



Accounting for Software

CPE Edition

Distributed by AccountingTools, Inc.

www.accountingtools.com

Accounting for Software

Steven M. Bragg



Copyright © 2020 by AccountingTools, Inc. All rights reserved.

Published by AccountingTools, Inc., Centennial, Colorado.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to Steven M. Bragg, 6727 E. Fremont Place, Centennial, CO 80112.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For more information about AccountingTools® products, visit our Web site at www.accountingtools.com.

Course Information

Course Title: Accounting for Software

Learning Objectives:

- Recognize when software is considered to be for internal use or for sale.
- Specify the circumstances under which software costs and website development costs can be capitalized.
- Identify the steps required to recognize revenue from a software sale transaction.
- Recognize the presentation and disclosure topics relating to software.

Subject Area: Accounting

Prerequisites: None

Program Level: Overview

Program Content: Software is one of the more unusual assets, because the accounting for it varies, depending on whether it is intended for internal use or for sale to others. In the *Accounting for Software* course, we cover the accounting rules for both of these scenarios, as well as the most likely revenue recognition rules for a software sale, and the software presentation and disclosure topics that should be included in a company's financial statements.

Advance Preparation: None

Recommended CPE Credit: 1 hour

Table of Contents

Accounting for Software	1
<i>Introduction</i>	<i>1</i>
<i>Internal-Use Software.....</i>	<i>1</i>
<i>Internal-Use Software Accessed via a Hosting Arrangement.....</i>	<i>4</i>
<i>Website Development Costs.....</i>	<i>4</i>
<i>Software Intended for Sale.....</i>	<i>5</i>
Initial Cost Recognition.....	5
Subsequent Cost Measurement	7
Product Enhancements.....	7
<i>Software Inventory.....</i>	<i>7</i>
<i>Accounting for Costs to Develop Entertainment and Educational Software</i>	<i>7</i>
<i>Revenue Recognition for Software Sales</i>	<i>8</i>
Step One: Link Contract to Customer.....	8
Step Two: Note Performance Obligations	8
Step Three: Determine Prices	9
Step Four: Allocate Prices to Obligations.....	11
Step Five: Recognize Revenue	12
Measurement of Progress Completion.....	13
Output Methods	13
Input Methods.....	13
<i>Presentation Issues</i>	<i>13</i>
<i>Disclosure Issues</i>	<i>14</i>
<i>Summary.....</i>	<i>15</i>
<i>Review Questions.....</i>	<i>16</i>
Answers to Review Questions	17
Glossary	19
Index	20
Final Examination	21

About the Author

Steven Bragg, CPA, has been the chief financial officer or controller of four companies, as well as a consulting manager at Ernst & Young. He received a master's degree in finance from Bentley College, an MBA from Babson College, and a Bachelor's degree in Economics from the University of Maine. He has been a two-time president of the Colorado Mountain Club, and is an avid alpine skier, mountain biker, and certified master diver. Mr. Bragg resides in Centennial, Colorado. He has written the following books and courses:

7 Habits of Effective CEOs	Capital Budgeting
7 Habits of Effective CFOs	CFO Guidebook
7 Habits of Effective Controllers	Change Management
Accountant Ethics [for multiple states]	Closing the Books
Accountants' Guidebook	Coaching and Mentoring
Accounting Changes and Error Corrections	Conflict Management
Accounting Controls Guidebook	Constraint Management
Accounting for Breweries	Construction Accounting
Accounting for Casinos and Gaming	Corporate Bankruptcy
Accounting for Derivatives and Hedges	Corporate Cash Management
Accounting for Earnings per Share	Corporate Finance
Accounting for Income Taxes	Cost Accounting (college textbook)
Accounting for Intangible Assets	Cost Accounting Fundamentals
Accounting for Inventory	Cost Management Guidebook
Accounting for Investments	CPA Firm Mergers and Acquisitions
Accounting for Leases	Credit & Collection Guidebook
Accounting for Managers	Crowdfunding
Accounting for Mining	Developing and Managing Teams
Accounting for Retirement Benefits	Effective Collections
Accounting for Software	Effective Employee Training
Accounting for Stock-Based Compensation	Effective Innovation
Accounting for Vineyards and Wineries	Effective Leadership
Accounting Information Systems	Effective Negotiation
Accounting Procedures Guidebook	Effective Time Management
Activity-Based Costing	Employee Onboarding
Activity-Based Management	Enterprise Risk Management
Agricultural Accounting	Entertainment Industry Accounting
Auditor Independence	Ethical Frameworks in Accounting
Behavioral Ethics	Ethical Responsibilities
Better Business Writing	Excel Charts and Visualizations
Bookkeeping Guidebook	Excel Data Analysis Tools
Budgeting	Excel Data Management
Business Combinations and Consolidations	Excel Formulas and Functions
Business Insurance Fundamentals	Fair Value Accounting
Business Ratios	Fiduciary Accounting
Business Strategy	Financial Analysis
Business Valuation	Financial Forecasting and Modeling

(continued)

