

CHAPTER IV: SOFTWARE AND DATABASES

Introduction

Of the three new fixed asset categories introduced in the 1993 SNA the one with greatest impact on the magnitude of GDP in most countries was software (including databases). Most OECD countries had adopted the new standard by the year 2000, but it was not long after before it became apparent that country estimates of software gross fixed capital formation (GFCF) varied considerably not only in their size relative to GDP, but also in the growth rates of the volume estimates.

In October 2001 an OECD Task Force was set up to address the issue, and one of its first actions was to conduct a survey of member countries. The survey had several aims:

- a. quantify the differences between estimates,
- b. identify what the conceptual treatments were in countries and the rationale for them,
- c. determine the different methods being used to quantify the various software flows (GFCF, trade in software, etc.) and what might constitute best practice, and
- d. determine how countries compiled price indices for deflating software and what might constitute best practice.

The OECD Task Force made a large number of recommendations covering the definition of software, the scope of software that should be recorded as GFCF, the treatment of originals and copies and how licences-to-use and licences-to-reproduce software should be dealt with, how to differentiate between GFCF and maintenance, how to estimate the value of own-account GFCF of software and the derivation of appropriate price indices to derive volume estimates of software GFCF.

In the course of its work the OECD Task Force found that some of the recommendations made in the 1993 SNA either required clarification or review, and so once the UNSC had decided that a revision of the SNA should be undertaken they were brought to the attention of the newly formed Canberra II Group. The Canberra II Group subsequently proposed to the ISWGNA that two issues concerning software and databases should be included in the SNA review, namely *Originals and copies* and *Databases*. These were then formally adopted by the UNSC as issues to be considered in the SNA review: the issue descriptions are as follows:

Originals and copies

Following the 1993 SNA's introduction of computer software as capital formation, it became more evident that the SNA does not provide guidance on the treatment of originals and copies as distinct products. Should expenditures on originals and copies both be recorded as expenditure (on new goods) on the basis that originals are distinct from copies, or should originals be considered as being analogous to a 'stock' of copies, and so expenditure on a copy partly (or mostly) reflects a sale of an existing good? How should the transactions in copies be recorded?

Databases

The 1993 SNA recommends that large databases should be capitalised. Should the SNA provide a clearer definition of databases to be capitalised covering characteristics such as size and marketability of the data as well as the database itself, or should all databases be capitalised? How should the value of a database be determined?

The outcomes of the SNA review of these two issues is described in detail in the relevant sections of the Handbook, but a few key recommendations of the 2008 SNA deserve particular attention:

- a. Originals and copies are recognised as independent fixed assets providing they meet the general definition of an asset;
- b. If a licence to use a copy (*e.g.* software) is purchased with annual payments over a multi-year contract, and if the licensee assumes all the risks and rewards associated with economic ownership of the copy, this may be regarded as the acquisition of an asset under a financial lease;
- c. If annual payments are made for a licence to use a copy without a long-term contract, the payments are treated as payments for a service under an operating lease;
- d. If the terms under which a unit is given permission to reproduce copies resemble an operating lease, then the payments to the holder of the original are recorded as payments for a service. If the holder of the original divests itself of part or all of the responsibility to issue and service copies under licences to use, this constitutes the sale of part or all of the asset represented by the original;
- e. All databases holding data with a useful life of more than one year are fixed assets;
- f. In the absence of a more satisfactory alternative, the value of a database created on own account should be valued on a sum of cost basis, with the database management system (DBMS) recorded separately as software. The costs of acquiring the data are not included in the value of the database; and
- g. Databases for sale should be valued at their market price, which includes the value of the information content.

Recommendations (a) and (b) are consistent with the OECD Task Force's interpretation of the 1993 SNA, but recommendation (c) marks a change. The Task Force recommended that if the licensee has the intention to renew an annual licence-to-use then the expenditures should be recorded as GFCF, but the 2008 SNA takes a stronger view on the meaning of 'intention' and states that the contract must be for more than a year for GFCF to occur. Recommendation (d) is a change to the SNA, because it explicitly allows for treating the sale of a licence-to-reproduce as the sale of the whole original, or part of it. While for databases, the recommendations mark a complete revamp in the 2008 SNA. With the exception of recommendation (c), all of these changes and clarifications are consistent with the recommendations made by the Canberra II Group (Ahmad 2004a, 2005 and 2004b).

The guidelines provided in this handbook largely reflect the recommendations made by the OECD Task Force in its report to the OECD Working Party on National Accounts in 2002 (OECD 2002). Some, but not all, of the differences arise from SNA recommendations (c) to (g), above.

27. Software

Software GFCF accounts for more than 1% of GDP in many OECD countries and its share is growing. It is also of special interest because investment in software and other ITC products have been found to be significant contributors to growth in output (Colecchia, 2001). This makes it very important that software GFCF and related capital measures should be measured accurately and in an internationally comparable way.

In its survey of OECD countries in 2001-2, the OECD Software Task Force found a considerable variation in all aspects of the measurement of software: intermediate consumption, software GFCF, volume measures, consumption of fixed capital and capital stock. This part of the Handbook focuses on the measurement of software GFCF.

As noted above, the guidelines presented here are largely based on the report of the OECD Software Task Force presented to the 2002 meeting of the OECD National Accounts Experts. Besides the differences arising from the revised recommendations in the 2008 SNA, there are other differences arising from the following three factors:

- The outcomes of the 2008 software survey of OECD member and accession countries
- The introduction of new industry and commodity classification systems
- New information of country practices from reports and papers

27.1 *Definition and scope*

In the 2008 SNA computer software and databases are recognised as two sub-categories of the category “computer software and databases”, and the SNA defines computer software as follows:

10.110 Computer software consists of computer programs, program descriptions and supporting materials for both systems and applications software. Gross fixed capital formation in computer software includes both the initial development and subsequent extensions of software as well as acquisition of copies that are classified as assets.

10.111 The development of computer software represents the development of an IPP. It is treated as an asset if it is to be used in production by its owner for more than one year. The software may be intended only for own use or may be intended for sale by means of copies. If copies of the software are sold on the market, their treatment follows the principles described in paragraph 10.101. Software purchased on the market is valued at purchasers’ prices, while software developed in-house is valued at its estimated basic price, or at its costs of production if it is not possible to estimate the basic price.

The act of creating an original piece of software leads to the acquisition of a fixed asset if the original satisfies the conditions of an asset, *i.e.* it is expected to be a source of economic benefits to the owner over a period of years. These benefits derive from allowing other units to use the content of the original by means of issuing licences for a fee and/or the owner using the original directly.

27.2 *Licences to use and reproduce*

Licences may be issued for use by one or a specified number of users, or may be issued with permission to reproduce copies. These are referred to as “licences-to-use” and “licences-to-reproduce” respectively.

It is useful to distinguish between the sub-categories “original software” and reproduced software, otherwise known as “software copies”, in more detail. This should help to avoid mistakes made in the past by some national accountants in not valuing “originals” as fixed assets on the mistaken grounds that doing so resulted in double-counting.

- a. Original software: Original software should be considered as machines used in the process of producing other products, and as such are considered as investment. Originals can be produced on own-account (they are then called an “own-account original software”) or can be bought (“purchased original software”). This includes games’ originals. Games software is treated in the same way as conventional software because of the similar production processes (and producers) for games and conventional software. There are two types of originals:
 - i. Originals for reproduction: Original software intended to be reproduced for sale or lease, which are generally produced by specialist software companies.
 - ii. Other originals: Software intended to be used in the process of production of other products, and generally produced on own-account or acquired as custom-made software from a specialist software company.
- b. Software copies: Software copies are reproductions of original software. They include software giving users the rights, or licence, to use, and software that gives the rights, or licences, to reproduce:
 - i. Licences-to-use: They are mostly marketed, and are referred to by a variety of names, including “packaged software” “packaged software” or “off-the-shelf software”. In general, they legally provide a licence to use the software. This category includes software copies for final use and software copies for bundling in hardware, other equipment or other software. This category also covers “multiple copy” licences-to-use and software “rented” for use, for which payments often take the form of “royalties”. It excludes licences that permit copies to be made for sale.
 - ii. Licences-to-reproduce: Licences-to-reproduce permit companies to make further software copies for subsequent sale. These copies can be sold via licences-to-use or as part of a bundle, whether the bundled software is included separately or embedded directly onto hardware. Often, licences-to-reproduce are paid for as royalties.

The acquisition of a licence-to-use or a licence-to-reproduce may be recorded as either GFCF or intermediate consumption, depending on the circumstances – see section 1.3. The acquisition of a licence to use is recorded as GFCF if the licence is for more than one year and the licensee assumes all the risks and rewards of ownership. A licence to reproduce is only an asset if the holder of the original divests itself of part, or all, of the asset represented by the original. This occurs usually when the holder of the original sells its rights to issue and service copies in a country or group of countries.

27.3 *Bundled/embedded software*

Bundling/embedding of software occurs when software copies are purchased or produced with the explicit intention of on-selling as part of, or within, another product – be it office machinery, other machinery, other software, etc. Bundled/embedded software can be created in one of two ways. First, when copies are purchased from a software producer and subsequently bundled and sold on to another consumer. Second, when a licence-to-reproduce has been acquired and (the value of) the copied software is embedded

in another product which is then sold on. It is recommended to treat any software (including outsourced software) purchased for bundling or embedding into products to be sold on as intermediate consumption.

Bundled software can be invoiced separately to the customer, in which case the purchase of software can be treated like any other purchase of software by the final-use customer. It may, however, be included in the purchaser's price of the bundle in which case the software is included within the value of the bundled product, normally computers. The value of total investment is not affected by the difference in treatment.

27.4 *Maintenance and repairs*

The 2008 SNA draws a distinction between ordinary, regular maintenance and repairs of a fixed asset on the one hand and major renovations on the other. Ordinary, regular maintenance and repairs should be recorded as intermediate consumption. But major renovations that are undertaken at a time not dictated by the condition of the asset and that increase the performance or expected service life of the asset should be recorded as GFCF. However, the SNA states also that the distinction between maintenance and repairs, and gross fixed capital formation is not clear-cut (paragraph 10.45).

What makes the consideration of maintenance and repairs particularly problematic for software is that it is difficult to describe a software repair that is not an addition to an existing software system. For example, there are few equivalents to the replacement of a part, say, in conventional plant and machinery.

A repair to software systems involves a change in the configuration or code of any program, but not the replacement of a part, or repairing something that no longer works. In this way software repairs may largely be seen as improvements. Repairing "faults" introduced by bugs say, may be one example where an analogy can be made with replacements of defective parts. But other repairs or modifications, for example modifying software to provide protection from a bug, can be seen as analogous to giving a car a paint-job to protect it from unusually, but anticipated, wet weather.

Conventional maintenance (distinct from repairs), such as systems' checking, does not change the characteristics of the software and so is clearly intermediate consumption. Changes to software that extend its service life should be generally recorded as GFCF. For example, modifications to software to deal with the Y2K problem were an upgrade (involving changes to the code to record years using four digits rather than two), which extended the expected service lives of software. Modifications to software so that they can operate on a new operating system are part of the cost of adopting the latter and should be recorded as GFCF. However, frequent changes to the software to accommodate changes to the format of input data are more in the way of intermediate consumption. But how should such a demarcation be implemented in practice?

Taking account of both practical matters and conceptual principles, it is best to follow recommendation 2: IPPs are not subject to wear and tear, but they can be subject to amendment and augmentation. Substantial, planned improvements should be recorded as GFCF, while minor, unplanned improvements are better recorded as IC.

27.5 *Upgrades and outright sales of original software*

When a software original is updated or upgraded, for example the update of Word 5 to Word 6, GFCF occurs. If possible the value of the update, or upgrade, should be determined as the present value of the expected increase in income it will provide. If it is not possible to measure this, then the GFCF of the update, or upgrade, should be measured by summing the costs incurred in updating, or upgrading, the software original. This does not include the cost of creating the earlier software original (*e.g.* Word 5). The

value of updated or upgraded software is equal to the GFCF plus the depreciated value of the software before the upgrade.

Recommendation 28: Own-account software updates or upgrades should not include the value of the "original" version, and instead should only reflect the increase in value. The value of the upgraded software on the balance sheet comprises the value of the upgrade plus the depreciated value of the original version.

When a software original is sold outright the sale is recorded at the value of the actual market transaction. Most software originals are either produced for own use or to be licensed to others to use, and unless it is possible to determine with reasonable certainty that the software original was produced with the intention of sale the transaction should be treated in the same way as sales of existing assets as specified in paragraph 10.38 of the 2008 SNA. In which case GFCF of the seller of the original is negative and that of the new owner is positive.

