TCO Analysis
Implementation Advance Planning Document (IAPD) updates	Planning Advanced Planning Document (PAPD) creation
Business Process Redesign	Implementation Advance Planning Document (IAPD) creation
Cost Allocation Plan Updates and Federal Financial Participation (FFP) tracking	Requests for Proposals/Request for Responses preparation
Independent Verification and Validation (IV&V) vendor costs	Procurement Management/Proposal Evaluation
Project management office (PMO) vendor costs	

Instructions—General Guidance:

- ▶ A TCO analysis can extend over 3, 5, 10 years or the anticipated useful life of the information system – the accompanying TCO worksheet [<https://capacity.childwelfare.gov/states/focus-areas/cwis/total-cost-ownership>] has been created to support a 5 year analysis but can be modified to fit the needs of the jurisdiction.
- ▶ It is recommended that the expectancy used in the TCO be aligned with that submitted with the IAPD. Jurisdictions should [consult their Federal analyst](#) with any questions.
- ▶ States/Tribes/jurisdictions can decide how to define a year—State fiscal year, Federal fiscal year, etc., for purposes of completing the TCO.
- ▶ Costs categories should include:
 - Software, hardware, personnel, network/communication, facilities/project office location (space)
 - Sample costs for each category are included in the table below.
- ▶ Costs will be spread across different phases of the project and different fiscal years:
 - Acquisition/DDI, M&O, Change
- ▶ Costs for help desk support, hosting, and other operational support costs should be included. There will be costs for standing up environments early on in the project and maintaining the system (and its environments) in later years when the system is operational
- ▶ Cost of retiring end-of-life systems or services and cost for converting data should be taken into consideration.
- ▶ Year 1 should include agency readiness activities—updating policies, documenting business processes, alternatives analysis, establishing project office, and startup.
- ▶ Salary costs for internal staff should be fully loaded (inclusive of benefit costs) and should include expected salary adjustments (raises, cost of living increases)
- ▶ Costs of licenses, hosting fees, or chargebacks should include an estimate of year-over-year increases.

Categories of cost below are organized to reflect the costs that might be expected to acquire the tools needed (software and hardware), to ensure that the appropriate personnel are in place and involved, set up the facilities and utilities to support the system and establish and maintain the appropriate environment in which the system operates.

Cost Examples

Software Costs

- Licenses (initial and annual)
- Maintenance contact
- Warranties

Hardware Costs

- Servers
- PC/laptops/tables (initial purchase and replacement costs)
- Storage capacity
- Network administrator workstation

Network/Communications Costs

- Network (WAN, other)
- Hosting/infrastructure (cloud, onsite)
- Environment upgrades/replication
- IT support (subscription)
- Network software and hardware support

Personnel Costs

Business/Program Staff

- Project Manager and/or Program Manager
- PMO team
- Subject matter experts
- Policy and program development staff
- Testers and quality assurance staff
- Trainers/training staff

System Administration

- System operators
- System programmers
- Application programmers
- Network Administrator
- Storage management
- IT management
- Help desk staff
- Network security and disaster recovery

Contracts and Legal Staff

- Other administrative

Facilities/Other Costs

- Project office space
- Computer room upgrades
- Security upgrades
- Disaster recovery services (including offsite backup)
- IT support (subscription)

Cost Assumptions

- ▶ Full Time Equivalents can be estimated based on 2,080 hours per year (thus 1/2 time = 1,040 hours, 1/3 time = 694 hours).

- ▶ The jurisdiction has an estimate of the number of end users of the system; end user figures will drive a range of costs such as license costs, laptop/pc/tablets purchases, end user training (initial and ongoing).
- ▶ The jurisdiction may include a contingency percentage for unknown costs in their overall TCO.

Excerpt From A TCO Worksheet

Below is an example of a TCO worksheet that a jurisdiction may complete. Note the various types of cost and cost units in the left columns and the years when those costs are expected to occur in the columns to the right.

The complete TCO Worksheet can be downloaded here: <https://capacity.childwelfare.gov/states/focus-areas/cwis/total-cost-ownership>

Direct Costs						
1. Personnel--Initial Development and Deployment (sum of each FTE * #hours per year for each role)						
	# (FTE)	% FTE	Hourly Cost	# Hours	Total	Units
Contract Manager					\$0	
Project Manager					\$0	
Business Analyst					\$0	
Technical Data Analyst					\$0	
Quality assurance/testers					\$0	
Subject matter experts					\$0	
IV&V vendor cost (if applicable)					\$0	
PMO vendor cost (if applicable)					\$0	
[Add additional rows, if needed, and refer to NOTES in Lines 3-8 for guidance on inputs.]					\$0	
Total Project Management Costs						
2a. Case Management Software (includes license fees and development costs per year)						
	# (FTE)	% FTE	Hourly Cost	# Hours	Total	Units
License fees					\$0	
Implementation fees (vendor-performed services)					\$0	
Testing (internal--estimate of QA/Analyst time spent testing)					\$0	
Configuration (by vendor)					\$0	
Development--Interfaces (by vendor)					\$0	
Development--Data Migration (by vendor, by jurisdiction) (hourly cost or lump sum)					\$0	
Customization--if priced separately from Implementation					\$0	
Contingency--Software & Development Related (hourly cost, percentage, or lump sum)					\$0	
[Add additional rows, if needed, and refer to NOTES in Lines 3-8 for guidance on inputs.]					\$0	
Total Case Management Software Costs						
2b. Hardware/Infrastructure						
	# (FTE)	% FTE	Hourly Cost	# Hours	Total	Units
Servers (application server, database server)					\$0	
Software (assume virtual machine, database licenses)					\$0	
Workstations, tablets, laptops					\$0	
Network charges (includes hosting --cloud or other)					\$0	
Other infrastructure improvements/infrastructure costs					\$0	
Office space--rent and/or buildout for project staff, hosting service, DBA, etc.)					\$0	
[Add additional rows, if needed, and refer to NOTES in Lines 3-8 for guidance on inputs.]					\$0	

Support From the Center for States

The Center for States is available to provide coaching and capacity building support on the TCO worksheet. If you have questions about how the worksheet can be used, how to complete it, or other concerns related to child welfare information systems, please contact your assigned [State Liaison](#) or send an email to capacityinfo@icfi.com.