Fixed Asset Accounting	Law Firm Accounting
Foreign Currency Accounting	Lean Accounting Guidebook
Form 1099 Compliance	Mergers & Acquisitions
Franchise Accounting	Money Laundering
Fraud Examination	New Controller Guidebook
Fraud Schemes	New Manager Guidebook
GAAP Guidebook	Nonprofit Accounting
Governmental Accounting	Oil & Gas Accounting
Guide to Analytical Procedures	Operations Management
Guide to Audit Sampling	Optimal Accounting for Cash
Guide to Audit Working Papers	Optimal Accounting for Payables
Guide to Auditor Legal Liability	Optimal Accounting for Payroll
Guide to Data Analytics for Audits	Partnership Accounting
Health Care Accounting	Payables Management
Hospitality Accounting	Payroll Management
How to Audit Benefits & Compensation	Performance Appraisals
How to Audit Cash	Persuasive Presentations
How to Audit Equity	Project Accounting
How to Audit Fixed Assets	Project Management
How to Audit for Fraud	Property Management Accounting
How to Audit Inventory	Public Company Accounting
How to Audit Liabilities	Purchasing Guidebook
How to Audit Outsourced Functions	Real Estate Accounting
How to Audit Payroll	Records Management
How to Audit Procurement	Recruiting and Hiring
How to Audit Receivables	Revenue Management
How to Audit Revenue	Revenue Recognition
How to Conduct a Compilation	Sales and Use Tax Accounting
How to Conduct a Review	Succession Planning
How to Conduct an Audit Engagement	The Balance Sheet
How to Run a Meeting	The Income Statement
Human Resources Guidebook	The MBA Guidebook
IFRS Guidebook	The Soft Close
Interpretation of Financial Statements	The Statement of Cash Flows
Introduction to Excel	The Year-End Close
Inventory Management	Treasurer's Guidebook
Investor Relations Guidebook	Unethical Behavior
Key Performance Indicators	Working Capital Management

On-Line Resources by Steven Bragg

Steven maintains the accountingtools.com web site, which contains continuing professional education courses, the Accounting Best Practices podcast, and thousands of articles on accounting subjects.

Accounting for Software

Introduction

The accounting standards relating to software are included in two sections of the GAAP codification. Section 350 covers internal-use software and website development costs, while Section 985 covers software that is intended for sale. In this course, we cover the accounting for both scenarios, as well as software accessed via a hosting arrangement, software inventory, software revenue recognition, and related presentation and disclosure topics.

Related Podcast Episode: Episode 270 of the Accounting Best Practices Podcast discusses the accounting for software as a service. It is available at: accountingtools.com/podcasts or [iTunes](#)

Internal-Use Software

Companies routinely develop software for internal use, and want to understand how these development costs are to be accounted for. Software is considered to be for internal use when it has been acquired or developed *only* for the internal needs of a business. Examples of situations where software is considered to be developed for internal use are:

- Accounting systems
- Cash management tracking systems
- Customer service software
- Data collection and analysis systems
- Database search tools
- Membership tracking systems
- Production automation systems
- Video production systems

Further, there can be no reasonably possible plan to market the software outside of the company, as would be the case when a marketing channel has been selected that has specifically-identified promotional, delivery, billing, and support activities. A market feasibility study is not considered a reasonably possible marketing plan, nor is an arrangement that provides for the joint development of software for mutual internal use. However, a history of selling software that had initially been developed for internal use creates a reasonable assumption that the latest internal-use product will also be marketed for sale outside of the company.

EXAMPLE

Harrison Manufacturing develops just-in-time software that it uses internally to monitor the flow of jobs through its production process. The IT manager convinces management to sell the JIT software outside the company, which meets with some success. Subsequently, the firm develops warehouse management software to manage the flow of goods through its own distribution warehouses. Since Harrison has a history of selling its internally-developed software outside of the firm, there is a reasonable assumption that the same action will be taken for the new warehouse management software – which prevents the firm from accounting for it as internal-use software. Instead, it must account for the new software using the requirements for software that is intended for sale.

EXAMPLE

Inouye Semiconductor uses highly-complex wafer testing software to determine whether the semiconductor chips it is producing are operational. This is internal-use software, since customers do not acquire the software, nor do they obtain the future right to use it.

The accounting for internal-use software varies, depending upon the stage of completion of the project. The relevant accounting is:

- *Stage 1: Preliminary project activities.* All costs incurred during the preliminary stage of a development project should be charged to expense as incurred. This stage is considered to include making decisions about the allocation of resources, determining performance requirements, conducting supplier demonstrations, evaluating technology, and supplier selection.
- *Stage 2: Application development.* Capitalize the costs incurred to develop internal-use software, which may include coding, hardware installation, and testing. Any costs related to data conversion, user training, administration, and overhead should be charged to expense as incurred. Data conversion may include purging or cleansing existing data, reconciling old and new data, and converting old data to the new system. Only the following costs can be capitalized:
 - Materials and services consumed in the development effort, such as third-party development fees, software purchase costs, and travel costs related to development work.
 - Costs to develop or acquire software that allow for access to or conversion of old data by new systems.
 - The payroll and benefit costs of those employees directly associated with software development, to the extent of the time spent directly on the project.
 - The capitalization of interest costs incurred to fund the project. These costs can only be capitalized for the period during which there are activities related to software development.
- *Stage 3. Post-implementation.* Charge all post-implementation costs to expense as incurred. Examples of these costs are training and maintenance costs.

Any allowable capitalization of costs should begin *after* the preliminary stage has been completed, management authorizes and commits to funding the project, it is probable that the project will be completed, and the software will be used for its intended function.

Accounting for Software

The capitalization of costs should end when all substantial testing has been completed. If it is no longer probable that a project will be completed, stop capitalizing the costs associated with it, and conduct impairment testing on the costs already capitalized. The cost at which the asset should then be carried is the lower of its carrying amount or fair value (less costs to sell). Unless there is evidence to the contrary, the usual assumption is that uncompleted software has no fair value. The following are general indicators that software is no longer expected to be completed and placed in service:

- Expenditures are no longer being budgeted or incurred for the software.
- There are programming difficulties that will not be resolved on a timely basis.
- There have been significant cost overruns.
- The costs that have been or will be incurred significantly exceed the cost of competing third-party software, so management intends to stop development and buy the third-party software.
- New technology in the marketplace is driving management to acquire third-party products, rather than completing the internal development project.
- The business unit to which the software relates is either unprofitable or will be discontinued.