Recommendation 29: Sales of "originals" should be treated as sales of pre-existing assets as specified in paragraph 10.38 of the 2008 SNA, unless it can be determined that they were produced for sale.

27.6 *Measurement of software GFCF*

Software GFCF generally takes one of three forms: the acquisition of licences to use software copies, the acquisition of custom-made software from a software development enterprise and the own-account creation of software originals. Separate estimates are commonly derived for each of the three, but some countries choose to obtain an aggregate of the first two.

There are two ways of deriving GFCF estimates. The first is by surveying businesses and government and asking them to report their expenditures. The second is for the NSO to derive estimates at the macro-level by using the commodity flow approach for purchased software and by deriving estimates based on the number of people developing software on own account. As discussed in Chapter I, the different approaches have their strengths and weaknesses, and recommendations 6 and 7 should be followed, *i.e.* all the approaches should be used and confronted with each other.

In 2008 another software survey of OECD member countries was conducted. It was designed to determine what changes had taken place in country practices since the 2001-02 survey and the subsequent release of the report of the OECD Software Task Force. The major difference between the findings of the 2002 and 2008 OECD software surveys was that in the earlier survey few countries reported using the demand-side approach, whereas in the later survey the majority – 15 out of the 20 countries that responded – reported using the demand-side approach. Nearly all of the 15 countries with survey data for purchases of software also derive supply-side estimates. They then go through a confrontation and balancing process. Some countries appear to rely more on the supply-side data and effectively just use the proportions from the survey estimates to allocate the supply-side aggregate to using industries and sectors, but others, such as the Netherlands, place more reliance on the demand-side data.

Consistent with their varying reliance on the survey estimates of software purchases, countries had different views on their quality, with some countries thinking they were of satisfactory quality and others having concerns. Six countries indicated that they also use surveys to obtain estimates of own account software GFCF. Three of them also use a macro method to derive estimates, but the other three rely solely on the survey estimates. The remaining 5 of the 20 countries use the supply-side approach only for purchased software and the macro approach only for own account software.

An important matter concerns the measurement of GFCF of software licences to use and the need to discriminate between those licences purchased by production units that meet the criteria of an asset and

those that do not. Essentially, the issue boils down to whether a licence to use is for more than a year (recorded as GFCF) or less (recorded as intermediate consumption). When the demand-side approach is used it is simply a matter of asking respondents to separately report their expenditures on licences of more than a year and those of a year or less. The supply-side approach, however, does not by itself permit this distinction. There are two possibilities: the first is to rely on the demand-side approach to estimate the split and the second is to obtain information from software suppliers – either directly or indirectly. For many countries, much licensed software is imported, which means that the second approach would require obtaining information from software suppliers in other countries, the national statistical offices of other countries, or some other source, such as the Gartner Group.

Recommendation 30: It is very important to distinguish between licences to use for more than a year and licences to use for a year or less. Expenditures on the former, purchased by production units and not embodied and sold on within other products, are recorded as GFCF, while expenditures on all other licences to use are recorded as consumption. Whatever approach is used it is vital that the accurate discrimination between the two should be central to measurement.

Demand-side approach

The demand-side approach for software follows the generic demand-side approach outlined in Chapter I of the Handbook. This section covers those aspects that are specific to software.

Software is ubiquitous and so the scope of a demand-side survey is the entire economy. While nearly all units purchase software a great many of them also undertake their own-account production - both components are substantial.

Purchased software

Software purchases come in a number of different forms, but it is necessary to distinguish between packaged, or ready-made, software and customised software for a number of reasons that will become clear. Units may or may not record either type of software expenditure as capital formation, but under-reporting of capital formation – from an SNA point of view – is particularly prevalent for software services. Therefore, units should be requested to include all their expenditures made on software related services, including expenditures made on original software (on which the company retains all property rights, and from which the company may make copies to be sold) but excluding all expenditures made on software to be re-sold, whether embedded in other software or in hardware.

External expenditures can be categorised as follows:

- a. Purchases of packaged software for own use recorded as capital expenditures by the enterprise. They should include single and multiple licences-to-use copies that meet the definition of an asset, *i.e.* they should include expenditures on software for which the licence agreement is for more than one year, but not for a year or less.
- b. Payments and royalties for own use of packaged software that is expensed by the enterprise. This sub-category includes all payments, including rentals and royalties for licences-to-use, for the use of packaged software (including system software) inside the enterprise that have been expensed by the enterprise, excluding expenditures on software for which the licence agreement is for no more than one year. All payments made for licences and royalties to reproduce copies to be sold as such or embedded in hardware or a software original for which the company does not have all property rights should be excluded.

- c. Payments for services related to the development of customised software for own use. They should comprise all external costs of developing customised software for own use of the enterprise, including payments for services such as R&D, analysis, design and programming or modifications to packaged software. A software original developed with a view to selling copies is considered here as “own use”. Payments for outside consultants participating in the development of in-house software are to be included whereas payments related to the development of custom software on which the company will not retain exclusive property rights should be excluded. This sub-category should not contain expenditures on software to be used for a year or less.
- d. Purchases of all property rights of software originals. This sub-category covers the purchase of all ownership rights of a software original from another enterprise, whether by outright purchase or by the acquisition of a licence-to-reproduce.
- e. Other software related expenditures for own use. They should exclude sub-contracted maintenance costs.

Own-account production of software

This category covers the costs of developing in-house software whether for internal use or for which the company intends to sell licences-to-use or reproduce. It includes internal costs of developing a software original for which the company retains all property rights and of which the unit will sell copies or embed copies in hardware or other material.

Own-account software production is usually undertaken in several stages. This production process can be outlined in the following way:

1. Feasibility analysis;
2. Functional analysis;
3. Detailed analysis;
4. Programming;
5. Tests;
6. Documentation;
7. Training; and
8. Maintenance.

Only the costs incurred in stages 2-6 should be summed to estimate the value of the GFCF of the creation of the software. The costs of the other three stages (feasibility analysis, training and maintenance) do not contribute to the basic price of the asset, and should be expensed. Note, however, that when summing costs to measure GFCF the costs of general staff training should be included. It is only the training in the use of the particular software asset that should be excluded from its GFCF.

Recommendation 31: The value of own-account software GFCF should include the costs of all expenditures on stages 2-6, above.

The calculation of total labour costs and other costs within stages 2-6, follows the rules described in Chapter I, and should be derived as follows:

Total labour costs (the product of (a) (b) and (c) below):

- a. The number of in-house staff involved in the development of software;
- b. Estimate of average percentage of time spent by in-house staff on software development, excluding maintenance and commercial tasks but including time spent on software R&D;
- c. Average compensation of staff engaged in software development, including wages, salaries, bonuses, employer social contributions and other special benefits.

Other costs (the sum of (d), (e) and (f):

- d. Overheads associated with employing the staff engaged on software development^{*}, includes management costs, training, personnel management, office requisites, electricity, rent, etc. and the use of fixed assets owned by the enterprise;
- e. Any other intermediate consumption associated with producing the software, including the licence fees for software or R&D not recognised fixed assets;
- f. Taxes associated with the cost of producing the software, such as payroll taxes^{*};

^{*}In proportion to the expenditures on software development.

Supply-side approach

The main difficulty in applying the supply-side approach to software (apart from discriminating between licences of different durations) is to avoid double-counting some flows, including sub-contracts. The method is two-fold. For purchased software (including licences to use that qualify as assets) the commodity flow method, starting with sales statistics, is used to derive a figure for purchased GFCF as a residual. For own-account software (absent by definition from sales statistics) the method is based on a macro-estimate of the cost of inputs.

Purchased software

General principles

The recommended commodity flow method for estimating GFCF in purchased software can be outlined as follows:

$$\begin{array}{r}
 \text{Estimated total gross fixed capital formation of purchased software} \\
 \text{equals} \\
 \text{Value of domestic output of software} \\
 \text{plus} \\
 \text{Imports} \\
 \text{plus} \\
 \text{Trade margins and taxes on domestic supply and imports} \\
 \text{minus} \\
 \text{Software embedded by hardware industry} \\
 \text{minus} \\
 \text{Sub-contracting flows between software companies} \\
 \text{minus} \\
 \text{Other software purchases by production units that do not qualify as GFCF} \\
 \text{minus} \\
 \text{Household consumption of games and other packaged software}
 \end{array}$$

$$\begin{array}{r} \text{minus} \\ \text{Exports} \\ \text{minus} \\ \text{Maintenance expenditures} \end{array}$$

Step-by-step implementation

The starting point in the commodity flow method is sales. To be fully applicable, sales statistics should be available in a quite detailed classification. In a European context, a four-digit breakdown of the “2008 Statistical Classification of Products by Activity in the European Economic Community” (CPA-2008) is a minimum. If available, sales data are classified by activity (main activity of the business), a preliminary step is necessary to reclassify these data to obtain sales data of software products. When implementing a supply-side approach from industry sales data, all sales of software products should be taken into account, in other words including sales of software products as a secondary activity. This is more likely to be an important issue if the survey data relate to enterprises and there are establishments producing software products for sale that are included in enterprises allocated to industries other than computer services.

The CPA-2008 and ISIC Rev.4 classifications relating to software and concordance tables with their predecessors are presented in annex H.

Step 1a: from industry (ISIC Rev.4) data to product data

If sales data originate from statistics based on business receipts classified by activity (main activity of the business), a preliminary step is necessary to reclassify the sales data to obtain sales data of *software products*. Indeed, the commodity flow approach is based on resources of the product, even if it is sold as a secondary activity. *Software publishing* (5820) is the principal supplier of packaged software and *Computer programming activities* (6201) is the principal supplier of customised software from within a country. There is a third domestic source in *Data processing, hosting and related activities* (6311) comprising *Application service provisioning*, which includes the provision of leased software from a centralised, hosted, and managed computing environment, some of which may be customised.

This step should also include another important verification for the consistency of the method: sales data should include revenues classified by businesses as *royalties*.

Step 1b: starting with CPA data

The CPA-2008 distinguishes between software services at a fine level of detail. Corresponding to ISIC Rev.4, *Software publishing services* (58.2) and *Computer programming services* (62.01) dominate the supply of software. *Application service provisioning* (63.11.13) is separately identified.

Recommendation 32: Industry sales data can only be used if they are sufficiently detailed. When implementing a supply approach from industry sales data, all sales of software products should be taken into account, including relevant businesses not classified under the category “computer services”.

Step 2: inclusion of imports to obtain total resources

For many countries imports are the major source of packaged software, and it is useful at this point to consider how the importation occurs. The Australian Bureau of Statistics (ABS) identifies a three stage process (ABS 2006). The process begins with the production of an original piece of software in country A.

The second stage can take one of two forms:

- a. the original is copied in A and exported in a ‘boxed’ format (*i.e.* disk(s), manuals and packaging) to country B, or
- b. and becoming increasingly common, a copy is sent over the Internet or on a disk to country B. A wholesaler then makes as many copies as required.

The third stage involves the distribution of the software copies through licences-to-use. In the case of 2(a), this can occur directly between a distributor in country A and the final customer in country B, or it can occur indirectly through a distributor in country B. In some instances the third stage involves the export of software copies from a distributor in country B to customers in country C.

In the case of 2(b), the terms of the contract between the software owner in country A and the wholesaler/distributor in country B could take several forms. The wholesaler may be paid a fee by the software owner to distribute copies and the owner receives the remainder of the sale receipts. Alternatively, the wholesaler may make payments to the software owner for a licence-to-reproduce, and it is the wholesaler that receives all or most of the sale receipts. The 2008 SNA recommends that in the first case the payments made by the owner should be recorded as IC. In the second case, the payments made by the wholesaler should be recorded as IC if the licence has the appearance of an operating lease. However, the SNA also recommends that if the holder of the original divests itself of part or all of the responsibility to issue and service copies under licences to use, then this constitutes the sale of part, or all, of the asset represented by the original. In which case, the payment(s) by the wholesaler represent GFCF.

Measuring international trade in software is not easy, and it is likely that Balance of Payments data will be insufficiently detailed (see below) and will have to be supplemented by data from other sources. For example, at present, royalties and licence fees in the BOP are generally not distinguished by type of product. Statistics Canada uses its annual survey on software development and computer services to derive figures for exports of computer services and exports of royalties and licence fees. A significant amount of imported royalties and licence fees are added to goods and services data to obtain an estimate of software imports. The forthcoming changes to the EBOPS classification system, outlined in section 5, should greatly improve matters in the future.

