KfK 5111
März 1993

Computer Aided Accountancy for Tritium Handling Systems

G. Spannagel, C. Schmid
Hauptabteilung Sicherheit
Projekt Kernfusion

Kernforschungszentrum Karlsruhe

Kernforschungszentrum Karlsruhe
Hauptabteilung Sicherheit
Projekt Kernfusion

KfK 5111

Computer Aided Accountancy for Tritium Handling Systems

G. Spannagel and C. Schmid*

*Schmid Systemberatung, D-7513 Stutensee 4

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Abstract

A program will be presented which can be used in tritium bearing systems for inventory taking and accountancy purposes. In particular, a detailed description will be given of the environment in which this program has been integrated. It is explained in which way a high user friendliness has been attained and which structures contribute to achieving the flexibility required.

Rechnergestützte Tritiumbilanzierung für tritiumführende Systeme

Zusammenfassung

Es wird ein Programm für die Inventur und die Bilanzierung von Tritium vorgestellt. Insbesondere wird die mathematisch-statistische Umgebung, in die das Programm eingebunden ist, beschrieben. Es wird dargelegt, auf welche Weise das Programm den Wünschen des Anwenders entgegenkommt und inwieweit die Programmstrukturen einer hohen Flexibilität Rechnung tragen.

Table of Contents

1. Introduction

According to present knowledge the first power stations equipped with nuclear fusion reactors will operate on a deuterium-tritium mix which will call for the nuclear fuel cycle to be complete. Its components are being investigated already now in the laboratories of large research establishments. Also the Karlsruhe Nuclear Research Center will run a so-called "tritium laboratory" from approximately 1993 on.

Tritium is rather expensive; it normally occurs as a gas and it is radioactive. Consequently, appropriate safety precautions must be taken in handling it. They include inventory taking and as far as possible a closed material balance.

Special attention should be paid to the discussion going on in the Federal Republic of Germany. It has been initiated recently by reports about some "occurrences" in handling tritium. The discussion has e.g. caused the highest court in the Federal Republic of Germany to decide to the effect that tritium - even if present in any small amount - shall be classified in principle as a weapon grade material.

For these reasons inventory taking corresponding to the most recent state of the art had to be established, supplementing the prescribed methods of containment and surveillance (so-called C/S measures). This can be done only by reference to the in-process inventories after proper accounting. For this, a program has been developed which will be described below. Its architecture and nearly all details generally apply to tritium bearing systems. Thus, a group of users will be addressed with this program which goes far beyond the number of operators of the Tritium Laboratory Karlsruhe. However, some particular features of KfK have evidently been taken into account.

It should be recalled that the program described here is just an element fitted into an "environment" commensurate to the task outlined above. This environment includes first modeling of the nuclear fuel cycle with its subsystems. This leads to simulation of the process which will provide so-called "raw" data of the in-process inventories. As both the subsystems and networking them are susceptible to changes - either from laboratory to laboratory or actually in reactor design - flexible tools of simulation will be necessary. Models have been worked out for NET/ITER relevant, but also for laboratory scale configurations [1-6]. As outlined in Figure 1, the measured data are subjected to methods of computation in an interim step, and after data reduction they are transferred into the program described here. Data reduction is of particular importance because by application of mathematical-statistical methods the computed final results can be validated scientifically. This has been discussed in [7]. The next step consists in test statistics whereby it is to be verified whether the balance is closed. If not, aids for decision making must be available: e.g., the operator wishes to know whether a measure (intervention into the system) is rewarding in terms of improving the quality of balance accountancy - which means that the pros and cons have to be considered [8-11].

Finally, it should be mentioned that the program described here can be referred to Near Real-Time Accountancy as well. This quite modern method of balance accountancy is favored in other fields too. First calculated results for tritium bearing systems can be found in [12].

2. General Task

It is the objective of accountancy to follow for a given system the acceptance and whereabouts of a substance, with the system boundaries well defined. Figure 2 shows an example of a tritium bearing system. Subsystems denoted according to the processes taking place in them are interlinked; tritium, but other substances too, may cross subsystem boundaries following the lines marked by arrows. Thus, it should always be attempted to take test inventories in cases where tritium is to be transferred from one subsystem into another. Therefore, we will speak in the following text of "transfers" only; the underlying observation has been outlined in Figure 3, namely output from subsystem O (output) via so-called "balance accountancy tanks" A and input into subsystem I (input). In this way, it is possible to take the inventory within the boundaries of the overall system.