A business may purchase software for internal use. If the purchase price of this software includes other elements, such as training and maintenance fees, only capitalize that portion of the purchase price that relates to the software itself. When this purchase results in the replacement of older software, any remaining unamortized cost of the older software should be charged to expense as soon as the replacement software is ready for its intended use.

In addition, any later upgrades of the software can be capitalized, but only if it is probable that extra system functionality will result from the upgrade. These upgrades typically require the formulation of new software specifications, which may alter some portion of the existing specifications. The costs of maintaining the system should be charged to expense as incurred. If the maintenance is provided by a third party and payment is made in advance for the services of that party, amortize the cost of the maintenance over the service period.

Note: When it is not possible to separate internal costs on a cost-effective basis between maintenance and relatively minor upgrades, one should expense these costs as incurred.

Once costs have been capitalized, amortize them over the expected useful life of the software. This is typically done on a straight-line basis, unless another method more clearly reflects the expected usage pattern of the software. Amortization should begin when a software module is ready for its intended use, which is considered to be when all substantial system testing has been completed. If a software module cannot function unless other modules are also completed, do not begin amortization until the related modules are complete.

It may be necessary to regularly reassess the useful life of the software for amortization purposes, since technological obsolescence tends to shorten it. The reassessment may also involve consideration of new technology, competing products, and other economic factors.

The capitalized cost of internal-use software should be routinely reviewed for impairment. The following are all indicators of the possible presence of asset impairment:

- The software is not expected to be of substantive use
- The manner in which the software was originally intended to be used has now changed
- The software is to be significantly altered
- The development cost of the software significantly exceeded original expectations

Once a business has developed software for internal use, management may decide to market it for external use by third parties. If so, the proceeds from software licensing, net of selling costs, should be applied against the carrying amount of the software asset. For the purposes of this topic, selling costs are considered

to include commissions, software reproduction costs, servicing obligations, warranty costs, and installation costs. The business should not recognize a profit on sales of the software until the application of net sales to the carrying amount of the software asset have reduced the carrying amount to zero. The business can recognize all further proceeds as revenue.

Internal-Use Software Accessed via a Hosting Arrangement

The preceding guidance also applies to internal-use software to which the organization gains access via a hosting arrangement – but only if both of the following criteria are met:

- The company has the contractual right to take possession of the software at any time without incurring a significant penalty; and
- It is feasible to run the software on the company's own hardware, or contract with a third party to host the software.

When the preceding criteria are not met, then the hosting arrangement is considered to be a service contract, and so does not constitute a software purchase.

Under a hosting arrangement, it is relatively common for the price to include several elements, such as the software license, hosting, and employee training. When this is the case, the company should allocate the price paid among these various elements, where the allocation is based on the standalone price of each element in the contract.

If any costs associated with a hosting arrangement are capitalized, they should be amortized over the term of the hosting arrangement. The term of a hosting arrangement is considered to be its fixed non-cancellable term, plus any option extension periods, if the company is reasonably certain to exercise the option. The term of the arrangement should be periodically reassessed, with any changes in the period accounted for as a change in accounting estimate.

The straight-line amortization basis should be used, unless another method is more representative of the benefit level provided by the hosting arrangement. Amortization should begin for each module of a hosting arrangement when the module is ready for its intended use (which has occurred when all substantial testing is completed). If the functionality of the module is dependent on the completion of other modules, then amortization should begin when the other related modules are ready for their intended use.

The accountant should examine the capitalized costs associated with a hosting arrangement for impairment when any of the following conditions are present:

- The arrangement is not expected to provide any substantive service potential to the company.
- A significant change has occurred in the extent to which the hosting arrangement is used.
- A significant change has been made to the hosting arrangement.

Website Development Costs

A company may allocate funds to the development of a company website in such areas as coding, graphics design, the addition of content, and site operation. The accounting for website development varies, depending upon the stage of completion of the project. The relevant accounting is:

- *Stage 1: Preliminary.* Charge all site planning costs to expense as incurred. This stage is considered to include project planning, the determination of site functionality, hardware identification, technology usability, alternatives analysis, supplier demonstrations, and legal considerations.
- *Stage 2: Application development and infrastructure.* The accounting matches what was just described in the last section for internal-use software. In essence, capitalize these costs. More specifically, capitalize the cost of obtaining and registering an Internet domain, as well as the procurement of software tools, code customization, web page development, related hardware, hypertext link

creation, and site testing. Also, if a site upgrade provides new functions or features to the website, capitalize these costs.

- *Stage 3: Graphics development.* For the purposes of this topic, graphics are considered to be software and so are capitalized, unless they are to be marketed externally. Graphics development includes site page design and layout.
- *Stage 4: Content development.* Charge data conversion costs to expense as incurred, as well as the costs to input content into a website. Content may include articles, photos, maps, charts, and so forth.
- *Stage 5: Site operation.* The costs to operate a website are the same as any other operating costs, and so should be charged to expense as incurred. Operating costs relate to training, administration, site updates, site security, and maintenance. The treatment of selected operating costs associated with a website are:
 - Charge website hosting fees to expense over the period benefited by the hosting
 - Charge search engine registration fees to expense as incurred, since they are advertising costs

Software Intended for Sale

When computer software is to be sold, leased, or otherwise marketed for sale, there are a number of accounting requirements for how the associated costs are to be treated. Examples of software intended for sale are as follows:

- Software that is required to operate a company's products
- Machines that are sold with software incorporated into them that is needed to operate the machines
- Vehicles that are sold with automated driving systems installed in them
- A database that is sold with an integrated software search function
- An operating system that is installed on computers sold to customers

The accounting requirements for software intended for sale are noted in the following sub-sections.