Table 6: Imports and exports of software in Canada, 1998

	Imports	Exports
Merchandise trade	1003	107
Software services	314	731
Royalties and licence fees	685	1311
Total	2002	2150

Recommendation 33: In the supply approach, import and export definitions have to be consistent with definitions of domestic supply. Both should include royalty payments and licence fees.

Step 3: inclusion of trade margins and taxes

Sales data are valued at basic prices and imports at either their f.o.b. or c.i.f. prices. To be comparable with estimates of GFCF they need to be expressed at purchasers’ prices. This is achieved by adding trade margins and taxes *less* subsidies on products (including VAT for household consumption). Only after this adjustment can the commodity-flow method (on which the supply-side approach is based) function properly. For example, in Canada trade margins and taxes on resources (sales and imports) account for 17% of the value of total supply of software products.

*Step 4: avoiding double counting and exclusion of intermediate consumption**Exclusion of intermediate consumption*

Refer to the concordance tables in annex G to see the exclusions of intermediate consumption.

As described earlier, the 2008 SNA has introduced two significant changes regarding licences to use software. First, the acquisition of a licence to reproduce may be GFCF, whereas before it was always intermediate consumption. Second, the acquisition of a licence-to-use a software copy can only be GFCF if the contract is for more than a year. The concordance tables in annex G reflect these changes and strategies need to be developed to take account of them.

There are three types of double counting to be avoided: sub-contracting, embedding of packaged software and own-account production.

Exclusion of subcontracting

Because the domestic supply of software is obtained using output data there is an inherent risk of double counting. For example, software product sales corresponding to the main activity of *Computer programming services* (62.01) or *activities* (6201) are to be classified as GFCF except for those corresponding to purchases by a non-final user of the software or if subsumed in an own account original (see annex G). Let us assume that company U, the final user of the software, orders a software product from company A, a software consultancy company. The software will cost 100. Suppose A sub-contracts 25% of the costs of the software to company B, another software consultancy company. Then total sales of software would be equal to 125, while the value of capitalizable software is 100. The 25 subcontracted to B by A is an intermediate consumption of A, and should not be capitalised.

Problems also arise for *Software Publishing*. There are three cases for which sales should not be considered as GFCF: (1) when the software product is purchased by a bundler to be included into hardware or some other equipment, (2) when the software product is purchased by another software company to be embedded in another reproduced software for resale, (3) when the software is purchased by final user households or exported.

Exclusion of packaged software purchased by hardware and software bundlers

Packaged software is bought by the hardware computer industry to be embedded in the hardware they sell. To the extent it cannot be excluded from estimates of GFCF of computer hardware, then it must be excluded from the estimates of software GFCF. If no data are available as to what proportion of packaged software is included in hardware GFCF, the 2002 OECD Software Task Force suggested that it be assumed to be 50%.

Exclusion of own-account production of software

Expenditures on software originals that are expected to be used repeatedly to produce copies for more than a year should be recorded as GFCF. In addition, the acquisition of software copies that meet the definition of an asset is also to be treated as GFCF. Hence in such cases both the own-account creation of the original and the acquisition of the copies should be recorded as GFCF. Customised software by its nature is not generally reproduced and so only the acquisition of an original is to be recorded as GFCF. This implies that double counting can only arise with respect to customised software. Hence double counting can be avoided by excluding customised software production from the estimates of own-account GFCF.

Recommendation 34: In the supply approach, double-counting of investment can be avoided by (1) excluding flows corresponding to sub-contracts, (2) excluding 50% (if no specific data) of purchased packaged software by the computer hardware industry, and (3) by excluding, in the macro-estimate of own-account production, costs of analysts and programmers corresponding to sales of custom computer programming services that have already been accounted for using the sales data.

Step 5: Maintenance

As explained earlier, maintenance is not GFCF. There is thus the need to exclude from sales data those sales corresponding to maintenance in order to derive GFCF.

Countries that have implemented the supply approach have excluded in-house maintenance costs, when building their macro-estimate of own account production. However, businesses also use external services to maintain and repair their software. Therefore, there is also the need to estimate external costs of maintenance.

For those countries using ISIC Rev.4, maintenance-type services are confined to *Computer consultancy and computer facilities management activities* (6202) and *Other information technology service activities* (6209). With the exception of that part of 6202 services deemed to be used in own account production of software, these should all be recorded as sources of intermediate consumption.

Regarding the European case, (CPA 2008), the situation is much the same.

Recommendation 35: In the supply approach, external costs of maintenance are to be excluded. When using either ISIC Rev.4 or CPA-2008 all but that part of 6202 or 62.02 providing services for own account software production should be excluded.

Step 6: Exclusion of household purchases and exports.

Exclusion of household purchases

An estimate should be made of household purchases using household budget surveys or other relevant statistics.

Games are an important part of software expenditures by households and need to be excluded if they are included in the supply estimates, above. *Software publishing* in ISIC Rev.4 (5820) and *Software publishing services* in CPA-2008 (52.80) include the supply of computer games services. In the case of the CPA, it is explicitly stated that 58.20 includes licences to reproduce and distribute. In the case of the CPA-2008, they are all included in *Publishing services of computer games* (58.21) and so they are readily excluded, but there is no further breakdown in ISIC Rev. 4.

Households also buy non-games software and that part used by individuals acting as own account workers should be recorded as GFCF, but the rest should be excluded.

Data obtained from Australia and USA seem to converge to an amount of 4 to 5% of total supply being assigned to household consumption. Canada has a similar figure with a methodological note stating that its figures exclude spending on games. France has a smaller amount (2.1%).

Recommendation 36: In the supply approach, consumption by households should be estimated through household budget surveys or other equivalent sources and excluded from sales (adjusted for trade margins and indirect taxes).

Exclusion of exports

A previous paragraph has already discussed issues regarding external flows.

Summary of recommendations for implementation of the supply approach for purchased software

The following table summarises the different steps to derive software GFCF, including specific parameter settings.

Table 7: Summary of implementation step for supply approach

ISIC Rev. 4: value of sales of capitalizable software services: <i>Software publishing</i> (582) plus <i>Computer programming activities</i> (6201) plus <i>Application service provisioning</i> (part of 6311) CPA-2208: value of sales of capitalizable software services: <i>Software publishing services</i> (58.2) plus <i>Computer programming activities</i> (6201) plus <i>Application service provisioning</i> (63.11.13), with contracts for more than one year., including royalties and licence fees, and games	A
Inclusion of imports (including royalties and licence fees and games)	B
Inclusion of trade margins and taxes on domestic supply and imports	C
Exclusion of software embedded by hardware industry (50% of purchases of packaged software by hardware industry), treated as intermediate consumption	D
Exclusion of sub-contracting flows between “software companies”	E
Exclusion of household consumption of packaged software and games if included above	F
Exclusion of exports (including royalties and licence fees and games)	G
Exclusion of maintenance	H
Total GFCF in purchased software	A+B+C-D-E-F-G-H

It is very important to note that the total value for GFCF in software should be adjusted if software already capitalised by businesses is included in total GFCF independently from this process. This adjustment is described at the end of this section.

Appendix 3 in annex G shows the concordance between CPA-2008, ISIC Rev. 4 and CPC ver.2. As can be seen, there is a many to one relationship between *Software publishing* (5820) and the CPC, but an almost one-to-one relationship between the CPA’s *Software publishing services* and the CPC. It is a similar story for computer programming: there is a many-to-one relationship for *Computer programming activities* (6201) and the CPC, and almost one-to-one relationship between *Computer programming services* (62.01) and the CPC.

Own-account original software

General principles

The OECD Software Task Force found that the GFCF of own-account original software accounted for about a third of total software GFCF. This implies that it cannot be ignored and a reasonable amount of

care should be taken in estimating it. In broad terms, own-account software GFCF can be estimated as follows:

$$\begin{aligned}
 & \text{Estimated value of own-account software production} \\
 & \quad \text{equals} \\
 & \text{Labour costs of software personnel (i.e. compensation of employees)} \\
 & \quad \text{plus} \\
 & \text{Non-labour costs of own-account software production} \\
 & \quad \text{(intermediate consumption, administrative overheads.)} \\
 & \quad \text{plus} \\
 & \text{User cost of fixed capital or gross operating surplus (depreciation only for non-market producers)} \\
 & \quad \text{minus} \\
 & \text{Costs linked to other activities} \\
 & \quad \text{(maintenance, etc.)} \\
 & \quad \text{minus} \\
 & \text{Costs linked to the production of original custom-made software and reproduction software to be sold}
 \end{aligned}$$

Explanations

To understand the estimation process used by individual countries at the macro level, the difference between *production of software personnel* and *own-account software production* needs to be clarified. Software production by software personnel refers to the total amount of software produced by all the software personnel, which includes both software to be used internally (own-account software) and software to be sold. Own-account software production refers to the total amount of software produced in-house by software personnel for internal use. It thus excludes the software production linked to software to be sold. It is important to note here that original software for reproduction, such as Microsoft's Windows corresponds to software to be used internally. Only reproductions of Windows are sold, not the original.

Therefore, in order to estimate own-account software production carried out by software personnel, a "sales adjustment" needs to be made to exclude market activities (i.e. sale of original custom-made software and sale of reproductions). This adjustment allows that no double counting is recorded under the supply approach, because software sold has been already accounted for using sales data.

The production of own-account software is measured as the sum of production costs. These costs consist of compensation of employees, administrative overheads, intermediate inputs, indirect business taxes (e.g. payroll tax), user cost of capital, etc.

Labour costs

The labour compensation costs of software personnel can be measured by multiplying the number of the relevant labour force by their average compensation. Average compensation should be derived using the national accounts measure of compensation of employees. It is recommended that the number of software personnel should be broken down by group of economic activity, including the government sector, and particularly the ISIC Rev.4 categories *Computer programming activities* (6201) and *Computer consultancy and computer facilities management activities* (6202).

The number of software personnel can be estimated either by direct business surveys or employment data by occupation, but most countries do the latter. The appropriate identification of software personnel is not straightforward, however. The OECD Software Task Force recommended that in the absence of direct survey data on the number of software personnel, employment data by occupation should be used and limited to the number of computing services department managers and computing professionals according to International Standard Classification of Occupations 1988 (ISCO-88), codes 213 (computing professionals), in the hope that the contribution of other occupation codes associated with computer programming was insignificant. The reason for this assumption was that there was a lack of information as to how much time other software personnel spent on software development and the belief that it was not substantial.

A more recent consultation of key firms and institutions in the software industry by the United Kingdom's Office for National Statistics (ONS) has found that a wider range of occupations should be considered. Although software professionals constitute the most important occupational group, significant contributions are also made by computing services managers (ISCO-88, code 1236), computer assistants (code 3121), computer equipment operators (code 3122) and data entry operators (code 4113). Since this study included discussions with important firms in the software industry, *e.g.* CISCO Systems and IBM UK, and the broadening of the scope was found by the ONS to increase estimates of own-account software GFCF by about 20%, the broadened scope is recommended.

As the multiplication of the number of software personnel by their average compensation provides their total compensation, adjustments have to be made to obtain the labour costs of own-account software production. This can be done by subtracting the labour compensation that is not linked to own-account software production from the total labour compensation of software personnel. These adjustments are made based on data on the working time of the labour force classified by industry of activity by ISCO code. In a first step, the working time of the software personnel that is spent on the production of original custom-made software and reproduction software that are to be sold should be excluded, leaving the working time for the production of own-account software and of originals for reproduction. A second adjustment has to be made for the working time of software personnel linked to other activities such as system repair, maintenance of computer systems, etc. This part of their working time has to be estimated and deducted from their total working time. The UK ONS survey found the following approximate percentages of time spent on software development by software personnel occupation group. The data are reported in terms of the UK Standard Occupational Classification, along with closest ISCO-88 equivalents. The respondents to the UK survey reported that about 70% of the time of software professionals (213) was spent on software development, but the UK decided to adopt 50%, in line with the recommendations of the report of 2002 OECD Software Task Force.

Table 8: UK occupational codes used in estimating labour costs of own account software production

ISCO-88	UK SOC	Occupation	Proportion (%)
1236	1136	Information and communication technology managers	15
213	2131	IT strategy and planning professionals	35
213	2132	Software professionals	70 (50)
3121	3131	IT operations technicians	20
3122	3132	IT user support technicians	15
4113	4136	Database assistants/clerks	5
213	5245	Computer engineers, installation and maintenance	5

The recommendations of the 2002 OECD Software Task Force were based on US experience. The US has adopted a 50% deduction rule for the time spent by software professionals on tasks other than software development. The 50% share originates from a 28-year old study on the share of software development and maintenance costs in 487 business organizations reported by Barry Boehm (Boehm 1981). The detailed

shares are shown in the Table 9. The categories that are classified as software investment are in bold italics.