Also the input and output of tritium has been indicated in Figure 3. The "container" subsystem which, actually, is located outside the boundaries of the overall system could e.g. be a delivery container. It should be repeated here that the boundary of the overall system "should be well defined for the whole period of operation." Although accountancy can be adapted to new circumstances, it should be kept in mind that this actually increases the probability that errors will occur. Depending on the design of a test facility "walls" made of various materials will serve as the containment. An elegant solution would consist in using a large container made of metal so that the system boundary could be defined in a relatively meticulous way. When the walls have been concrete cast or bricked up, the architect can make use of more degrees of freedom and the geometry is allowed to deviate from the well defined container structure.

We turn once more to the example of the "delivery container" as mentioned before. As shown in Figure 3, tritium will be transferred in this container into the overall system; shortly before being emptied, this container is already inside the boundaries of the overall system. Consequently, the information provided by the shipper has to be included in the accounting system. It will be understood below that this will allow a first balance to be made. However, it is of special interest to split up e.g. a delivery into partial steps which also furnish data on inventory taking, although tritium is outside the first, i.e. the innermost wall of the system. This is an exception worth mentioning; regular inventory taking relates to tritium inside the first wall. In our example this will be the case for the first time when the valve of the delivery container opens towards the subsystem coupled to it. At that moment the transfer represented in Figure 2 will take place. The receiving subsystem will in each case be the "store." The "Stationary Transfer Station" (STS) accommodates the balance accountancy tank A. Thus, upon completion of the transfer, the receiver inventory has been taken and, as already indicated above, balance accountancy will be possible which gives the well-known shipper-receiver difference.

The relationships described make already clear how closely accounting is interlinked with individual steps in the process. This leads to helpful approaches to solutions. One of them has already been indicated: the process of "tritium transfer" is imaged in a program module as shown in Figure 2. As a matter of fact, the data needed for inventory taking can be determined conveniently during a transfer because many transfers lead via the balance accountancy section inside the Stationary Transfer Station. Also is it useful in accountancy that data are measured in the process steps which are helpful in inventory taking. Moreover, these signals are available on-line and they can be kept ready for the program without further interventions by the operator. This avoids e.g. the "input" error source. Finally, imaging of the process for inventory taking allows quite a number of plausibility checks to be made. They actually become more and more important because familiarization with EDP assisted sequences gives rise to a dangerous type of credibility.

The accounting program shall provide for a dialog between the operator and the accounting system. Both will acquire the information needed for the dialog.

For instance, the operator wishes to query whether a measurement signal generated has been acquitted by the program. The acquittance may differ greatly in quality: The lowest level which could be defined would be "signal stored as such;" the highest level could be "signal used to calculate the final result," and an intermediate level might note the confidence level at which the measured result belongs to a given statistical population.

Presumably, not all signals generated will be provided on-line to the accountancy system in the first phase of operation; consequently, the system will wait for input by the operator through a dialog.

Imaging the "transfer" process step in a program module means that this module shall record the process step completely. This implies that a noticeable deviation from the part steps previously defined is no longer possible. Although steps within the subsystem involved can be repeated as many times as desired - batchwise operation imposes this option - a component unknown to the process step cannot be included. If the operations will be conducted on a routine basis - which can be anticipated - the remark above might be superfluous. But experience has told us that the initial phase is accompanied by surprises. The alternative would be to further improve the flexibility of the program which is always feasible although a price has to be paid for it. In our case this would mean that substituting an extended version for a presently still transparent one will multiply the sources of error.

Upon completion of each transfer operation the inventories of the subsystems involved in the transfer are updated. All the rest of inventories are corrected to take into account the radioactive decay of tritium. Overall inventory taking can subsequently be performed which first of all relates to all points in the system where noticeable amounts of tritium are being stored or kept in an interim store. Besides, there are amounts of tritium which in quantitative terms would not be noticeable; however, for safety reasons they have to be recorded together with other information. This will be explained by the following example:

Uranium tritide, a stable compound at room temperature, may e.g. be used to store tritium. When the temperature is raised, this compound decays and tritium is released. It can hardly be avoided during that process that also the wall of the uranium tritide accommodating container is heated to a relatively high temperature which increases tritium diffusion through the container wall. Diffused tritium is collected and its amount can be determined. According to the definition above of the overall system boundary also this tritium makes part of the system inventory. Therefore, during each transfer making use of such processes tritium concentrations, wall surfaces, temperatures, time intervals, etc. must be entered into the accountancy system. Only experience gathered during operation will show whether a single information can be gradually replaced by laws. This would also relieve accountancy, e.g. inclusive of the archive.

The archive contains all data received from and calculated by the accounting system. Thus, the maximum data set available also includes information which is not of interest to each user. Maybe, the totality of information may be reduced, at least before it is filed in a "long-term archive."