Initial Cost Recognition

The following requirements apply to the initial recognition of various costs associated with computer software that is intended for sale:

- *Research and development costs.* Any cost incurred to establish the technological feasibility of software is to be charged to expense as incurred. Technological feasibility has been established when the developing entity has completed all planning, designing, coding and testing activities required to determine whether the software can meet its design specifications. Evidence of technological feasibility can be proven by engaging in either of the following activities:
 - *Detail program design option.* If a detail program design was developed for the software, then all three of the following requirements must be met in order to establish technological feasibility:
 1. The needed skills, hardware, and software technology are available to produce the product.
 2. The detail program design is sufficiently complete, based on design documentation and by tracing the design to product specifications.
 3. The detail program design has been examined for high-risk development issues and these concerns have been resolved with coding and testing. Examples of high-

Accounting for Software

risk development issues are unproven functions and features that incorporate technological innovations.

Note: When a high-risk development issue is discovered after technological feasibility was established and costs capitalized, this is considered a change in accounting estimate that requires all previously capitalized costs to be charged to expense.

Note: When software does not include all of the features originally planned for it and the product is not sellable without the dropped features, then technological feasibility has not been established for it.

- *No detail program design.* When there is no detail program design, then technological feasibility can be proven by having a completed product design and a working model of the software, where the completeness of the model and its consistency with the product design have been confirmed with sufficient testing. A working model must be operative, written in the same language as the product to be sold, include all planned major functions, and be ready for initial customer testing.

Note: When a product consists of several modules that cannot be sold separately, technological feasibility must be established for the product as a whole, rather than on a module-by-module basis. This means that the detail program design or the working model for the entire product must be completed before capitalization can begin.

- *Production costs.* The recognition of production costs is addressed in the following topics:
 - *Production masters.* Once technological feasibility has been established, the cost of producing production masters should be capitalized. These costs may include any coding and testing conducted after technological feasibility was established.
 - *Software production costs.* Software production costs cannot be capitalized until technological feasibility has been established, and all research and development activities for the other components of the product have been completed.
 - *Allocated overhead costs.* Overhead costs can be allocated to production. This allocation may include overhead relating to programmers and their work facilities, but should not include an allocation of general and administrative costs.
 - *Capitalization stoppage.* The capitalization of production costs must stop when the product is available for general release to customers.
 - *Maintenance and customer support costs.* The costs of maintenance and customer support should be charged to expense at the earlier of when these costs are incurred or when related revenue is recognized.
- *Purchased software.* When software is purchased in order to be integrated into a product, the acquisition cost should only be capitalized if technological feasibility can be established *and* all research and development activities of the other components of the product have been completed as of the software purchase date. If there is an alternative use for the purchased software, then the purchase cost can be capitalized in accordance with its projected use.

EXAMPLE

Norrone Software buys software for \$50,000 that it can resell for \$40,000. Technological feasibility has not yet been established. At this point, \$10,000 of the purchase cost should be charged to research and development expense, while the remaining \$40,000 is capitalized. The technological feasibility of the software product is then established, so Norrone can include the capitalized \$40,000 in the cost of the product.

Subsequent Cost Measurement

Once software costs have been capitalized, they should be amortized on a product-by-product basis, beginning with the date when the product is available for general release to customers. The amount of annual amortization recognized should be the greater of the following two calculation options:

- The ratio of current gross product revenues to the total of both current and expected future gross revenues for that product; or
- The straight-line method over the remaining estimated economic life of the product.

At the end of each reporting period, compare the unamortized capital costs of each software product to its net realizable value, and write off any amount by which the unamortized capital costs exceed the net realizable value. Net realizable value is the estimated future gross revenues for that product, minus the estimated future costs of completing and disposing of it, which includes any maintenance and customer service costs. Any amounts written off shall not be restored in future periods.

Product Enhancements

Enhancements may be made to a product after it has been initially released for sale. The costs incurred for these enhancements should be charged to research and development expense until such time as the technological feasibility of each enhancement has been established. Technological feasibility tends to be easier to establish for a product enhancement, since it was already established for the baseline product. This is particularly the case when a software product has been ported (adapted) to run on different hardware.

If the original product (prior to the enhanced version), will no longer be sold, then include any unamortized cost of the original version in the cost of the enhancement when applying the net realizable value test. However, if the original product will continue to be sold along with the enhanced version, then allocate the unamortized cost of the original product between the original and enhanced versions.

The capitalized costs of a product enhancement, including any costs allocated to it from the original product, should be amortized over the estimated useful life of the enhanced product.

Software Inventory

When a software provider incurs costs to duplicate software, documentation, and training materials from product masters, as well as to package these items, the costs should be capitalized as inventory. These capitalized costs are then charged to the cost of goods sold when the revenue associated with the sale of these units is recognized.

Accounting for Costs to Develop Entertainment and Educational Software

The Securities and Exchange Commission (SEC) observer sitting in on an Emerging Issues Task Force meeting has pointed out that the SEC is aware of diversity in the entertainment industry in how software development costs relating to educational and entertainment products are accounted for. The treatment ranges from capitalizing all costs to expensing a significant proportion of them as incurred. The SEC observer pointed out that educational and entertainment products that are sold, leased, or otherwise marketed

are subject to the requirements of Topic 985, *Software*. This course dealt with Topic 985 in its *Software Intended for Sale* and *Software Inventory* sections.

Revenue Recognition for Software Sales

There used to be a number of revenue recognition rules relating to the sale of software, which were eliminated when Topic 606 of the Accounting Standards Codification, *Revenue from Contracts with Customers*, was implemented. Topic 606 establishes a series of actions that an entity takes to determine the amount and timing of revenue to be recognized. The main steps are:

1. Link the contract with a specific customer.
2. Note the performance obligations required by the contract.
3. Determine the price of the underlying transaction.
4. Match this price to the performance obligations through an allocation process.
5. Recognize revenue as the various obligations are fulfilled.