Boehm identified 43% of the time spent by programmers and systems analysts was on development, but for national accounts purposes some of the items he identified as maintenance (shown in bold in the table) should be recorded as development, which brings the total to 62%. Nevertheless, a 50% share was chosen to emphasize the approximate nature of the estimate. The 50% share is also based on anecdotal evidence that the share has diminished with the growing importance of personal computer and pre-packaged software. So far, no recent study on the matter has been identified. The 50% deduction rule is also adopted in Canada, France and Italy.

Table 9: Time spent on software development and maintenance costs by programmers and systems analysts in the US

<i>Development</i>	<i>43 per cent</i>
Maintenance	
a) Emergency program fixes	6 per cent
b) Routine debugging	4 per cent
c) Accommodate changes to input data, files	8 per cent
d) <i>Accommodate changes to hardware, operating systems</i>	<i>3 per cent</i>
e) Enhancements for users:	
<i>New reports</i>	<i>8 per cent</i>
<i>Added data for existing reports</i>	<i>6 per cent</i>
Other	7 per cent
f) Improve documentation	3 per cent
g) <i>Improve code efficiency</i>	<i>2 per cent</i>
h) Other	8 per cent
Other	2 per cent

Statistics Netherlands conducted an analysis in order to compare the result of the labour costs of own-account software production derived from two different data sources: direct survey (Automation Survey) and a labour survey (employment and wages by occupation). The main conclusion is that the correction factor of 50% leads to an overestimate of the labour cost of own account software production.

Recommendation 37: If a country does not have reliable data on the share of time spent on the various tasks of computer professionals, the share should be assumed to be no more than 50% in calculating the labour cost of own-account software production.

Non-labour costs

As direct data on non-labour costs of own-account software production are hardly available, they have generally to be estimated based on the relationship between labour costs and non-labour costs of relevant industries. The data for the relationship should generally be derived by survey or census data for computer

services industries (if possible, custom software developers would be preferable). The calculated ratio of non-labour costs to labour costs is quite different across countries. This is mainly due to the varying availability of data on the cost structure of related industries. Some countries have data at a detailed level of computer services industries, but others might have only data on service industries as a whole.

It is reasonable to assume that the cost structure of own-account software production is similar to the custom software development or contract software programming industries. These industries tend to be more labour intensive than service industries as a whole. However calculated, the ratio should be adjusted to exclude any double counting of external costs that would have been already recorded in the other branch of the supply method, covering purchases. This is due to the fact that sales of programmer services included in the process of production of a final user's in-house software are to be recorded directly as investment. At the same time, the above process of estimating non-labour costs using the structure of the computer software industry, implicitly includes a mark-up for these external costs, because the computer software industry also purchases software services for its own use. There would therefore be a double-counting element if these costs were included both as purchases and, implicitly, in the mark-up process used in adding in non-labour costs. That is why a downwards adjustment of the ratio is recommended. Furthermore, it is recommended that allowances for administrative overheads should be included to take account of their contribution to the process of software production.

Cost of capital services

The cost of the capital services provided by all the non-financial assets used in the production of own-account software should be included. It is recommended that the ratio of the gross operating surplus to labour costs in the custom software development or contract software programming industries be used to make the imputation – see section 2.2

Sales adjustment

As described above, it is necessary to make an adjustment to exclude the costs of producing custom software to be sold, as not all software personnel produce own-account software. Many of them, especially in the computer services industry, are involved in the production of software to be sold, and this activity should not be included in the estimation of own-account software production. Ideally, surveys of the computer services industry should be undertaken to determine what proportion of their software personnel are used to produce custom software and what proportion is used to produce originals for producing copies and originals for internal use. In the absence of actual data, it is worth noting the experience and practice of the US (reference) and UK.

In the US, a sales adjustment of $2/(\text{proportion of software professionals to total employment} \times 100)$ is made in the custom software industry.

Multiplicative model

The additive model described above for deriving macro estimates of own-account software GFCF may not be the best way of deriving estimates in practice. It may well be reasonable to assume that direct labour costs (*i.e.* compensation of employees) are directly proportional to some of the other factors in the model, *e.g.* costs other than direct labour costs (such as management, taxes, intermediate inputs, and the capital services from fixed assets) and the sales adjustment. If so, a multiplicative model, as shown below, or a mixed multiplicative-additive model should be used.

Own-account software GFCF

equals

Wages and salaries of software personnel
 multiplied by
 Mark-up to take account of other labour costs
 multiplied by
 Adjustments for time spent on other activities
 multiplied by
 All overheads (management overheads, intermediate inputs, user cost of fixed capital)
 multiplied by
 Sales adjustment

R&D and software

The 2008 SNA recognises R&D expenditures that meet the general definition of an asset as fixed assets. This change occurred after the 2002 OECD Software Task Force report and after the UK study described above. As such the OECD Software Task Force recommended that all costs associated with software R&D should be included in estimates of own-account software GFCF, and recommended that they should be capitalised as they occurred. With the recognition of R&D as assets however only the capital services provided by R&D should be included in these costs. This matter is addressed in section 1.4 and recommendation 3 and in the R&D chapter.

Further adjustments ensuring consistency of national accounts

When the estimates of gross domestic product using the income approach are based directly or indirectly on business reports, an adjustment has to be made when compiling gross operating surplus to ensure consistency of the national accounts because the “supply approach” leads to a significantly different breakdown between current expenses (intermediate consumption) and investment (gross fixed capital formation) than in the business reports. These adjustments should be based on the difference between the independent “supply approach” estimate of gross fixed capital formation and what is declared capitalised by businesses. In order to compile this difference, surveys should continue to monitor capitalised software investment as they are recorded in business accounts.

Prior to the decision to implement the SNA recommendation on capitalizing software, and in order to be fully consistent with SNA, all software expenses should have been treated fully as intermediate consumption and not gross fixed capital formation. As a result, the above adjustment to corporate profits should not be a new feature in the process of compilation of the national accounts. In other words, before the implementation of 1993 SNA, corporate profits should have been adjusted by adding to intermediate consumption the software “wrongly” classified as investment in the business accounts. In parallel, gross fixed capital formation reported by businesses should have been diminished by the same amount.

Another source of double-counting of software capitalisation is the use of specific business reports. In applying the supply-side approach, double-counting of software investment already included in national accounts (sometimes under “hardware”) occurs when the general process of estimation of gross fixed capital formation uses business reports which include software capitalised by business since sometimes this software, even bought separately from hardware, is included as hardware. That is why the built-in capitalised software already included as hardware in business reports has to be deducted from hardware investment. A possibility to do this is to compare the reports from respondents to the relevant survey (e.g. on capital expenditures) who declared software capitalisation to the software investment resulting from the

commodity flow method. The resulting ratios can be applied to adjust the hardware data in order to avoid this kind of double-counting.

28. Databases

The 1993 SNA described the treatment and measurement of databases as a special case of software and recommended that only large databases should be capitalised. There was a good deal of difficulty in implementing this recommendation for a number of reasons: there was no precise definition of what a database was, how should the qualification “large” be interpreted, should the value of information stored on a database be included in its value or not and, hence, how should a database be valued in general. As a result, many countries did not capitalise databases at all or not in an internationally comparable way. An OECD survey of Member countries in 2004 (OECD 2004) found that of the 13 countries who responded 5 said they excluded databases from their estimates of GFCF and the remaining 8 said that they included them in principle, but the values were not separately identifiable.

All of these issues were addressed by the Canberra II Group, and its proposals for addressing the above deficiencies led to changes reflected in the 2008 SNA.

28.1 *Definition and treatment*

The 2008 SNA identifies databases as a separate sub-category of the asset category “software and databases”, and in paragraphs 10.112 to 10.114 clarifies what databases are and how their value should be determined.

10.112 Databases consist of files of data organized in such a way as to permit resource-effective access and use of the data. Databases may be developed exclusively for own use or for sale as an entity or for sale by means of a licence to access the information contained. The standard conditions apply for when an own-use database or a purchased database or the licence to access a database constitutes an asset.

10.113 The creation of a database will generally have to be estimated by a sum-of-costs approach. The cost of the database management system (DBMS) used should not be included in the costs but be treated as a computer software asset unless it is used under an operating lease. The cost of preparing data in the appropriate format is included in the cost of the database but not the cost of acquiring or producing the data initially. Other costs will include staff time estimated on the basis of the amount of time spent in developing the database, an estimate of the capital services of the assets used in developing the database and costs of items used as intermediate consumption.

10.114 Databases for sale should be valued at their market price, which includes the value of the information content. If the value of a software component is available separately, it should be recorded as the sale of software.

This definition implies that all databases holding data with a useful life of more than one year should be recorded as fixed assets providing they meet the general definition of an asset (*i.e.* are expected provide benefits to their owners and over which ownership rights are exercised). Databases created on own-account and those for sale are included in the asset boundary if they meet this criterion notwithstanding their size or their type. The value of the DBMS will normally be recorded elsewhere as software. The definition also implies that the scope of databases should not be limited to specific types of databases nor to databases created by specific activities and that the reference to “large” mentioned in 1993 SNA no longer applies.

Databases produced for outright sale should be valued at their market prices. Likewise, expenditures on licences to use databases should be recorded at their market prices and recorded as GFCF if the licences meet the definition of an asset, or as intermediate consumption if they do not, in the same way as software licences, see above.

Most databases are produced on own-account, either for internal use only or for distribution via licences-to-use or reproduce. The GFCF of those databases that satisfy the definition of an asset has to be estimated by summing costs in the same way as software. However, there is one important difference between a piece of software and a database (excluding the DBMS); unlike software, the data on a database that meets the definition of an asset does not require maintenance. The value of data may decline over time due to obsolescence but it does not decline due to deterioration, and the cost of updating a database that qualifies as a fixed asset should be recorded as GFCF.

The recommendation in the 2008 SNA not to include the cost of obtaining information when summing costs to value database GFCF was made for measurement reasons and because otherwise the door to the capitalisation of knowledge in general would have been opened indirectly. In addition, the capitalisation of knowledge would create an inconsistency in the SNA, because its capitalisation would depend on how it was stored. If the knowledge was stored and embodied in a database it would be capitalised, however, if it was stored elsewhere, *e.g.* on paper files, it would not be capitalised. In addition, the data/information may already be recorded in the accounts as fixed assets, in the category “entertainment, artistic or literary originals”, or they may not be, *e.g.* paper records.

A key question is which information provides services for more than one year since it is the length of the expected working life of the data/knowledge that determines whether the database should be recorded as a fixed asset. A good indication that it should, is if either of the following two conditions is met:

- a. a typical datum is expected to be stored on the database for more than a year, or
- b. if a typical datum is expected to be updated and replaced within a year on the principal database, then it will be archived on a secondary database.

Recommendation 38: a database should be recorded as a fixed asset if a typical datum is expected to be stored on the database, or archived on a secondary database, for more than a year.

28.2 *Measurement*

Most creation of databases occurs on own-account, either for internal use or for sale via licences-to-use. The rules for determining whether the purchase of a licence should be treated as GFCF or intermediate consumption are the same as those for software.

A feature of most, but not all, databases is that they are frequently updated, and external users of a database pay for a copy that is frequently replaced with an updated version. This is the case for many statistical databases, for example. Access to frequently updated databases is generally obtained by annual subscription and users are undertaking consumption of database services rather than fixed capital formation. There are exceptions, such as the sale of population census data on a CD-ROM by a national statistical office and for which the database may be used for five or ten years. However, there is a dearth of information as to how significant the GFCF of databases by purchase is.

As for software, estimates of database GFCF can be derived using the demand-side and/or supply-side approaches, at least in concept. But unlike software, purchases of databases or database services that qualify as GFCF is thought to be very minor, with examples such as population census data being very much the exception. It is therefore recommended that the focus should be on measuring own-account

database GFCF and that purchases of databases or database services only be recorded as GFCF on an exceptional basis, if and when such sales come to light.

It has been difficult to determine how great expenditures on database creation are. There is no particular database industry and CPC Ver. 1.1 did not provide an adequate set of categories that covers databases without including too many other things besides. With the introduction of CPC Ver. 2, this has changed and now there is a single category “Original compilations of facts/information” (83940) that relates to databases. How well data can be collected for this category remains to be seen.