It goes without saying today that the unauthorized access to the program is impeded by a suitable user verification. This locking may be extended ad lib. Above all balances which, obviously, are calculated in the program, are not made accessible to everybody.

It has already been mentioned at the beginning that the work described here has been integrated in an environment. Modeling required in this environment again and again makes use of an "accountancy quality" parameter. It might be prescribed e.g. by an authority. This makes clear that accountancy calls for statistical tools which set the general conditions, e.g. concerning the range of validity of the results of evaluation. This is true in particular for the calculus of errors which is not completely contained in this version of the program. The necessary mathematical-statistical tools have been discussed in [7]. It should be added in this context that it is known from other areas - especially those classified as sensitive - that above all systematic error components contribute to the uncertainty of inventory taking and balance accountancy so that special attention should be paid them.

The program was first elaborated at the KfK Central Computer Department where mainframes and the related infrastructure are available which offer optimum conditions. The language selected has been FORTRAN because the PC to be used at a later stage will be equipped with a FORTRAN compiler so that optimum portability of the program will be guaranteed [13]. As the work progressed - then already at the PC level - it became more and more evident that especially the access times had grown longer. Therefore, the present version has been written in C; Figure 4 shows the scheme and above all its modular nature. It is visible from its structure that extensions or modifications are so simple that this system is suited for application in each accountancy of tritium bearing systems.

3. Description of Program Modules

The following description will be brief because details in the program listing [14] have also been provided with adequate explanations. The summary serves the main purpose of illustrating to the reader the relationships existing between the modules. But still, some examples will be described accurately, especially in case such program parts will recur rather frequently and the reader should be enabled to recognize structures at other places.

3.1 TLK

The accounting program makes always use of similar data types and functions, respectively. They will be described here in general terms and can then be called from other modules. It should be underlined that TLK does not provide information on the networking of the system components which are contained in the logic of the program.

The generic term selected for all system components is the "container" which actually occurs as such; in the program "container" may mean also the "pipeline" system component.

Data which have been specified in almost all cases are summarized under the term of "structural data." For instance, the denotations or volumes of the containers have been clearly defined because they have been permanently installed in rather large systems for an extended period of time. As a matter of fact, it might be conceived that e.g. the maximum filling pressure of a container is changed as a result of operating experience accumulated. But even those data have been included in the structural data. The second type of data include information which undergoes relatively fast variations during the time. This applies e.g. to the gas pressure prevailing in a container. This type of information is termed "balance data".

The TLK module defines all types of container and all accountancy items, i.e. including those of conceivable losses. With a view to filing, information for all components are compiled in a data set and filed in TLK.DAT as "snapshot." Thus, this file contains the comprehensive data collection and, consequently, serves as a basis for further records, e.g. relating to users. As an example, the authority will be satisfied to receive a short survey and expect more comprehensive records only in case of discrepancies which cannot by cleared up.

Also m-dimensional matrices have been defined in TLK (m can be chosen ad lib). As operating experience grows, the matrices will have to be filled with measured values which are helpful in computing the so-called "adhesion to the wall." Moreover, definitions can be found in this module which are necessary for parts of the program assisting the operator. They include e.g. the computation of inventories using the dimensional systems presently available (gram, mole, Curie, Becquerel).

3.2 TLK.STR

In the first part the structures of the containers will be described which, as already mentioned, may also be pipelines, pumps, etc. This description of each container has, in principle, been entered on two lines; the first line contains four characteristics, the second line contains eight characteristics.

The first line contains only denotations for the container: first the non-abbreviated version including a maximum of 40 characters which, however, is not interpreted by the program. This is followed by the mode of operation and the type of container in an abbreviated description, both of them capable of interpretation. In addition, numbering has been included with a view to display on the screen.

On the second line the maximum admissible amount of tritium in the containers is indicated. This is followed by information about the inner surface of the wall, the pressure, the points of measurement and (for getter containers) the amounts of uranium. It should be underlined once more here that many of the characteristics indicated above are helpful in the plausibility checks surveyed by the program. For instance, indication of the amount of uranium makes no sense for a transport flask and causes output of an error message.

The second part of this module contains the occupation of the matrices above; the latter serve to calculate tritium amounts which, due to the conditions in the system, cannot be transferred into the desired position of the store. The example cited has been adhesion to the wall.

3.3 TLK.C

This module contains the main program where all variables are agreed upon and the functions are defined which have been used so far. These functions will be described below.

Among all losses the decay of tritium can be recorded most conveniently. This is done by the first function. The latest data set is always updated and stored together with the older data. The updated inventory list can be output both on the screen and on the printer.