We will expand upon each of these steps in the following sub-sections, limiting the discussion to those aspects of revenue recognition that are most likely to apply to the sale of software.

Step One: Link Contract to Customer

The contract is used as a central aspect of revenue recognition, because revenue recognition is closely associated with it. In many instances, revenue is recognized at multiple points in time over the duration of a contract, so linking contracts with revenue recognition provides a reasonable framework for establishing the timing and amounts of revenue recognition.

A contract only exists if there is an agreement between the parties that establishes enforceable rights and obligations. It is not necessary for an agreement to be in writing for it to be considered a contract.

Step Two: Note Performance Obligations

A performance obligation is essentially the unit of account for the goods or services contractually promised to a customer. The performance obligations in the contract must be clearly identified. This is important in recognizing revenue, since revenue is considered to be recognizable when goods or services are transferred to the customer.

If there is more than one good or service to be transferred under the contract terms, only break it out as a separate performance obligation if it is a distinct obligation or there are a series of transfers to the customer of a distinct good or service. In the latter case, a separate performance obligation is assumed if there is a consistent pattern of transfer to the customer.

The “distinct” label can be applied to a good or service only if it meets both of the following criteria:

- *Capable of being distinct*. The customer can benefit from the good or service as delivered, or in combination with other resources that the customer can readily find; and
- *Distinct within the context of the contract*. The promised delivery of the good or service is separately identified within the contract.

Goods or services are more likely to be considered distinct when:

- The seller does not use the goods or services as a component of an integrated bundle of goods or services.
- The items do not significantly modify any other goods or services listed in the contract.
- The items are not highly interrelated with other goods or services listed in the contract.

Accounting for Software

The intent of these evaluative factors is to place a focus on how to determine whether goods or services are truly distinct within a contract. There is no need to assess the customer's intended use of any goods or services when making this determination.

To reduce the cost of noting performance obligations, it is not necessary to assess whether promised goods or services are performance obligations if they are immaterial in the context of the contract with the customer.

EXAMPLE

Norrna Software enters into a contract with a Scandinavian clothing manufacturer to transfer a software license for its clothing design software. The contract also states that Norrna will install the software and provide technical support for a two-year period. The installation process involves adjusting the data entry screens to match the needs of the clothing designers who will use the software. The software can be used without these installation changes. The technical support assistance is intended to provide advice to users regarding advanced features, and is not considered a key requirement for software users.

Since the software is functional without the installation process or the technical support, Norrna concludes that the items are not highly interrelated. Since these goods and services are distinct, the company should identify separate performance obligations for the software license, installation work, and technical support.

In the event that a good or service is not classified as distinct, aggregate it with other goods or services promised in the contract, until such time as a cluster of goods or services have been accumulated that can be considered distinct.

Step Three: Determine Prices

This step involves the determination of the transaction price built into the contract. The transaction price is the amount of consideration to be paid by the customer in exchange for its receipt of goods or services. The transaction price does not include any amounts collected on behalf of third parties (such as sales taxes).

EXAMPLE

The Productivity Software Company sells its software to individuals through its chain of retail stores. In the most recent period, the company generated \$3,800,000 of receipts, of which \$200,000 was sales taxes collected on behalf of local governments. Since the \$200,000 was collected on behalf of third parties, it cannot be recognized as revenue.

If the transaction price is to be paid over a period of time (as may be the case with a large software sale), this implies that the seller is including a financing component in the contract. If this financing component is a significant financing benefit for the customer and provides financing for more than one year, adjust the transaction price for the time value of money. In cases where there is a financing component to a contract, the seller will earn interest income over the term of the contract.

A contract may contain a financing component, even if there is no explicit reference to it in the contract. When adjusting the transaction price for the time value of money, consider the following factors:

- *Standalone price.* The amount of revenue recognized should reflect the price that a customer would have paid if it had paid in cash.
- *Significance.* In order to be recognized, the financing component should be significant. This means evaluating the amount of the difference between the consideration to be paid and the cash selling price. Also note the combined effect of prevailing interest rates and the time difference between when delivery is made and when the customer pays.

Accounting for Software

If it is necessary to adjust the compensation paid for the time value of money, use as a discount rate the rate that would be employed in a separate financing transaction between the parties as of the beginning date of the contract. The rate used should reflect the credit characteristics of the customer, including the presence of any collateral provided. This discount rate is not to be updated after the commencement of the contract, irrespective of any changes in the credit markets or in the credit standing of the customer.

EXAMPLE

Norrna Software sells one license for enterprise resources planning software to Eskimo Construction, under generous terms that allow Eskimo to pay Norrna the full amount of the \$119,990 receivable in 24 months. The cash selling price of the software is \$105,000. The contract contains an implicit interest rate of 6.9%, which is the interest rate that discounts the purchase price of \$119,990 down to the cash selling price over the two-year period. The controller examines this rate and concludes that it approximates the rate that Norrna and Eskimo would use if there had been a separate financing transaction between them as of the contract inception date. Consequently, Norrna recognizes interest income during the two-year period prior to the payment due date, using the following calculation:

<u>Year</u>	<u>Beginning Balance</u>	<u>Interest (at 6.9% Rate)</u>	<u>Ending Balance</u>
1	\$105,000	\$7,245	\$112,245
2	112,245	7,745	\$119,990

As of the shipment date, Norrna records the following entry:

	<u>Debit</u>	<u>Credit</u>
Loan receivable	105,000	
Revenue		105,000

At the end of the first year, Norrna recognizes the interest associated with the transaction for the first year, using the following entry:

	<u>Debit</u>	<u>Credit</u>
Loan receivable	7,245	
Interest income		7,245

At the end of the second year, Norrna recognizes the interest associated with the transaction for the second year, using the following entry:

	<u>Debit</u>	<u>Credit</u>
Loan receivable	7,745	
Interest income		7,745

These entries increase the size of the loan receivable until it reaches the original sale price of \$119,990. Eskimo then pays the full amount of the receivable, at which point Norrna records the following final entry:

	<u>Debit</u>	<u>Credit</u>
Cash	119,990	
Loan receivable		119,990

There is assumed *not* to be a significant financing component to a contract in the presence of any of the following factors:

- *Advance payment.* The customer paid in advance, and the customer can specify when goods and services are to be delivered.
- *Variable component.* A large part of the consideration to be paid is variable, and payment timing will vary based on a future event that is not under the control of either party.
- *Non-financing reason.* The reason for the difference between the contractual consideration and the cash selling price exists for a reason other than financing, and the amount of the difference is proportional to the alternative reason.

Step Four: Allocate Prices to Obligations

Once the performance obligations and transaction prices associated with a contract have been identified, the next step is to allocate the transaction prices to the obligations. The basic rule is to allocate that price to a performance obligation that best reflects that amount of consideration to which the seller expects to be entitled when it satisfies each performance obligation. To determine this allocation, it is first necessary to estimate the standalone selling price of those distinct goods or services as of the inception date of the contract. If it is not possible to derive a standalone selling price, the seller must estimate it. This estimation should involve all relevant information that is reasonably available, such as:

- Competitive pressure on prices
- Costs incurred to manufacture or provide the item
- Item profit margins
- Pricing of other items in the same contract
- Standalone selling price of the item
- Supply and demand for the items in the market
- The seller's pricing strategy and practices
- The type of customer, distribution channel, or geographic region
- Third-party pricing

The following three approaches are acceptable ways in which to estimate a standalone selling price:

- *Adjusted market assessment.* This involves reviewing the market to estimate the price at which a customer in that market would be willing to pay for the goods and services in question. This can involve an examination of the prices of competitors for similar items and adjusting them to incorporate the seller's costs and margins.
- *Expected cost plus a margin.* This requires the seller to estimate the costs required to fulfill a performance obligation, and then add a margin to it to derive the estimated price.
- *Residual approach.* This involves subtracting all of the observable standalone selling prices from the total transaction price to arrive at the residual price remaining for allocation to any non-observable selling prices. This method can only be used if one of the following situations applies:
 - The seller sells the good or service to other customers for a wide range of prices; or
 - No price has yet been established for that item, and it has not yet been sold on a standalone basis.

The residual approach can be difficult to use when there are several goods or services with uncertain standalone selling prices. If so, it may be necessary to use a combination of methods to derive standalone selling prices, which should be used in the following order:

1. Estimate the aggregate amount of the standalone selling prices for all items having uncertain standalone selling prices, using the residual method.
2. Use another method to develop standalone selling prices for each item in this group, to allocate the aggregate amount of the standalone selling prices.

Once all standalone selling prices have been determined, allocate the transaction price amongst these distinct goods or services based on their relative standalone selling prices.

Tip: Appropriate evidence of a standalone selling price is the observable price of a good or service when the seller sells it to a similar customer under similar circumstances.

Once the seller derives an approach for estimating a standalone selling price, it should consistently apply that method to the derivation of the standalone selling prices for other goods or services with similar characteristics.

EXAMPLE

Disaster Recovery Corporation produces apps for the iPhone. The company receives an order from the federal government for 50,000 units of its hurricane preparedness app, as well as for 10,000 units of a new app that sends out alerts when tornadoes are nearby. The company has not yet sold the new tornado app to anyone. The total price of the order is \$600,000. The company assigns \$450,000 of the total price to the hurricane preparedness app, based on its own sales of comparable orders. This leaves \$150,000 of the total price that is allocable to the tornado app. Since the company has not yet established a price for this app and has not sold it on a standalone basis, it is acceptable to allocate \$150,000 to the tornado app under the residual approach.

If there is a subsequent change in the transaction price, allocate that change amongst the distinct goods or services based on the original allocation that was used at the inception of the contract. If this subsequent allocation is to a performance obligation that has already been completed and for which revenue has already been recognized, the result can be an increase or reduction in the amount of revenue recognized. This change in recognition should occur as soon as the subsequent change in the transaction price occurs.

Step Five: Recognize Revenue

Revenue is to be recognized as goods or services are transferred to the customer. This transference is considered to occur when the customer gains control over the good or service. Indicators of this date include the following:

- When the seller has the right to receive payment.
- When the customer has legal title to the transferred asset. This can still be the case even when the seller retains title to protect it against the customer's failure to pay.
- When physical possession of the asset has been transferred by the seller.
- When the customer has taken on the significant risks and rewards of ownership related to the asset transferred by the seller. For example, the customer can now sell, pledge, or exchange the asset.
- When the customer accepts the asset.
- When the customer can prevent other entities from using or obtaining benefits from the asset.

Measurement of Progress Completion

When a performance obligation is being completed over a period of time, the seller recognizes revenue through the application of a progress completion method. The goal of this method is to determine the progress of the seller in achieving complete satisfaction of its performance obligation. This method is to be consistently applied over time, and shall be re-measured at the end of each reporting period.

Both output methods and input methods are considered acceptable for determining progress completion. The method chosen should incorporate due consideration of the nature of the goods or services being provided to the customer. The following sub-sections address the use of output and input methods.