Demand-side approach

The scope of an “ideal” survey should be all units in the economy. There would be considerable advantage in combining it with a software survey or more general survey, as this could minimise costs and may help avoid including the value of a DBMS in the value of a database created on own account. As for software, the survey should distinguish between external costs (expenditures) related to databases for own final use and internal costs of in-house database creation. In addition, the survey should ask for the company’s own estimate of its capitalised databases, if any.

Concerning external costs (expenditures), businesses should be requested to include expenditures made on original databases (on which the company retains all property rights, and from which the company may make copies to be sold) but excluding all expenditures made on databases to be re-sold and all payments for data base management software. External costs should exclude all payments for data or other information to be incorporated in a database, but include the cost of any services entailed in preparing or loading the data into a database.

Costs of in-house database creation

This category covers the internal costs of developing a database original on which the company retains all property rights and of which the company will sell copies or embed copies in hardware or other material. It also covers databases developed for internal use. But it does not include the costs of creating databases intended for outright sale. It comprises the costs of utilising a DBMS (but not the cost of the DBMS itself) and loading data/information into a database, including updates. It requires the calculation of total labour costs and other costs as follows:

Total labour costs:

- a. The number of in-house staff involved in the specification of the DBMS and loading data/information into it, including updates;
- b. Estimate of average percentage of time spent by in-house staff on database tasks;
- c. Average compensation of the staff engaged in database creation, including wages, salaries, bonuses, employer social contributions and other special benefits.

Other costs:

- d. Overheads associated with employing the staff engaged on database creation and updating*, includes management costs, training, personnel management, office requisites, electricity, rent, etc. and the cost of using the enterprise’s fixed assets;
- e. Any other intermediate consumption associated with database creation, including the costs of software not recognised as a fixed asset;

- f. Taxes associated with the cost of database creation, such as payroll taxes^{*};

^{*}In proportion to the spent on database creation.

Supply-side approach

The method to be used is the same as that for own-account original software (see above). In the absence of any data on the proportions of time spent by occupation groups on database creation, it is recommended that the direct labour costs be determined by the time spent by database assistants/clerks (ISCO-88 4113) not allocated to software production. Non-labour costs, the user cost of fixed assets should be derived in the same way as for software.

Not all database creation qualifies as GFCF. In the absence of any information on the proportion that does, it is recommended that it be assumed to be 50%.

29. International trade in software and databases

The need for valid, detailed and international comparable data on trade in services has increased as its share in international trade volumes has grown over recent years. The key interest focuses especially on data on trade in software since this is assumed to be extensive and very dynamic.

However, measuring international trade flows of software can be very difficult. These difficulties result from the fact that software may be traded on a variety of media, both tangible and intangible, and by a variety of means. Moreover, software sales may take the form of licences to use or reproduce software, which may or may not be accompanied by a physical supply of software.

Software is often bundled with hardware or other computer or consultancy services. Computer software is only one of a number of so called digitized products along with, for example music, film, data, TV programmes, news and literature that may be regarded as presenting analogous measurement problems regarding international trade.

International trade is for practical reasons partitioned into goods and services more rigidly than production. Eight examples of ways in which software can be traded as goods or services internationally are distinguished in the following (there may be more).

- a. The most straightforward case is where packaged software is traded with manuals on a physical disk, *e.g.* a CD-ROM. However, valuation is sometimes a problem here, if it is based on the medium rather than the software content and/or the extent of the user licence.
- b. Software may be installed on equipment or machinery, *e.g.* a PC. The software traded then may be counted simply as trade in that type of equipment. Both case (1) and (2) are treated as trade in goods.
- c. A single (physical or online) copy of some software may be sold to a foreign firm which pays a licence fee to make further use of it. The licence payments are counted in trade in services but will not be separately identified as software in the current international classifications. It should be noted that it is not uncommon for large firms/organisations to renegotiate the licence-to-use agreements and ensuing payments can be divorced from any physical supply of software.
- d. A single (physical or online) copy of some software may be sold to a foreign firm, possibly an affiliated firm, under licence to reproduce/sell further copies within certain (geographical/numerical) limits or bundle the software with hardware or software for resale. The

royalty payments are counted in trade in services but again will not be separately identified as software in the current international classifications.

- e. Traded customised software, if sold in physical format, is likely to be counted as trade in goods in the Customs reports, but may transferred to trade in computer services, following BPM6 guidelines, or possibly to purchase of assets depending on the nature of the transaction.
- f. Software may be traded internationally online and in such a case it will by default not be counted in Customs reports. BPM6 recommends that downloaded software should be treated as computer services.
- g. Customers can subscribe to software services where the software is frequently updated, e.g. anti-virus software or databases, and access updates online (possibly downloading all or part).
- h. Finally, software may be sold internationally from one firm to an affiliated firm within the same multinational. This is likely to form a significant part of trade in software. Here, there is no guarantee of uniform treatment and although this may be treated as in the cases above another possibility is that such transactions may be treated as internal computer services, royalties, classified as miscellaneous management charges, trade in services with related enterprises, goods trade, or in extremis not recorded at all. There is also no guarantee of a market price valuation.

This section gives a definition of international trade in software and addresses its measurement whether or not particular sets of transactions are regarded as part of capital formation. The aim is to identify areas where measurement could be improved and to make recommendations on improvements to classifications, reporting practice and further work, in particular on the measurement of trade in software goods, of computer services and of software royalty payments. Furthermore, it addresses both specific problems concerning software delivered online and the borderline between merchandise trade and trade in services.

It seems probable that nearly all expenditures on database services are consumption and not capital formation.

29.1 Concepts and classification issues

As described in chapter I the product “computer software” is not currently well identified in current international trade codes or balance of payments (BOP) items, although BPM6 and the 2010 IMTS and EBOPS classifications will improve this situation somewhat. Rather than repeat the discussion of software in chapter I, this sub section focuses on issues related to international trade in software not covered earlier.

For merchandise trade there are a number of categories of HS products which may approximately relate to trade in software goods. The IMTS in its paragraph 27 sets out guidelines for the recording of software goods in international trade. It describes goods that are carriers of software within HS heading 85.24. *“This category includes, for example, ... packaged sets containing diskettes or CD-ROMs with stored computer software and/or data developed for general or commercial use (not to order), with or without a users’ manual. However ... diskettes or CD-ROMs with stored computer software and/or data developed to order are to be excluded from international merchandise trade statistics.”* IMTS goes on to add that where these goods are carrying software and/or data developed to order they should be treated as part of trade in services. It should be noted that the HS codes do not distinguish media carrying customised software from packaged software. In the 2007 update of the HS, HS07, 85.24 is replaced by 85.23 and the situation is even worse, as there is no distinction between media with and without anything recorded on them.

For online delivery of standard (*i.e.* packaged, not customised) software or databases, no clear classification guidance currently exists as is the case for some other digitized products. However, BPM6 will recommend that downloaded software be recorded as computer services.

There are some points of difference between the basic trade data and the basis on which investment assets are measured in the SNA. First, the BOP trade series (as per BPM5) make no distinction as to the length of time traded goods or services are used, whereas the SNA recommends that only software for use in production for more than one year should be recorded as a fixed asset. Second, the BOP records (by exception) repairs in computer services, rather than in goods, without clearly demarcating the extensiveness of the repair, whereas the SNA includes improvements to existing fixed assets that go beyond ordinary maintenance and repair in GFCF. This difference is remedied in BPM6.

29.2 Conclusions and recommendations

Current international trade and balance of payments classifications and statistics are not as helpful as they could be in identifying international trade in computer software. Only a few countries appear to have access to a satisfactory set of data concerning trade in software. Supply-use tables should be made more consistent with trade flows in software.

That said the improvement to classification systems in BPM6, the 2010 IMTS and EBOPS classifications should lead to improvements in this area.. Changes to the HS system are also welcome however and the development of a standard international grouping of HS codes that represents trade in software goods (HS96 codes 8524.31; 8524.40; 8524.91 and 8524.99) to improve international comparability is encouraged. .

In addition, two main areas for follow up work are identified where questions are unanswered and it appears premature to make any specific recommendations is needed. First, research should be undertaken into how software goods are valued and whether and how countries coordinate software measurement (valuation) in trade in goods and services to ensure a standard allocation, full coverage and avoid double-counting. Second, the online sale/purchase and delivery of software to/from other countries needs to be further investigated, probably through Internet use and e-commerce surveys. A further task is to identify the CPC Ver. 2 codes associated with the relevant HS codes mentioned above.

30. Prices and volumes

When deriving volume estimates of software and databases it is advisable to decompose software into three components: packaged (or off-the-shelf), custom-made and own account, and to deflate them and databases separately. There are several reasons for doing this. First, the three components of software and databases vary in the extent to which price data are available to compile price indices. Second, it is likely that their prices and volumes grow at different rates, particularly between packaged, the other two software components and databases. Third, despite the previous point, price indices for packaged software may be used to construct price indices for the other two software components if more appropriate price indices are unavailable. Fourth, volume estimates of the items are useful indicators in their own right.

Packaged software is purchased on a very large scale, generally via licences-to-use, and there is an abundance of price data available. The challenge is to construct price indices free of the effects of changing specifications and any other aspects of quality change. With ever larger numbers of copies of popular software being sold, growing economies of scale allow prices to fall. Custom-made software is also sold on the market, but each custom-made software product is a one-off, which presents an obvious problem for compiling price indices. Although each custom-made product is different, different products may share common components, or a strategy used to develop one product may be able to be used for another. This

not only suggests a possible way of compiling a price index, but also suggests means by which productivity gains could be made that would put downward pressure on prices.

The 2002 OECD Software Task Force found that the deflators used to derive volume estimates of software GFCF varied enormously between member countries. This largely reflected the fact that many countries did not have suitable price indices and used the price indices of other goods and services as proxies. Partly as a result of the OECD Software Task Force's report and partly as a result of the EU decision to make it mandatory for its members to adopt more appropriate deflators those differences are being reduced. In making its decision, the EU provided indications of what form suitable price indices might take. But these are of a general nature, and reflect the fact that there is still more work to be done in determining the best way to derive suitable price indices for these products in practice.

One thing that is clear is that the best results can be achieved in an input-output framework. This ensures that solutions made in the deflation are internally consistent. For many countries a significant share of purchased software is imported. If prices and volumes on the use side are consistent with imports, then errors, at least at the GDP level, will not be very significant.

There are two particular features of software GFCF that make the derivation of suitable price indices challenging. First, there are rapid quality and specification changes, and, second, price data are only readily available for purchases of packaged software. This section describes how price indices for software and databases should or could be compiled. It distinguishes between packaged software, customised software, own-account software and databases. When the most desirable way of compiling price indices may only be possible in the long term, then second-best, short-term solutions are given.

30.1 *Constructing price indices for software and databases*

Packaged software deflators

Generally, all OECD member countries should develop price indices for packaged software in the long term. These should cover software acquired by both businesses and households (including games) and adequately take into account qualitative changes of software. As shown above, developing an unbiased index is difficult and adjustments might still be needed. If that is the case, adjustments should be based on available objective data and made transparent to users. Improvement of the comparability with other countries is an important criterion in the adjustment procedure.

The US has been the leader in developing price indices for packaged software. The US producer price index for packaged software is compiled by the Bureau of Labor Statistics (BLS) and was first published in December 1997. It is based on a survey of producer selling prices, *i.e.* at the first line of distribution, collected from a sample of manufacturers of packaged software. The BLS collects price quotes from both the Original Equipment Manufacturer (OEM) and finished goods channels, and for full versions and upgrades.

The methodology of the BLS price index for packaged software is a fixed basket matched-model Laspeyres price index, with plans to update the weights every five to seven years. Because of the bias in price changes measured by matched-model price indices, the US Bureau of Economic Analysis (BEA) began, in 2000, to make an adjustment to the BLS packaged software price index. This adjustment is based both on a matched-model price index for spreadsheets, word processors, and databases (Oliner 1994) and on a BEA hedonic price index for spreadsheets and word processors. The average annual difference between these two sets of price indices over the 1985 to 1993 time period is – 6.3 percent. The BEA calculates its bias adjustment as one-half of this difference, or – 3.15 percent. Self-evidently, use of mechanically adjusted price index is not an acceptable solution in the long term.