So far all measured data have been used without prior verification to calculate the inventory. The functions for the gas equation and a simplified error calculus have been included in the program. As soon as a certain amount of operating experience has been accumulated an expensive data analysis should be performed. Appropriate procedures assisisting in that analysis have been compiled in [7]. Further functions will be responsible for scanning measuring points and results on the gas analyses.

Another function serves to read in data which, ordered by containers and parameters (e.g. tritium pressure, dwell time), are necessary in computing losses. Also networking of the containers (structural file) and the latest status of the balance accountancy file are read in. The last section of TLK.C serves in operator log on/verification and in assisting the operator (mouse control).

3.4 UTLG

As mentioned above, all tritium transfers are passed via the so-called "balance accountancy section." The amount transferred is measured and so-called "losses" - unavoidable in each transfer - are estimated and entered in the accounting system. Despite the fact that the transferring unit is different from the receiving unit, these processes resemble each other to such an extent that it will be sufficient to describe the modules by one single example, namely by the UTLG-module. Evidently, these modules will image the respective transfer processes. Consequently, it is reasonable to make available the greatest possible number of signals generated in the process/process control system to the accounting program - as a matter of fact on-line. This has been discussed before.

The module mentioned above considers the transfer of tritium from a delivery container into a storage container. The delivery container can be connected to the transfer station at a point equipped accordingly. Then the condition is examined of the subsystem which is used for the transfer; reports, e.g. concerning the necessary evacuation of all containers involved, must be elucidated; there are inquiries which guarantee that exactly the containers used in the process are actually approached in the program. By means of plausibility checks quite a number of error sources can already be detected here. For instance, the inquiry concerning the amount delivered and the date of measurement is useful in such a check because the so-called shipper-receiver difference calculable in this way should not exceed a defined value.

The process step investigated here, namely tritium transfer from one delivery container into the store, has been selected so as to be of limited extension, but still this step can be actually subjected to minor variations. This has been taken into account in the UTLG module and can be controlled by a mouse. The first inquiry includes such a possibility: The gas analysis of a delivered amount should be available at the earliest possible date - or should not be available in case the values had been obtained by different means.

The most important steps with respect to inventory taking consist in the batchwise transfer of partial amounts from the delivery container into a storage container. The inventories of these partial amounts are taken in so-called "balance accountancy tanks." The data measured are introduced into the gas equation and the amounts of the inventories can be output or stored, expressed in the presently usual units. The operator decides about the partial amounts to be transferred until, ultimately, a residual amount defined under these aspects is taken over by a tritium retention system. Even the last partial amount transferred in case of changes in process control (helium buffering) is recorded by the module as tritium taken over by the store. During these partial steps both the so-called "balance accountancy tanks" and the storage containers might be changed. Changing of the "balance accountancy tank" is reasonable if the amount which varies in each individual transfer underrates the limits specified by the operator: actually, the partial amounts become gradually smaller. Change of the storage container accepting the amount is also reasonable; for instance, the case might arise that an experiment requests an amount which is approximately equal to the amount of tritium transferred (i.e. a batch). Then this batch is transferred into an empty storage container so that transfer to the experiment is simplified; the accountancy system provides for these special cases.

As so-called "losses" should perhaps be recorded, information is stored in the program as to which system components had been exposed how long and under which boundary conditions to tritium. Maybe, tritium bound by the containers through adhesion to the wall can be estimated in this way and this amount would be identical to such loss.

3.5 UTIL

It has already been mentioned that this program profits from the fact that quite a number of process steps recur in a similar form. Therefore, it is reasonable to combine into the UTIL module mentioned above the auxiliary functions which, likewise, are repeatedly used. Also this approach would facilitate extension which is conceivable - at least proven structures do exist.

The following auxiliary functions can be used already now:

- discretionary access control of an operator is to be verified;
- routine for menu inquiries;
- input and display control;
- control of data output;
- definition of the container names and the corresponding numbers which are required for accountancy purposes;
- information helpful in conversion into the four dimensional systems (mole, g, Ci, Bq);
- interpolation between supporting values of the matrices.

3.6 TLK.DAT

Data holding of the balance accountancy program is presently performed in a plainly structured sequential file. It consists of data sets which in turn are composed of individual fields.

Each data set present in the file defines the complete information contained in the balance accountancy system at a specified point in time. The data sets are arranged within the file in an order of values which increase with time. It has been specified already now that all data sets within a file must have the same basic structure and the same size. If at a given point in time the structure of the balance accountancy area undergoes variations, this implicitly will change the structure of the data set, too. At that point in time a new file must be used which satisfies the new structure. A change in structure is effected by a change in the TLK.STR structural file.