Output Methods

An output method recognizes revenue based on a comparison of the value to the customer of goods and services transferred to date to the remaining goods and services not yet transferred. There are numerous ways to measure output, including:

- Surveys of performance to date
- Milestones reached
- The passage of time
- The number of units delivered
- The number of units produced

Another output method that may be acceptable is the amount of consideration that the seller has the right to invoice, such as billable hours. This approach works when the seller has a right to invoice an amount that matches the amount of performance completed to date.

Input Methods

An input method derives the amount of revenue to be recognized based on the to-date effort required by the seller to satisfy a performance obligation relative to the total estimated amount of effort required. Examples of possible inputs are costs incurred and labor hours expended.

A method based on output is preferred, since it most faithfully depicts the performance of the seller under the terms of a contract. However, an input-based method is certainly allowable if using it would be less costly for the seller, while still providing a reasonable proxy for the ongoing measurement of progress.

Presentation Issues

The following issues relate to the presentation of information for capitalized costs related to a software hosting arrangement:

- *Amortization classification.* The amortization expense for capitalized costs for a hosting arrangement should be included in the same line item on the income statement as the expense for fees associated with the hosting arrangement.
- *Balance sheet presentation.* The capitalized implementation costs for a hosting arrangement should be included in the same line item as the prepayment of hosting fees.
- *Statement of cash flows presentation.* The cash flows from capitalized implementation costs should be classified in the same manner as the cash flows for the associated hosting fees.

Accounting for Software

The following issues relate to the presentation of information for software that is intended for sale:

- *Amortization classification.* The amortization expense for software should be charged to the cost of sales on the income statement.
- *Balance sheet presentation.* Present capitalized software costs with a life of more than one year in the other asset line item.

Disclosure Issues

The disclosures that accompany the financial statements should include the following items:

- Unamortized software costs
- The amount charged to expense for software amortization
- The software amount written down to net realizable value
- The nature of any hosting arrangements that are service contracts

SAMPLE DISCLOSURE

Research and development expenses consist primarily of salary and related expenses, including stock-based compensation, for personnel related to the development of improvements and expanded features for our services, as well as quality assurance, testing, product management and allocated overhead. Research and development costs are expensed as incurred except for internal use software development costs that qualify for capitalization. The Company reviews development costs incurred for internal-use software in the application development stage and assesses costs for capitalization.

SAMPLE DISCLOSURE

The Company capitalizes software development costs incurred in connection with the localization and translation of its products once technological feasibility has been achieved based on a working model. A working model is defined as an operative version of the computer software product that is completed in the same software language as the product to be ultimately marketed, performs all the major functions planned for the product and is ready for initial customer testing (usually identified as beta testing). In addition, the Company capitalizes software purchased from third parties or through business combinations as acquired software technology, if the related software under development has reached technological feasibility.

The amortization of capitalized software costs is the greater of the straight-line basis over three years, the expected useful life, or a computation using a ratio of current revenue for a product compared to the estimated total of current and future revenues for that product. The Company periodically compares the unamortized capitalized software costs to the estimated net realizable value of the associated product. The amount by which the unamortized capitalized software costs of a particular software product exceeds the estimated net realizable value of that asset would be reported as a charge to the Consolidated Statements of Operations and Comprehensive Income (Loss). Capitalized software costs and accumulated amortization at January 31 were as follows:

(000s)	20X2	20X1
Capitalized software costs:		
Capitalized software development costs	\$2,314	\$1,516
Acquired software technology	<u>135</u>	<u>--</u>
	2,449	1,516
Less accumulated amortization	<u>-851</u>	<u>-526</u>
Capitalized software costs, net	<u>\$1,598</u>	<u>\$990</u>

It is the Company's policy to write off capitalized software development costs once fully amortized. Accordingly, during fiscal 20X2, \$0.3 million of costs and accumulated amortization was removed from the Consolidated Balance

Accounting for Software

Sheet and was primarily related to acquired software technology which was fully amortized during fiscal 20X2. Amortization of capitalized software costs is included in “Cost of license fees” in the accompanying Consolidated Statements of Operations and Comprehensive Income. The following table summarizes the estimated amortization expense relating to the Company’s capitalized software costs as of January 31, 20X2:

<u>Fiscal Years</u>	<u>(000s)</u>
20X3	\$759
20X4	572
20X5	226
20X6	27
Thereafter	<u>14</u>
	<u>\$1,598</u>

Summary

The accounting rules are significantly different for internal-use software and software intended for sale, so there should be clear internal guidelines for how to identify each situation. Impairment is also a significant concern, since software can be rendered technologically obsolete in short order; this means that the accounting staff must continually monitor the circumstances to see if an impairment analysis is warranted. Finally, the abbreviated revenue recognition rules noted in this course should apply to most software sales; for a more complete treatment of the topic, see the author’s *Revenue Recognition* course.

Review Questions

1. The accountant should examine the capitalized costs associated with a hosting arrangement for impairment when any of the following conditions are present, except for:
 - a. A significant change has been made to the hosting arrangement
 - b. The arrangement is not expected to provide any substantial service potential
 - c. When the term of the hosting arrangement is unusually long
 - d. A significant change has occurred in the extent to which the arrangement is used
2. The following accounting issue applies to the production costs for software intended for sale:
 - a. If a working model is not operative, costs are charged to expense as incurred
 - b. If there is an alternative use for purchased software, it can be capitalized in accordance with its projected use
 - c. If technological feasibility is not established, costs are charged to expense as incurred
 - d. The cost of producing production masters should be capitalized
3. A standalone selling price may be estimated by using all of the following methods, except for:
 - a. Residual approach
 - b. Adjusted market assessment
 - c. Expected cost plus a margin
 - d. Transfer pricing