Nevertheless, the BEA's price index is recommended for use in the short term, because on the one hand the US has a dominant share in the market and on the other hand the use of the same index ensures the best comparability between countries. For use outside the USA, the US price index could be adjusted on the basis of either changes in exchange rates or purchasing power parities (PPPs) and it should reflect different timings of releases of new software in the USA and in the country where the modified US price index is used. The problem with using exchange rates is that they can be volatile, and the software supplier may not adjust the prices of imported software in accordance with them for practical as well as competitive reasons. The problem with PPPs is that they are unlikely to be available in sufficient detail and they are only collected at infrequent intervals. In between times they are extrapolated using GDP IPDs.

One possibility is to contact major software importers and ask them how they set and adjust their prices, and at the same time ask them what the usual lag is between software released in the US and software released in the home country.

Although prices for domestically-produced software do probably not develop in the same way as prices for imported software, it is better to use the US BEA price index, appropriately adjusted, than a price index not directly related to software. It is suggested here that the BEA index be adjusted by the relative inflation rate between the home country and the US (preferably producer price index for the home country vis-à-vis producer price index for the US).

Customised software deflators

The standard price index techniques referred to above cannot be applied to customised software, at least not in a straightforward way, because each product is unique. Methods for constructing price indices for unique products are described in the *2004 Producer Price Index Manual*. They include model pricing, repeat recent real sale, specification pricing and component pricing. Of these possibly the first, model pricing, is the best possibility. Model pricing involves asking a producer to specify a notional product based on recent orders. For each period the respondent is asked to supply a hypothetical price. Model specifications need to be changed over time to reflect changes in the market.

For customised and own-account software the PPI Manual suggests function point analysis as a potential means of constructing price indices. The function point metric was devised as a means of measuring software size and productivity. It uses functional, logical entities such as inputs, outputs, and inquiries that tend to relate more closely to the functions performed by the software as compared to other measures, such as lines of code. Basic function points are categorized into five groups: outputs, inquiries, inputs, files, and interfaces. A function point is defined as one end-user business function, such as a query for an input. Determining the size of a software product involves counting the number of each type of function point and weighting them. This is a time-consuming business and there is the question of whether two trained analysts would make the same count for a software product. Nevertheless, there are a large number of software enterprises and others engaged in function point analysis and efforts are being made to address the difficulties just described (Carnegie Mellon Software Engineering Institute 2007).

At the time of writing we are unaware of any satisfactory price index that has been compiled for customised software, and so it is premature to make a recommendation as to how such price indices should be compiled, but model pricing and function point analysis look to be the best prospects.

The US derives its price index for customised software as a weighted average of its packaged software index and an input price index based on the costs of producing software (wage rate indices, PPIs for intermediate inputs, etc.). Weights of the two indices are arbitrarily defined: for packaged software 25% and input price index 75%. The rationale is that some productivity growth can be expected in the production of customised software, but not at the same rate as for the production of packaged software. At

least two other countries (Australia and Canada) have adopted the US approach. To do so, countries should take a weighted average of the US packaged index, adjusted for differential inflation rates (see above), and an input cost index compiled for their own country.

Another second-best approach is to adjust the input price index for customised software using estimates of multi-factor productivity growth in related industries where it is observable.

Own-account software deflators

In the long term, when price indices for customised software become available, it would be reasonable to use them for own-account software production. In the interim, countries could follow the same approach they use for customised software.

Database deflators

Databases are generally heterogeneous products with a small market since most databases are made for in-house purposes. This makes it difficult, if not impossible, to develop a true output price index. We must therefore consider second-best alternatives; there appear to be three. The first is to compile an input price index, but this would imply zero productivity growth. The second is to adjust the input price index by assuming MFP growth in database production is similar to some other industries. The third is to use a price index of some related activity for which there is a price index of reasonable quality.

31. Capital measures

The capital measures referred to in the 2008 SNA comprise gross fixed capital formation, capital services, net capital stock and consumption of fixed capital. Their definitions and the roles they play are all described there. Methods for estimating these measures are discussed in chapter I and are also the subject of the new edition of the OECD manual *Measuring Capital*. Note should be taken of recommendation 15: when using the PIM, it is important to have reasonably accurate service lives. The geometric model has a number of advantages and should be used unless there are strong conceptual or practical objections.

Nearly all countries derive their estimates of capital services, net capital stock and consumption of fixed capital using the perpetual inventory method (PIM). As its name suggests, the PIM involves aggregating GFCF over time, but allowing for declines in efficiency and value until assets reach the end of their service lives and are retired. The PIM is applied to groups of assets, generally at the most detailed level at which GFCF data are available.

The key parameters in the PIM are the expected service life of a group of assets of a similar type, the rate at which its productive capacity, or efficiency, is expected to decline as it ages and the rate at which its value is expected to decline as it ages. The last two are interdependent and their relationship hinges on a discount rate. Not all assets within a group can be expected to have exactly the same service life, and so a probability distribution function is usually specified. The following section addresses the matter of setting values for these parameters for software and databases.

31.1 Service lives

The most important PIM parameter is the service life. Specifying a service life of 10 years rather than 5 years would make a huge difference to the estimates of the capital measures. Net capital stock would be approximately double, and with a typical scenario of strong growth, consumption of fixed capital would be appreciably smaller. It therefore deserves a good deal of attention. There are several ways of obtaining estimates of service lives, they include: surveying software users, surveying software suppliers and consulting software consultants.

Surveying software users

This could entail asking software users what their expectations are of the service lives of the different forms of software they have acquired in the latest year, *i.e.* packaged, customised and own-account. Alternatively, they could be asked what the service lives have been of recently retired software products. A natural place to pose such questions would be in a demand-side survey.

Surveying software suppliers

Most packaged software is acquired by licences-to-use. Software suppliers can be expected to have records that may indicate the length of time of licences. But can they differentiate between business and household users?

Consulting software consultants

There are many IT consultancy firms, and some may have conducted studies into this matter. They generally do not provide such information free of charge, but it could still be a cost-effective solution. They may also be able to supply information on databases.

31.2 Country practices

Many countries currently do not derive estimates for capital services and they do not specify an age-efficiency function. But they do specify an age-price function which determines how the value of an asset, or group of assets, declines as it ages. The 2002 OECD software task asked Member countries to report the service life assumptions they used, and the functional forms of the age-price function and the retirement distribution function they use. Table 10 below presents the results.

**Table 10: Estimating the capital stock of software, PIM parameters
Responses to Questionnaire, 2002**

Country	Service lives		Age-efficiency or age-price function	Retirement distribution function
	Own-acc't & Customised	Pre-recorded/ packaged		
Australia	Pre 89/90 -8 Post 89/90 6	6 4	Hyperbolic for age efficiency function	Skewed retirement for packaged & other
Canada	5	3	Straight line	Truncated normal
Czech Republic	5	4.5	Straight line	Lognormal
Denmark	6 ^a	4 ^b	Straight line	Winfrey S3
Finland	5		Straight line	Skewed Weibull
France	5		Straight line	Lognormal.
Italy	5		Straight line	Truncated normal
Japan	5		Straight line	None
Netherlands (c)	3		One hoss shay for age-efficiency function	Weibull
Spain	4		Straight Line	Delayed linear
Sweden	10 ^a	5 ^b	Geometric	None
United Kingdom	5		Straight Line	Normal
United States	5	3	Geometric	None

(a) Own-account software only; (b) all purchased software; as of 2008.

With the exception of Sweden, most respondent countries reported service lives of approximately 5 years. A few countries specify service lives for customised and own-account separately from packaged, and invariably specify a shorter life for the latter. Given the high cost and specialised nature of customised and own-account software this is only to be expected. One country, Australia, indicated that it had found that service lives had declined over time, and had set shorter service lives from 1989-90.

Australia uses a hyperbolic age-efficiency function and derives corresponding age-price function by assuming a real 4% per annum discount rate. The Netherlands (as of 2008) follow a similar procedure but assume a one-hoss shay age-efficiency function. The other responding countries mostly reported using straight line depreciation, *i.e.* the age-price function is assumed to decline linearly. Two exceptions were Sweden and the US who reported using a geometric age-price function.

Nearly all those countries who did not report using a geometric age-price function reported using a retirement distribution function, but with little commonality.

Another survey of OECD countries was conducted in 2008, and the results are shown in table 11 below. This survey only sought service life information.

Table 11: Software service lives, 2008

	own account & customized	purchased	total
Austria			30% depreciation rate
Belgium			3
Canada	5	3	
Czech Republic			4.5
Denmark	6	4	
Hungary			5
Israel	5	3	
Japan			5
Netherlands			3
New Zealand			4
Poland			8
Slovak Republic			5
Spain			4
USA	5	3	

For those countries that reported service lives in both surveys there has been no change between 2002 and 2008.

ANNEX G: LESSONS FROM BUSINESS ACCOUNTING

Even before the introduction of the 1993 SNA, business accountants recognised that software whether purchased or produced in-house had asset characteristics. Generally, business accounting standards recommend the capitalisation of software as long as technical feasibility is established. In this section, three accounting standards will be described in more detail - The US Generally Accepted Accounting Principles, the International Financial Reporting Standards and the French business accounting recommendations - with some closing comments concerning the difficulties, notwithstanding these recommendations, in identifying software in business accounts in practice.

The US Generally Accepted Accounting Principles

The forerunner for accounting standards of software was in many ways the US accounting system GAAP (Generally Accepted Accounting Principles). Financial Accounting Standards Board (FASB) Statement No. 86 was the first statement to address a standard method for accounting for software. Although this did not cover software developed in-house, for internal use, the statement provided clear guidance on the treatment of software to be sold or leased (including “originals” produced for reproduction), and explicitly stated that all costs of software to be sold or leased during the research and development stage should be expensed until the point in time that the software becomes technologically feasible, when costs should be capitalised and treated as a master copy with subsequent costs capitalised as an intangible asset. After a clarification by the FASB, which became effective in December 1998, guidance was also given for software developed or purchased for internal use. This stated that the costs of such software should also be capitalised but not the costs in the final stage of implementation/operation such as training and maintenance which were to be expensed.

The International Financial Reporting Standards

Statement # 38 of the International Financial Reporting Standards (IFRS) stipulates that an enterprise should recognise an intangible asset (at cost) only if it is probable that the future economic benefits that are attributable to the asset will flow to the enterprise and that the cost of the asset can be measured reliably. Furthermore, it is stated that during the research phase all costs should be expensed as incurred. In the development phase costs are to be capitalised if the enterprise can demonstrate all of the following requirements: technical feasibility; intention and ability to complete the asset for use or for sale; ability to use or sell the asset; expectation of future economic benefits; and ability to measure expenditures during the development stage. The value of the intangible asset should be based on the accumulated costs of development. Costs of internally generated software would include expenditures on materials and services used in production; salaries, wages and other employment related costs of personnel directly engaged in production; any expenditure directly attributable to generating the asset; overheads that can be allocated on a reasonable and consistent basis. Software costs should not include selling, administrative and other general overhead expenditure nor should they include training costs for staff.

The French business accounting system

The French business accounting recommendations split an in-house software project into eight stages:

- i. Pre-analysis of feasibility;
- ii. Functional analysis;
- iii. Detailed analysis;
- iv. Programming;
- v. Tests;
- vi. Documentation;
- vii. Training;
- viii. Maintenance.

According to the recommendations only the costs incurred in stages (ii) to (vi) should be included in the valuation of in-house software.

Identification of software costs in business accounts

Proper identification of software costs in business accounts is complicated by the accepted ways in which accounting information is reported in business accounts. For capitalised software, costs are grouped under the general heading “intangible costs”. Intangible costs can include scientific or technical knowledge, design and implementation of new processes or systems, licences, intellectual property, trademarks (including brand names and publishing titles). Examples include computer software, patents, copyrights, motion picture films, customer lists, mortgage servicing rights, import quotas and marketing rights. In addition, the amortisation of intangible costs may include any or all of the above intangible assets without any specific identification of software related items.

Expensed software can also be problematic since the actual costs are often split among multiple accounts, which include consultancy expenses, research and development costs, computer expenses, labour costs, payroll costs, equipment depreciation, software amortisation, office supplies, direct manufacturing costs, miscellaneous accounts, utility costs, and other expense accounts. This heterogeneity makes it particularly difficult to have a comprehensive picture of the total costs on software. The expenditure problem can be particularly difficult related to product enhancements and updated versions (compared to original development costs).