It might become necessary to abandon this simple structure in a future version in order to be able to respond more flexibly to changes in structure. This would mean that the balance accountancy file (or the balance accountancy data base, if applicable, in case the use of a data base system offers itself) must be capable of filing a structure which undergoes changes with the time. One problem emerging here will be that not only the "historical" balance information must be available at any moment later-on, but the structure related information valid at that time must equally be amenable to readout so that balance information can be interpreted in the proper way.

The individual data sets in the balance accountancy file consist of several components. Some of them fulfil the requirement of the balance accountancy program, e.g. an unambiguous characterization of the types of each data set in order to be able to detect defective files. However, the most important information to the operator of the program are the data on the inventory which are stored for each container. A note will be added to each data set as to the point in time at which it had been valid. In this way, it will be possible to retrieve data sets for each point in time (assigned to the balance accounting period).

4. The Structure of the Balance Accounting Program

As has already been mentioned before, the present version of the program has been written in the C-language. It has been written so as to be portable and used on a PC under MS-DOS or also on UNIX systems. This is of great importance with a view to the improvement of the program because it might become necessary in the future to use the program in a multi-tasking environment. This is important with a view to possible future on-line balance accountancy involving permanent and automated data acquisition.

Global data structures and variables have been defined in the TLK.H file. It includes among others the definition of the balance accountancy file and variables used in a number of program modules, e.g. accountancy items which are frequently used. Thus, this file is of fundamental importance to the interested reader who wishes to understand the program.

The program proper has been composed of modules. Some of the modules are global in scope because they include functions which have been taken from other modules (TLK.C, UTIL.C, COMM.C, CTYPE.H). The TLK.C and UTIL.C modules have already been mentioned. The rest of modules define specific sequences in balance accountancy, among them the modules EVAK.C (evacuation), EXIS.C (transfer from an experiment to ISS), EXLG.C (experiment to store), ISLG.C (ISS to store), LGEX.C (store to experiment), LGUT.C (store to transport container), TCLG.C (TCS to store) and UTLG.C (transport container to store). The UTLG.C module has already been discussed.

The main program proper which provides the frame controlling the balance accountancy program is contained at the end of the TLK.C file. It is responsible above all for controlling the logon event with authorization check and includes a main menu which serves as a module distributor. When the user selects a point from the main menu, the main program is branched into the function processing the selected part. Normally, this is one of the modules mentioned in the previous section. However, also those points are anchored in the main menu which enable the user to retrieve information stored in the balance accountancy file. This feature is restricted at present to the instantaneous status of the balance, but it is easy from the point of view of program technology to extend it to any points in time in the past.

In order facilitate to the interested reader of the program text familiarization with the program, some elementary data and program structures will be described in more detail here.

Many processes within the balance accountancy program relate to one or several "containers." At present, some of the knowledge concerning the laboratory structure is contained in the TLK.STR structural file. However, another part of knowledge is contained in the program modules proper. This applies in particular to the modules which directly concern the sequences in balance accountancy (e.g. UTLG.C). In those modules the steps of the program refer to containers which had been defined in the structural file. Only that reference as well as the use of previously defined containers in the sequences of the program will cause the overall structure of the laboratory to come to life (in terms of accountancy).

In order to simplify in terms of program technology the programming of events which involve containers and to make them more transparent, recurring events have been combined into "functions" which define schematically the sequence of these events. However, depending on the program module, these functions have to be called with the help of different parameter lists. For instance, in one event the distance covered from the docked transport container to the balance accountancy tank might be of interest whereas in the next event the distance covered from the balance accountancy tank to the storage container might be of interest. But still, many sequences resemble each other greatly so that they have been combined into some few functions which are called again and again.

Appropriate parameters identifying the containers used have to be assigned to such functions. Since the number of containers differs from case to case, parameter assignment at that point must be dynamical. This has been achieved by the concept of the so-called "groups" (meaning groups of containers). One group is represented in the program by a single network list. Depending on the practical case, the groups are networked and having been used they are deleted as the case may be (as the memories available in the programs are a relatively limited resource). Several functions are available to build up such a group. One function (grp) is capable of composing an equivalent group from a chain of characters which contains the names of the containers, separated by commas. Only a finite number of such groups occur in the program and, therefore, they are not deleted after use but kept for potential reuse at some later date. Another function (grpcat) is able to concatenate two already existing groups (i.e. to build up a new group containing all elements from the two generating groups). Then the new group generated in this way can be used, independent of generating groups, and after use it should be deleted again by means of the grpdel function (the occupied memory position being cleared again). These functions have been defined in the UTIL.C module.