Answers to Review Questions

1. The accountant should examine the capitalized costs associated with a hosting arrangement for impairment when any of the following conditions are present, except for:
 - a. A significant change has been made to the hosting arrangement
 - b. The arrangement is not expected to provide any substantial service potential
 - c. When the term of the hosting arrangement is unusually long
 - d. A significant change has occurred in the extent to which the arrangement is used
 - a. Incorrect. A significant change to the hosting arrangement can trigger an impairment analysis, depending on the nature of the change.
 - b. Incorrect. When the arrangement is not expected to provide any substantial service potential, then an impairment analysis is warranted.
 - c. **Correct. There is no reason for a long hosting arrangement to trigger an impairment analysis, since it does not impact the viability of the arrangement.**
 - d. Incorrect. A significant change in the extent to which the arrangement is used is grounds for an impairment analysis.
2. The following accounting issue applies to the production costs for software intended for sale:
 - a. If a working model is not operative, costs are charged to expense as incurred
 - b. If there is an alternative use for purchased software, it can be capitalized in accordance with its projected use
 - c. If technological feasibility is not established, costs are charged to expense as incurred
 - d. The cost of producing production masters should be capitalized
 - a. Incorrect. The working model concept applies to the research and development phase for software intended for sale.
 - b. Incorrect. The alternative use concept applies to purchased software, not the production costs for software.
 - c. Incorrect. The technological feasibility concept applies to the research and development phase for software intended for sale.
 - d. **Correct. The cost of producing production masters is part of the production costs related to software, since those masters are used to create software copies.**

Answers to Chapter Questions

3. A standalone selling price may be estimated by using all of the following methods, except for:
- a. Residual approach
 - b. Adjusted market assessment
 - c. Expected cost plus a margin
 - d. Transfer pricing
- a. Incorrect. The concept of subtracting all standalone selling prices from the total transaction price to arrive at the residual amount that is applied to the remaining products is an accepted method for estimating a standalone selling price.
- b. Incorrect. Estimating standalone prices from what customers would be willing to pay is an accepted method for estimating a standalone selling price. This is a market-based approach, since it depends upon information gathered from third parties.
- c. Incorrect. It is acceptable to derive a standalone price by estimating the relevant cost and adding a margin to it. This is a cost-based approach, since it is derived from a compilation of the costs associated with a product.
- d. **Correct. Transfer pricing is an accounting method used to assign costs to goods that are transferred between subsidiaries; it is unrelated to the concept of the standalone selling price.**

Glossary

C

Coding. Detailed instructions in a computer language to enact the requirements stated in the associated detail program design.

D

Detail program design. The detailed specifications of a computer software product that takes product functions, features, and technical requirements to their most detailed form.

H

Hosting arrangement. An arrangement in which the end user of software does not take possession of it; instead, the software resides on the supplier's or a third party's hardware, where the customer accesses it.

M

Maintenance. Those actions taken after a product is available for general release, to correct errors or update the product with more current information.

P

Preliminary project stage. A stage during the development of software when performance requirements are determined, alternatives are explored, and technology feasibility studies are conducted, along with several related activities.

Product design. The logical representation of all product functions in enough detail to serve as product specifications.

Product enhancement. Improvements made to an existing product that are intended to prolong its life or significantly enhance its marketability. Product enhancements usually call for a redesign of all or part of an existing product.

Product master. A completed version of a software product that is ready for copying.

S

Software. A set of programs that interact with each other, causing a computer to perform work.

T

Testing. The performance of those steps needed to determine whether software meets the function, feature, and technical performance requirements stated in the product design.

W

Working model. An operational version of software that performs all major functions planned for it, and which is ready for initial customer testing.

Index

Adjusted market assessment	11	Price determination.....	9
Contracts, linkage to customers	8	Progress completion method.....	13
Disclosure issues.....	14	Residual approach.....	12
Discount rate.....	10	Revenue recognition	12
Distinct criterion	8	Steps	8
Expected cost plus a margin	11	Software	
Input methods	13	Hosted for internal use.....	4
Output methods	13	Intended for sale	5
Performance obligations	8	Internal-use	1
Presentation issues.....	13	Inventory.....	7
Price allocations.....	11	Product enhancements	7
		Time value of money	9
		Website development costs.....	4

Final Examination

The final examination for this course is provided below. Feel free to circle your choice for the best answer to each question. To enter your answers online and receive an immediate grade and completion certificate, follow these steps:

1. Go to www.accountingtools.com/cpe
 2. Click on the “Access the Training Module | Complete a Test” button near the top of the page.
 3. Login with your user name and password.
 4. Select the **Take a Test** option and then select the **Programs** option. Click on the program that you want to take.
 5. Take the test. You can stop and restart the test at any time.
-

1. The following is an example of internal-use software:
 - a. A membership tracking system
 - b. Software that is required to operate a company’s products
 - c. An operating system that is installed on computers sold to customers
 - d. A vehicle that is sold with an automated driving system in it
2. The stage of internal-use software development that can be capitalized is:
 - a. Preliminary planning
 - b. Application development
 - c. Subsequent use
 - d. Post-implementation
3. When developing a website, the following cost can be charged to expense (rather than being capitalized):
 - a. Graphics development
 - b. Internet domain registration
 - c. Data conversion costs
 - d. Code customization
4. The following factor should be present before including a financing component in a contract:
 - a. A short payment interval
 - b. There is a significant difference between the contract price and the cash price
 - c. The credit rating of the customer
 - d. The cash reserves of the seller
5. The following presentation issue is valid for software:
 - a. The amortization expense for software intended for sale is charged to the cost of goods sold
 - b. Capitalized software costs are included in the prepaid assets line item
 - c. The cash flows from capitalized implementation costs are included in the investing activities section
 - d. The amortization expense for capitalized costs for hosting arrangements should be included in the information technology expenses line item