EURO PROGRAMME
PRODUCT FLOW DIAGRAM
GUIDANCE NOTES

7 March 2003

CONTENTS

EURO PROGRAMME – PRODUCT FLOW DIAGRAM

This note accompanies the Product Flow Diagram (PFD) that sets out broad guidance on the typical project requirements that you might want to consider in planning for potential changeover work to make IT systems euro compatible.

The guidance explains what steps your organization might need to take from the planning stage of any IT system changeover project to completion. Boxes 1-20 will be expanded in more detail below to help users of the PFD decide what practical steps might be of relevance to your particular circumstances. In using the PFD, you should not feel constrained to follow the order of steps suggested here. For example you might want to formulate your Business Requirements & Migration Strategy (see Box 5 below) before the User Requirements by System (Box 3), rather than as suggested here. You are free to choose the order of steps that best suits the needs of the organization but the guidance should provide assistance as needed.

BOX 1 - PROGRAMME INITIATION

Starting considerations

1.1 Before considering a euro changeover/implementation programme you might want to ask the following questions to help you determine your organization's current situation:

- Are management and personnel familiar with the potential consequences of the euro for the organization?
- Has management formulated policies for euro compatibility?
- Has it been determined which currency related products are being offered or bought?
- Has anybody been made responsible for euro compatibility?
- Has the organization conducted an inventory to identify all currency related objects (information systems, contracts, forms etc.)
- Has the organization conducted risk analyses for euro risks?

Recommendations

1.2 At this stage, it would be prudent for management to:

- Formulate risk management strategies in relation to potential euro changeover, and
- Ensure this is cascaded throughout the organization through some form of Business Guidelines.

BOX 2- INVENTORY AND ALLOCATION OF RESPONSIBILITIES

Identify your systems and business processes

2.1 Conduct an inventory of all IT systems and business processes - including those business processes/applications that are to become redundant - with a view to identifying euro risks.

2.2 To assist this identification process you need to find out whether:

- The systems/processes include monetary fields?
- They make reference to monetary amounts or currency symbols?
- They impact or introduce business processes involving monetary amounts?
- You are sure they have no currency functionality.

Recommendation

2.3 You should ensure ownership and responsibility are assigned wherever a euro risk has been identified. Proper business ownership will facilitate effective and timely preparation of User Requirements (URs) for these systems.

Prioritize your systems in order of criticality

2.4 The changeover programme may require changes to several hundreds of IT systems. Business areas, internal technical teams and other suppliers may not have the resources to manage and undertake all the necessary changes to all the IT systems simultaneously. Similarly, during testing and deployment, there might be competition for user, technical and other resources that might need to be reprioritized. Appropriate prioritization would help the organization to reduce the risks of non-delivery of key deliverables for the programme as a whole.

Recommendation

2.5 The business priority of each IT system that needed to be changed would need to be identified and in particular, the minimum usable subset of IT systems that would support the business should be established and guaranteed to be delivered. (This subset of **'business critical'** systems would likely be limited to those in the 'Immutable' and 'High' priority categories as defined below.)

— A manual workaround or fallback procedure is already available or relatively easily achievable.

Priority categories

2.6 Each IT system (except those that would be 'exempt', i.e. no euro-impact) could be assigned a priority using one of the following ratings: 'Immutable', 'High', 'Medium' or 'Low'.

- **Immutable** – for IT systems:
  — That perform business processes and/or produce outputs where legal obligations to do so exist; or
  — Where it would be presentationally unacceptable or would create Ministerial and/or Departmental embarrassment if systems were not available by RT Day;

- **High Priority** – for IT systems that perform business processes and/or produces outputs that provide a major contribution to the achievement of key departmental targets.

- **Medium Priority** – for IT systems:
  — That perform business processes and/or produce outputs that provide an effective contribution to the achievement of key public sector/departmental targets; and

- **Low Priority** – for IT systems (typically standalone IT systems having only manual interfaces with screen based and paper based outputs, and have a small number of users):
  - That perform business processes and/or produce outputs that provide a moderate to zero contribution to the achievement of key departmental targets; and/or
  - Manual workaround or fallback procedures are already available or easily achievable.

BOX 3 – URs BY SYSTEM

Identify your systems interfaces (internal & external)

3.1 The systems inventory should be enhanced to show details of each system's key components, focusing in particular on interfaces.

System Owners to Prepare User Requirement (UR) Documents

3.2 Having identified key systems together with details of their components and interfaces, User Requirement (UR) documents will then need to be produced. URs will be used to assist creation of technical solutions and subsequently be critical to potential conversion planning.

Recommendation

3.3 System owners will be responsible for ensuring that URs are prepared on time and to the required standard. Programme Managers may therefore wish to issue a standard UR template to ensure the document includes key information.