A good example of the application of groups is the function "Inquiry 1" from the TLK.C module. This function performs a number of inquiries relating to information about the containers listed in the group and which is transferred as first parameter to that function. The retrieved information includes the pressure and temperature of each container of the group. If a container has several points for retrieval of pressure and/or temperature values, all are scanned and the arithmetic mean of the data is obtained. Plausibility checks are performed during inquiries by which it is, e.g., examined whether the inputs are within admissible limits.

Should the inventory of the group have changed compared with the previous status, the amounts of surplus and loss, respectively, must have originated at or must have gone to some place. Therefore, the second parameter in calling that function to be stated in addition will be to whom a surplus or loss has to be credited or charged.

Other parameters concern variables containing the specified pressures and temperatures, information about the purity, control of the inquiry and display on the screen and, finally, a file on which a log is to be written simultaneously.

It should be added that the function "Inquiry 1" makes use of the "druck_temp" subfunction to cause read-in proper of the pressure and temperature values. This type of structure improves the readability and maintainability of the program.

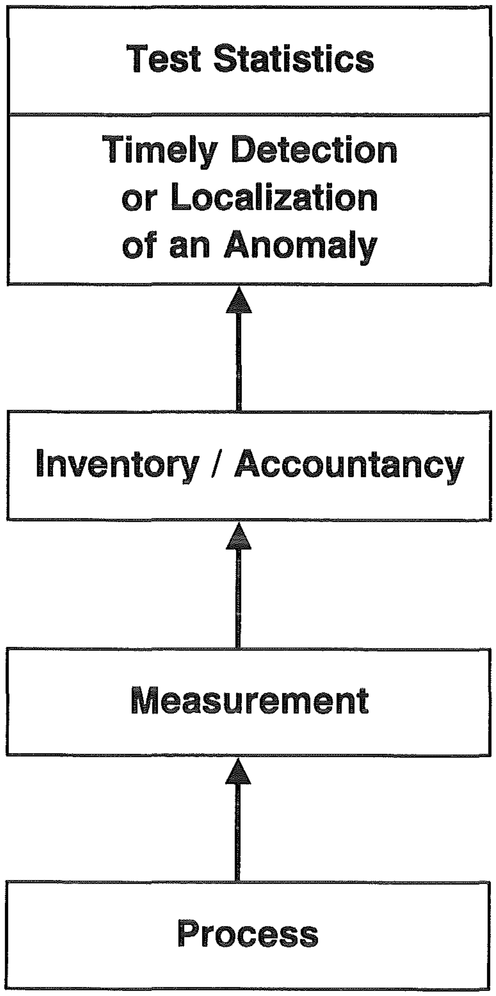

Fig. 1 Accountancy simulation package.