ANNEX H: SOFTWARE-RELATED CLASSIFICATIONS

Contents

Table 1a – CPA 2008 Purchases of Software

Table 1b – CPA 2008 Own-account production

Table 2a – ISIC Rev. 4 Purchases of Software

Table 2b – ISIC Rev. 4 Own-account production

Appendix 1: CPA 2008 Detailed description list

Appendix 2: ISIC Rev. 4 Detailed description list

Appendix 3: Concordance Table between CPA 2008 / ISIC Rev. 4 / CPC Ver. 2

Appendix 4: Concordance Table between ISIC Rev. 3.1 and ISIC Rev. 4

Appendix 5: Concordance Table between CPA 2002 and CPA 2008

CPA Concordance Table

What follows are concordance tables based on the European product classification system (CPA 2008). Where the treatment is non-contentious (and evident) no further explanation is given.

The first table considers purchases of software and the second own-account production. That is, where software is intended for final-use by the purchaser and not intended for further processing nor for bundling/embedding (including outsourced purchases) in a subsequent sale, nor where the software is purchased as part of own-account production.

Table 1a – Purchases of Software		
CPA 2008 Code	Product Description	Intermediate Investment or
58.2	Software publishing services	
	Our understanding is that this category includes sales of pre-existing originals (including games) and software copies (off-the-shelf software, whatever the media). It includes licences to use and licences to reproduce and rentals.	
	♦ Original software – (purchases of pre-existing software originals)	GFCF
	♦ Other reproduced, rented, leased or licensed software. Including payments for "multiple-copy" licences. (<i>Payment can include, royalties, commissions, fees etc</i>):	
	- When purchased for bundling/embedding into products for subsequent sale or other software products or just sold on	IC
	- With a contract for no more than one year	IC
	- With a contract for more than one year	GFCF excluding games ²⁹
	♦ Payments for licences-to-reproduce software for subsequent sale:	
	- If the licence has the appearance of a change of economic ownership of part or a whole of the software original	GFCF
	- If the licence has the appearance of an operating lease	IC
62.01	Computer programming services	
	♦ IT design and development services for applications	GFCF
	♦ IT design and development services for networks and systems	GFCF
	♦ Software originals	GFCF
62.02.20	Systems and software consultancy services	
	♦ Software expected to be used in production for more than one year:	
	- For inclusion/embedding in an own-account 'original' – the value of own-account production should <u>not include</u> these costs if they are directly capitalised	GFCF
	- If the software is purchased by a final-user for inclusion in an own-account "original" the expenditure may also be treated as <u>intermediate consumption</u> as long as its value is <u>included</u> in own-account production	IC
	♦ Software expected to be used in production for less than one year. (This includes "customised" software purchased to be sold-on to another user/client.)	IC
62.02.30	IT technical support services	IC
62.03	Computer facilities management services	IC
62.09	Other information technology and computer services	IC
63.11.11	Data processing services	IC
63.11.12	Web hosting services	IC
63.11.13	Application service provisioning	

Table 1a – Purchases of Software (continued)		
CPA 2008 Code	Product Description	Intermediate Investment or
	♦ With a contract for no more than one year	IC
	♦ With a contract for more than a year	GFCF
63.11.19	Other hosting and IT infrastructure provisioning services	IC
63.12.10	Web portal content	IC

The table below describes items that could be included in the cost of own-account production of software.

Table 1b – Own-account production		
CPA 2008 Code	Product Description	Intermediate Investment or
62.02.20	Systems and software consultancy services	
62.02.30	IT technical support services	
62.03	Computer facilities management services	

ISIC Rev. 4 Concordance Table

What follows are concordance tables based on the International Standard Industrial Classification of All Economic Activities (ISIC), Rev.4.

Table 2a – Purchases of Software		
ISIC Rev. 4 Code	"Product" Description	Intermediate or Investment
5820	Software publishing This category is similar to CPA 58.2	
	♦ Original software – (purchases of pre-existing software originals)	GFCF
	♦ Other reproduced purchased, rented, leased or licensed software. Including payments for "multiple-copy" licences. (<i>Payment can include, royalties, commissions, fees etc</i>):	
	- When purchased for bundling/embedding into products for subsequent sale or other software products or just sold on	IC
	- With a contract for no more than one year	IC
	- With a contract for more than one year	GFCF excluding games ³⁰
	♦ Payments for licences-to-reproduce software for subsequent sale:	
	- If there the licence has the appearance of a change of economic ownership of part or a whole of the software original	GFCF
	- If the licence has the appearance of an operating lease	IC
6201	Computer programming activities This category is similar to CPA 62.01	GFCF
6202	Computer consultancy and computer facilities management activities This category is similar to CPA 62.02, meaning that it includes CPA 62.02.10 'Hardware consultancy services', plus CPA 62.02.20 'Systems and software consultancy services', and CPA 62.02.30 'IT technical support services'. The definition of this category says that services can include the provision of hardware. If hardware is included, then it should be classified as GFCF in hardware. Some services include the provision of software or enabling software and in general should be included in either purchased software or included in the costs of developing own account software	
	♦ Software expected to be used in production for more than one year:	
	- For inclusion/embedding in an own-account 'original' – the value of own-account production should <u>not include</u> these costs if they are directly capitalised	GFCF
	- If the software is purchased by a final-user for inclusion in an own-account "original" the expenditure may also be treated as <u>intermediate consumption</u> as long as its value <u>is included</u> in own-account production	IC
	♦ Software expected to be used in production for less than one year. (This includes "customised" software purchased to be sold-on to another user/client.)	IC
6209	Other information technology and computer service activities This category is similar to CPA 62.09	IC
6311	Data processing, hosting and related activities	
	♦ Application service provisioning	
	- With a contract for no more than one year	IC
	- With a contract for more than one year	GFCF

Table 2b – Own-account production		
ISIC Rev. 4 Code	"Product" Description	Intermediate or Investment
6201	Computer programming activities	
6202	Computer consultancy and computer facilities management activities	

Appendix 1: CPA 2008 Detailed description list		
58.2	Software publishing services	
58.21	Publishing services of computer games	
58.21.10	Computer games, packaged	
58.21.20	Computer games downloads <i>This subcategory includes:</i> - <i>electronic files containing computer games that can be downloaded and stored on a local device</i>	
58.21.30	On-line games <i>This subcategory includes:</i> - <i>provision of games that are intended to be played on the Internet such as provision of: role-playing games (RPGs); strategy games; action games; card games; children's games</i> <i>Payment may be by methods such as subscription or pay-per-play.</i> <i>This subcategory excludes:</i> - <i>on-line gambling services, see 92.00.14</i>	
58.21.40	Licensing services for the right to use computer games <i>This subcategory includes:</i> - <i>licensing services for the right to reproduce, distribute or incorporate computer programs, program descriptions and supporting materials for computer games</i> <i>This subcategory excludes:</i> - <i>acquisition of rights and publishing services, see division 58</i> - <i>off the shelf (packaged) software, see 58.2</i> - <i>limited end-user licenses as part of packaged software, see 58.2</i>	
58.29.1	Systems software, packaged	
58.29.11	Operating systems, packaged <i>This subcategory includes:</i> - <i>low-level software which handles the interface to peripheral hardware, schedules tasks, allocates storage, and presents a default interface to the user when no application program is running.</i> <i>Includes all client and network operating systems.</i>	
58.29.12	Network software, packaged <i>This subcategory includes:</i> - <i>software that is used to control, monitor, manage and communicate with operating systems, networks, network services, databases, storage and networked applications in an integrated and cooperative fashion across a network from a central location</i> <i>Includes all network management software, server software, security and encryption software, middleware, etc</i>	
58.29.13	Database management software, packaged <i>This subcategory includes:</i> - <i>collection/suite of software programs that enables storage, modification and extraction information from a database</i> <i>There are many different types of DBMSs ranging from small systems that run on computers to huge systems that run on mainframes.</i>	
58.29.14	Development tools and programming languages software, packaged <i>This subcategory includes:</i> - <i>software used to assist in the development and/or authoring of computer programs</i> - <i>software products that support the professional developer in the design, development, and implementation of a variety of software systems and solutions</i>	

Appendix 1: CPA 2008 Detailed description list (continued)		
58.29.2	Application software, packaged	
58.29.21	General business productivity and home use applications, packaged <i>This subcategory includes:</i> - <i>software used for general business purposes to improve productivity, or at home for entertainment, reference or educational purposes</i> <i>Includes office suite applications such as word processors, spreadsheets, simple databases; graphics applications; project management software, computer-based training software, reference, home education, etc.</i>	
58.29.29	Other application software, packaged <i>This subcategory includes:</i> - <i>cross-industry application software, i.e. software that is designed to perform and/or manage a specific business function or process that is not unique to a particular industry.</i> <i>Includes professional accounting software, human resource management, customer relations management software, Geographic Information System software, web page/website design software, etc.</i> - <i>vertical market application software, i.e. software that performs a wide range of business functions for a specific industry such as manufacturing, retail, health care, engineering, restaurants, etc.</i> - <i>utilities software, i.e. a small computer program that performs a very specific task, such as compression programs, anti-virus, search engines, font, file viewers, and voice recognition software (utilities differ from other applications software in terms of size, cost and complexity)</i> - <i>application software n.e.c.</i>	
58.29.3	Software downloads <i>This category includes:</i> - <i>electronic files containing software that can be downloaded and stored on a local device for a later execution/installation</i>	
58.29.31	System software downloads	
58.29.32	Application software downloads	
58.29.40	On-Line software <i>This subcategory includes:</i> - <i>software that is intended to be executed on-line</i> <i>This subcategory excludes:</i> - <i>on-line games, see 58.21.30</i> - <i>software downloads, see 58.29.3</i> - <i>on-line gambling services, see 92.00.14</i>	
58.29.50	Licensing services for the right to use computer software <i>This subcategory includes:</i> - <i>licensing services for the right to reproduce, distribute or incorporate computer programs, program descriptions and supporting materials for both systems and applications software. This applies to various levels of licensing rights:</i> • <i>rights to reproduce and distribute the software</i> • <i>rights to use software components for the creation of and inclusion in other software products</i> <i>This subcategory excludes:</i> - <i>limited end-user licenses as part of software, see 58.29.1-58.29.4</i>	
62.01	Computer programming services	
62.01.1	IT design and development services	

Appendix 1: CPA 2008 Detailed description list (continued)		
62.01.11	<p>IT design and development services for applications</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - services of designing the structure and/or writing the computer code, including updates and patches, necessary to create and/or implement a software application, such as: <ul style="list-style-type: none"> • designing the structure and content of a web page and/or of writing the computer code necessary to create and implement a web page • designing the structure and content of a database and/or of writing the computer code necessary to create and implement a database (data warehouse) • designing the structure and writing the computer code as necessary to design and develop a custom software application, other than programming for websites, databases, or packaged software integration • customisation and integration, adapting (modifying, configuring, etc.) and installing an existing application so that it is functional within the clients' information system environment <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - service contracts where the design and development of a web page is bundled with the hosting of the web page, see 63.11.13 - service contracts where the design and development of the application is bundled with the hosting and management of the application on an on-going basis, see 63.11.19 - service contracts where the design and development of a database is bundled with the on-going management of the data holdings, see 63.11.19 	
62.01.12	<p>IT design and development services for networks and systems</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - designing, developing and implementing customer's networks such as intranets, extranets and virtual private networks - network security design and development services, i.e. designing, developing and implementing software, hardware and procedures to control access to data and programs and to allow for the safe exchange of information over a network <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - service contracts where this service is bundled with the day-to-day management of the client's network, see 62.03.12 	
62.01.2	<p>Software originals</p> <p><i>This category includes:</i></p> <ul style="list-style-type: none"> - copyrighted intellectual property produced without contract for outright sale (i.e. with all-attendant property rights) - intellectual properties for sale that are implicitly or explicitly protected by copyright (e.g. computer software) <p><i>This category excludes:</i></p> <ul style="list-style-type: none"> - software produced under contract for others, see 62.01.11 - wholesale and retail sale services of software, see 46.14.11, 46.51.10, 47.00.31 	
62.01.21	Computer games software originals	
62.01.29	Other software originals	
62.02.20	<p>Systems and software consultancy services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - provision of advice or expert opinion on IT matters related to the IT systems and software, such as: <ul style="list-style-type: none"> • advice on matters such as software requirements and procurement • systems security 	