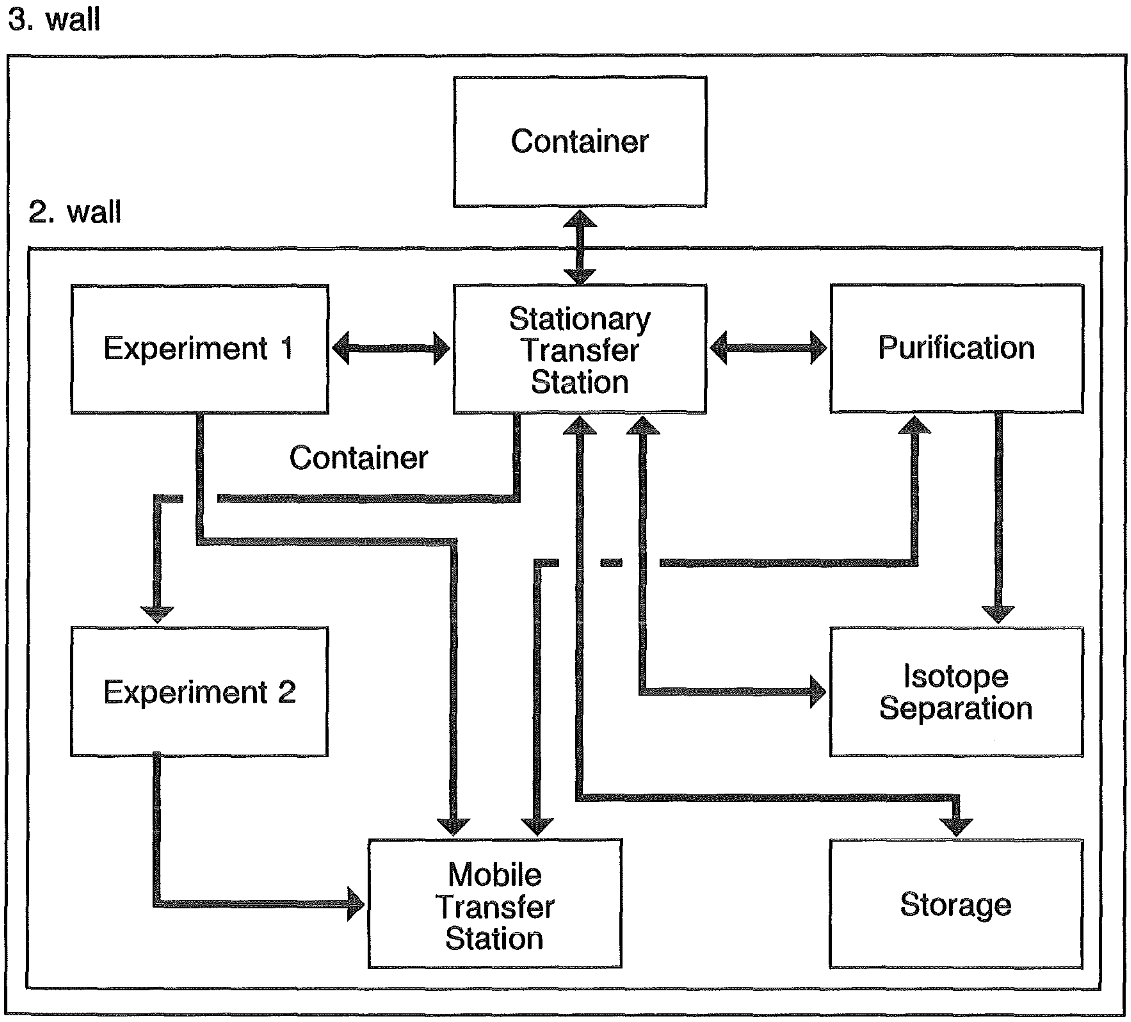

Fig. 2 Typical infrastructure of a tritum laboratory; the material balance area is located within the 3. wall.

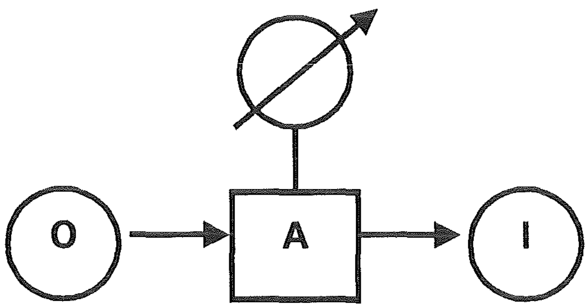

Fig. 3 Every tritium transfer contributes inventory data used in the accountancy system.

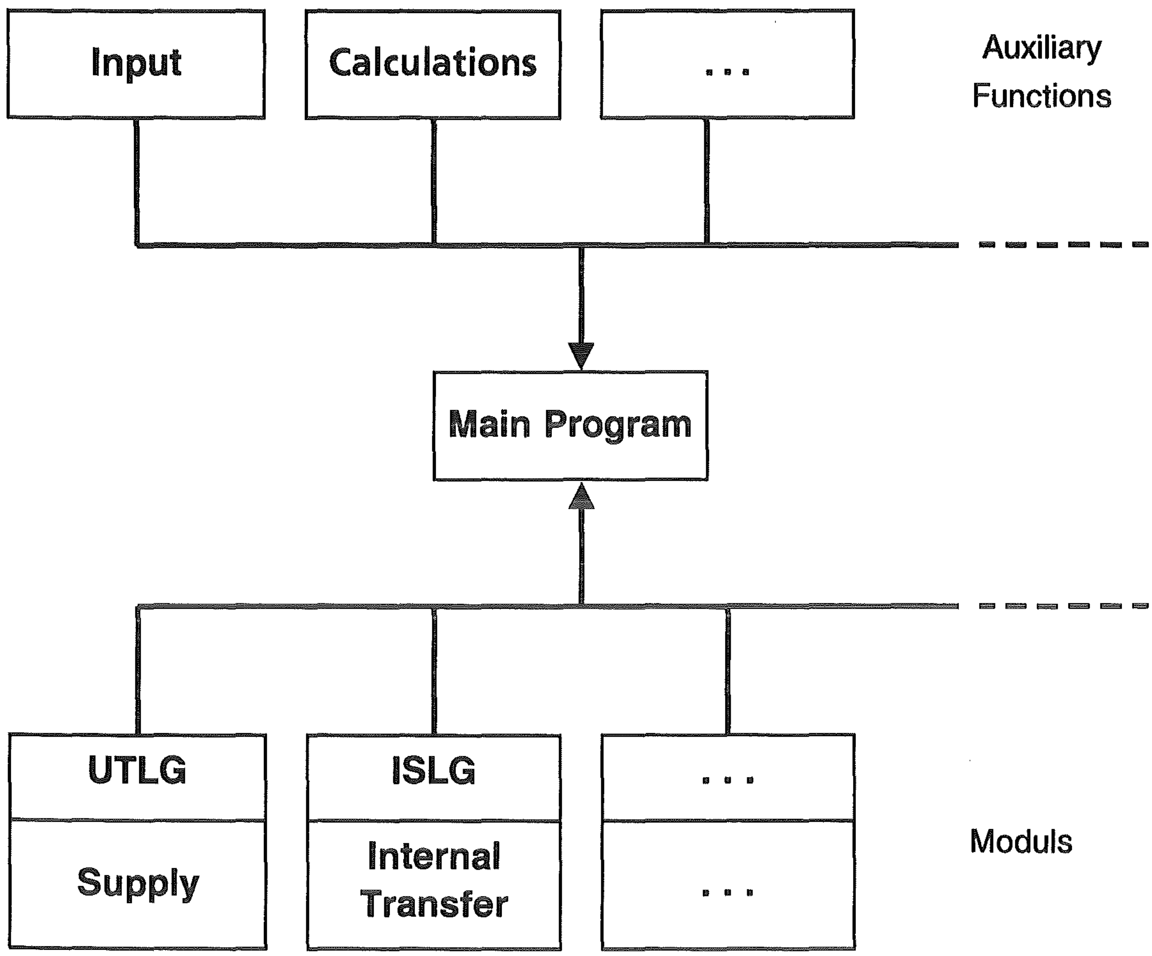

Fig. 4 Structure of the program.

Acknowledgement

The architecture of this accountancy tool is of general nature; with respect to a validation in the Karlsruhe Tritium Laboratory special strategies of operation were discussed with U. Besserer, G. Jourdan and P. Schira. These discussions are highly acknowledged.

5. References

[1] E. GABOWITSCH, G. SPANNAGEL, S. TACZANOWSKI, Modelling of the 3H-Cycle of Fusion Reactors, Proc. of 4th ICENES, Madrid, 44 (1986).

[2] E. GABOWITSCH, G. SPANNAGEL, S. TACZANOWSKI, Modelling of the ^{3}H-Cycle of Fusion Reactors, International Journal of Modelling and Simulation, 7, 122 (1987).

[3] E. GABOWITSCH, G. SPANNAGEL, S. TACZANOWSKI, Modelling of the 3H-cycle of Fusion Reactors, in Emerging Nuclear Energy Systems, Ed. G. Velarde and E. Mingnez, World Scientific Publ. Co., 434 (1987).

[4] E. GABOWITSCH, G. SPANNAGEL, Computer Simulation of Tritium Systems Used in Fusion Technology Research, Fusion Technology, Vol. 16, 143 (1989).

[5] E. GABOWITSCH, G. SPANNAGEL, Modelling and Simulation of Tritium Handling Systems, Kerntechnik 53, 202 (1989).

[6] G. SPANNAGEL and P. GIERSZEWSKI, Dynamic Tritium Inventory of a NET/ITER Fuel Cycle with Lithium Salt Solution Blanket, Fusion Eng. and Design 17, 271 (1991).

[7] G. SPANNAGEL, Statistical Elements for the Accountancy in a Tritium Handling Facility, to be publ. (1993).

[8] R. AVENHAUS, G. SPANNAGEL, Internal Report, KfK (1987).

[9] R. AVENHAUS, G. SPANNAGEL, Analysis of Tritium Laboratory Accountancy Data, Fusion Technology, Vol. 14, 1102 (1988).

[10] R. AVENHAUS and G. SPANNAGEL, Analysis of Accountancy Data of the Tritium Laboratory Karlsruhe, Fusion Technology, Vol. 21, 471 (1992).

[11] R. AVENHAUS, G. SPANNAGEL, Accountancy under Typical Operational Strategies used in Tritium Processing, KfK-Report 5101 (1992).

[12] G. SPANNAGEL et al., Near Real Time Accountancy for Tritium Handling Systems, to be publ. (1993).

[13] M. STAMS and G. SPANNAGEL, Internal Report, KfK (1990).

[14] Internal Report, KfK (1992).