Appendix 1: CPA 2008 Detailed description list (continued)		
62.02.30	<p>IT technical support services</p> <p><i>This subcategory includes the provision of technical expertise to solve problems for the client in using software, hardware, or entire computer system, such as:</i></p> <ul style="list-style-type: none"> - <i>the provision of customer support in using or troubleshooting the software</i> - <i>upgrade services</i> - <i>the provision of customer support in using or troubleshooting the computer hardware, including testing and cleaning on a routine basis and repair of IT equipment</i> - <i>technical assistance in moving a client's computer system to a new location</i> - <i>the provision of customer support in using or troubleshooting the computer hardware and software in combination</i> - <i>the provision of technical expertise to solve specialised problems for the client in using a computer system, such as services of auditing or assessing computer operations without providing advice or other follow-up action including auditing, assessing and documenting a server, network or process for components, capabilities, performance, or security</i> <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - <i>computer disaster recovery services, see 62.09.20</i> 	
62.03	Computer facilities management services	
62.03.1	<p>Computer facilities management services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - <i>services of managing and monitoring communication networks and connected hardware to diagnose networking problems and gather capacity and usage statistics for the administration and fine-tuning of network traffic. These services also remotely manage security systems or provide security related services.</i> 	
62.03.11	<p>Network management services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - <i>provision of day-to-day management and operation of a client's computer system</i> 	
62.03.12	Computer systems management services	
62.09	Other information technology and computer services	
62.09.10	<p>Installation services of computers and peripheral equipment</p> <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - <i>installation services of mainframe computers, see 33.20.39</i> 	
62.09.20	<p>Other information technology and computer services n.e.c.</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - <i>data recovery services, i.e. retrieving a client's data from a damaged or unstable hard drive or other storage medium, or providing standby computer equipment and duplicate software in a separate location to enable a client to relocate regular staff to resume and maintain routine computerised operations in event of a disaster such as a fire or flood</i> - <i>software installation services</i> - <i>other IT technical support services, n.e.c.</i> <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - <i>computer programming services, see 62.01.1</i> - <i>IT consultancy services, see 62.02</i> - <i>data processing and hosting services, see 63.11.1</i> 	
63.11.1	Data processing, hosting, application services and other IT infrastructure provisioning services	

Appendix 1: CPA 2008 Detailed description list (continued)		
63.11.11	<p>Data processing services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - complete processing services and specialised reports from data supplied by clients or providing automatised data processing and data entry services, including database running services 	
63.11.12	<p>Web hosting services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - provision of the infrastructure to host a customer's website and related files in a location that provides fast, reliable connection to the Internet, which may be: <ul style="list-style-type: none"> • limited to storage on a single server, in either shared or dedicated capacity, without the service provider managing or integrating software applications (Software hosted on the server is the client's responsibility and service level guarantees are standardised and limited in scope) • a bundled service package that consists of the hosting and management of the website and related applications <p><i>An important characteristic of this service is the promise of a secure and reliable site and Internet connections that can be quickly scaled to accommodate variations in traffic use. Frequently, consulting, customisation and systems integration are part of the package. Applications are frequently e-commerce related and enable online storefronts, shopping carts and catalogues with advanced and complex features such as order processing, fulfilment, procurement, invoicing, transaction processing, customer relational management and back-end database and data warehouse integration and migration services.</i></p>	
63.11.13	<p>Application service provisioning</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - provision of leased software applications from a centralised, hosted, and managed computing environment: <ul style="list-style-type: none"> • with integration to the systems and infrastructure of the client (Frequently, consulting, customization and systems integration services are bundled with the hosting and management of the application) • where the leased application is not customized and not integrated with other applications of the client <p><i>(the application is usually accessed over the Word Wide Web. A common example is office suite software applications).</i></p>	
63.11.19	<p>Other hosting and IT infrastructure provisioning services</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - collocation services, i.e. the provision of rack space within a secured facility for the placement of servers and enterprise platforms (The service includes the space for the client's hardware and software, connection to the Internet or other communication networks, and routine monitoring of servers. Clients are responsible for the management of the operating system, hardware, and software.) - data storage services, i.e. the service of managing or administrating the storage and back-up management of data such as remote back-up services, storage, or hierarchical storage management (migration) - data management services, i.e. the on-going management and administration of data as an organisational resource (Services may include performing data modelling, data mobilisation, data mapping/rationalisation, data mining and system architecture.) 	

Appendix 1: CPA 2008 Detailed description list (continued)		
63.12.10	<p>Web portal content</p> <p><i>This subcategory includes:</i></p> <ul style="list-style-type: none"> - content provided on web search portals, i.e. extensive databases of Internet addresses and content in an easily searchable format <p><i>This subcategory excludes:</i></p> <ul style="list-style-type: none"> - published on-line directories and mailing lists, see 58.12.20 	

Appendix 2: ISIC Rev. 4 Detailed description list		
ISIC Rev. 4 Code	"Product" Description	
5820	<p>Software publishing</p> <p><i>This class includes:</i></p> <ul style="list-style-type: none"> - publishing of ready-made (non-customized) software: <ul style="list-style-type: none"> • operating systems • business and other applications • computer games for all platforms <p><i>This class excludes:</i></p> <ul style="list-style-type: none"> - reproduction of software, see 1820 - retail sale of non-customized software, see 4741 - production of software not associated with publishing, see 6201 - on-line provision of software (application hosting and application service provisioning), see 6311 	
62	<p>Computer programming, consultancy and related activities</p> <p><i>This division includes the following activities of providing expertise in the field of information technologies: writing, modifying, testing and supporting software; planning and designing computer systems that integrate computer hardware, software and communication technologies; on-site management and operation of clients' computer systems and/or data processing facilities; and other professional and technical computer-related activities.</i></p>	
6201	<p>Computer programming activities</p> <p><i>This class includes the writing, modifying, testing and supporting of software.</i></p> <p><i>This class includes:</i></p> <ul style="list-style-type: none"> - designing the structure and content of, and/or writing the computer code necessary to create and implement: <ul style="list-style-type: none"> • systems software (including updates and patches) • software applications (including updates and patches) • databases • web pages - customizing of software, i.e. modifying and configuring an existing application so that it is functional within the clients' information system environment <p><i>This class excludes:</i></p> <ul style="list-style-type: none"> - publishing packaged software, see 5820 - planning and designing computer systems that integrate computer hardware, software and communication technologies, even though providing software might be an integral part, see 6202 	
6202	<p>Computer consultancy and computer facilities management activities</p> <p><i>This class includes:</i></p> <ul style="list-style-type: none"> - planning and designing of computer systems that integrate computer hardware, software and communication technologies <p><i>The units classified in this class may provide the hardware and software components of the system as part of their integrated services or these components may be provided by third parties or vendors. The units classified in this class often install the system and train and support the users of the system.</i></p> <p><i>This class also includes:</i></p> <ul style="list-style-type: none"> - provision of on-site management and operation of clients' computer systems and/or data processing facilities, as well as related support services <p><i>This class excludes:</i></p> <ul style="list-style-type: none"> - separate sale of computer hardware or software, see 4651, 4741 - separate installation of mainframe and similar computers, see 3320 	

	<ul style="list-style-type: none"> - <i>separate installation (setting-up) of personal computers, see 6209</i> - <i>separate software installation, see 6209</i> 	
6209	<p>Other information technology and computer service activities <i>This class includes other information technology and computer related activities not elsewhere classified, such as:</i></p> <ul style="list-style-type: none"> - <i>computer disaster recovery</i> - <i>installation (setting-up) of personal computers</i> - <i>software installation</i> <p><i>This class excludes:</i></p> <ul style="list-style-type: none"> - <i>installation of mainframe and similar computers, see 3320</i> - <i>computer programming, see 6201</i> - <i>computer consultancy, see 6202</i> - <i>computer facilities management, see 6202</i> - <i>data processing and hosting, see 6311</i> 	
631	<p>Data processing, hosting and related activities; web portals <i>This group includes the provision of infrastructure for hosting, data processing services and related activities, as well as the provision of search facilities and other portals for the Internet.</i></p>	
6311	<p>Data processing, hosting and related activities <i>This class includes:</i></p> <ul style="list-style-type: none"> - <i>provision of infrastructure for hosting, data processing services and related activities</i> - <i>specialized hosting activities such as:</i> <ul style="list-style-type: none"> • <i>Web hosting</i> • <i>streaming services</i> • <i>application hosting</i> - <i>application service provisioning</i> - <i>general time-share provision of mainframe facilities to clients</i> - <i>data processing activities:</i> <ul style="list-style-type: none"> • <i>complete processing of data supplied by clients</i> • <i>generation of specialized reports from data supplied by clients</i> - <i>provision of data entry services</i> 	
6312	<p>Web portals <i>This class includes:</i></p> <ul style="list-style-type: none"> - <i>operation of web sites that use a search engine to generate and maintain extensive databases of Internet addresses and content in an easily searchable format</i> - <i>operation of other websites that act as portals to the Internet, such as media sites providing periodically updated content</i> 	

Appendix 3: Concordance Table between CPA 2008 / ISIC Rev. 4 / CPC Ver. 2

(Note: the symbol ‘*’ identify where there is more than one single relationship between classifications)

ISIC Rev.4	CPA 2008	CPC Ver. 2
5820	58.2	-
-	58.21.10	47822
-	58.21.20	84342 *
-	58.21.30	84391
-	58.21.40	73311 *
-	58.29.11	47811
-	58.29.12	47812
-	58.29.13	47813
-	58.29.14	47814
-	58.29.21	47821
-	58.29.29	47829
-	58.29.31	84341
-	58.29.32	84342 *
-	58.29.40	84392
-	58.29.50	73311 *
6201	62.01	-
-	62.01.11	83141
-	62.01.12	83142
-	62.01.21	83143 *
-	62.01.29	83143 *
6202 part * (includes hardware consultancy service)	62.02.20	83131
6202 part * (includes hardware consultancy service)	62.02.30	83132 *
6203 *	62.03.11	83161
6203 *	62.03.12	83162
6209	62.09	-
-	62.09.10	87332
-	62.09.20	83132 *
6311 part (e.g. includes streaming services)	63.11.11	
6312	63.12	84394

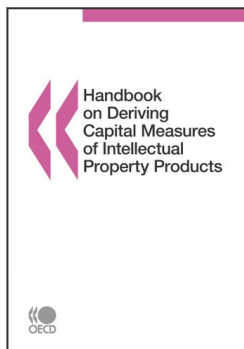
Appendix 4: Concordance Table between ISIC Rev. 3.1 and ISIC Rev. 4

ISIC Rev. 3.1	ISIC Rev. 4
7221 7240	5820
7229*	6201
7210 7211 7212 7229* 7230* 7290*	6202
7290 *	6209
7230*	6311

Appendix 5: Concordance Table between CPA 2002 and CPA 2008

CPA 2002	CPA 2008
72.21.20 *	58.2
72.21.12 *	58.21.10
72.40.11*	58.21.10
72.40.11*	58.21.20
72.40.11*	58.21.30
74.87.17 *	58.21.40
72.21.11*	58.29.11
72.21.11 *	58.29.12
72.21.20 *	58.29.12
72.21.11 *	58.29.13
72.21.20 *	58.29.13
72.21.11 *	58.29.14
72.21.20 *	58.29.14
72.21.12 *	58.29.21
72.21.20 *	58.29.21
72.21.12 *	58.29.29
72.21.20 *	58.29.29
72.21.11 *	58.29.31
72.40.11 *	58.29.31
72.21.12 *	58.29.32
72.40.11 *	58.29.32
72.40.11 *	58.29.40
74.87.17 *	58.29.50
72.22.12 *	62.01.11
72.40.13 *	62.01.11
72.22.13	62.01.12
72.22.12 *	62.01.21
72.22.12 *	62.01.29
72.22.11	62.02.20
72.22.14	62.02.30
72.30.10 *	62.03.11
72.30.10 *	62.03.12
72.60.10	62.09.10
72.22.15	62.09.20
72.40.13 *	63.11.11
72.40.12	63.12.10

-
- 29 Unless purchased by games arcades, game rental companies, etc
- 30 Unless purchased by games arcades, game rental companies, etc



From:

Handbook on Deriving Capital Measures of Intellectual Property Products

Access the complete publication at:

<https://doi.org/10.1787/9789264079205-en>

Please cite this chapter as:

OECD (2009), "Software and Databases", in *Handbook on Deriving Capital Measures of Intellectual Property Products*, OECD Publishing, Paris.

DOI: <https://doi.org/10.1787/9789264079205-6-en>

This document, as well as any data and map included herein, are without prejudice to the status of or sovereignty over any territory, to the delimitation of international frontiers and boundaries and to the name of any territory, city or area. Extracts from publications may be subject to additional disclaimers, which are set out in the complete version of the publication, available at the link provided.

The use of this work, whether digital or print, is governed by the Terms and Conditions to be found at <http://www.oecd.org/termsandconditions>.