For an example UR template please see the Customs & Excise Euro Programme User Requirement template*.*

BOX 4 – TECHNICAL SOLUTIONS

Cluster Specificity

4.1 Depending on the size and complexity of your organization, and in order to facilitate the development of technical solutions for your applications, you may find it sensible to group identified systems into clusters such as accounting, administration, payroll etc. A cluster can be a group of closely linked systems/processes and could possibly be addressed at the same time within a single project. Systems within a cluster will not necessarily be implemented at the same time but it is highly likely that major systems within a cluster could all be tested and implemented as one unit.

Produce Technical Solution Documents

4.2 A technical solution document will need to be drafted to describe the technical solution selected for each system (or cluster of systems). The audience will include estimators / planners and the authors of URs. If systems have been grouped into clusters, the document should state the reasons for clustering and identify any related systems not forming part of the cluster.

Recommendations

4.3 Identification of both interfaces and the level of business dependency on those interfaces are essential. Business continuity and end-to-end processing would be impacted by these dependencies. To this end:

- A clearly defined technical solution should be drawn up for each cluster identified to ensure that euro changes for these systems/processes would be carried out consistently, reflecting both technical considerations and user application needs.

- External interfaces in particular should be carefully negotiated and managed as failure to manage those interfaces would increase the risk of non-delivery. Risks can be minimized through use of certification at each stage in development. Rigorous monitoring of progress and timescales is essential to development. Contracts agreed well in advance should ensure that what is being delivered meets both business and IT needs, and timescales.

BOX 5 - BUSINESS REQUIREMENTS AND MIGRATION STRATEGY

Transition period service need

5.1 Organizations such as tax authorities may need to deliver services denominated in euro during any UK transition period, i.e. they may be required to conduct some business in both euro and sterling. The migration strategy should therefore take in hand issues such as whether to replace a system or whether to add dual-currency or multi-currency capability.

5.2 Experience shows there are a number of ways to deliver a desired outcome making best use of existing systems. It is therefore possible for the migration strategy to incorporate a variety of different approaches depending on individual systems or clusters.

Recommendations

5.3 The migration strategy should address the following:

- Detail the way in which currency amounts (balances and financial positions) would be recorded internally in a particular IT system with monetary functionality;
- Input flows from external sources and output flows to other systems in the correct form (interfaces); and
- The timetable.

BOX 6 - TEST STRATEGY AND GENERIC TEST PLANS

Test Strategy

6.1 Formulating a test strategy to document the approach to be taken throughout the testing lifecycle for systems changed in preparation for the potential changeover is essential. The test strategy should take account of User Requirements, any future proofing requirements (e.g. expanding field sizes, removing hard coded literals, parameterising thresholds) and any assumptions made when preparing the estimates.

Recommendations

6.2 You should identify what test environments are required and how long they will need to be available. By planning ahead it may be possible to combine the testing of more than one application on a platform at one time. You also need to consider the amount of storage space required to store the test systems and the test data required as part of this process.

6.3 You should state how configuration management is to be handled. This may have to be done with the team that supports the systems that may be internal or external to the organization. Ideally, testing should be split into cycles each of which can be successfully completed and signed-off before moving onto the next one.

6.4 You need to identify any specialist skills required for testing or any outside assistance required. Also if any independent third parties such as auditors are required to witness the testing, they should be involved in the structure of any test documentation required along the way.

Generic Test Plans

6.5 All interfaces to and from each system should be tested and all dependencies within that service identified and built into the test plans.

6.6 All the tests should be are documented and changes recorded and subject to change control to allow a full audit trail of the processes to assist with other activities.

6.7 There may be various features of any given system that may not need to be tested. These must be documented so time is not wasted testing features not affected by any code changes.

BOX 7 - EXPLORE TECHNICAL OPTIONS / PROTOTYPING

7.1 Equipped with an inventory of systems which would be affected by the euro, there are a number of tasks that can then be undertaken independently while work on business requirements, technical solutions etc. is carried out. Exploration of technical options and prototyping is one area of work that, if planned correctly, will save valuable time/resource later in the programme.

Involve users in developing solutions

7.2 It is recommended that prototyping be employed to build and explore options / solutions based on emerging User Requirements. Users should be closely involved in this prototyping work. Application of a project methodology, such as DSDM (Dynamic Systems Design Methodology), that requires users and technical staff to work closely together is strongly recommended.

7.3 If resource is limited, it may be sensible to identify the organization’s most commonly used platforms / languages and target these areas for prototyping or other work relating to technical options. If your organization has systems of various types (accounting, invoicing, salaries etc.) it may be worthwhile to select a typical system from each type and explore and develop possible solutions in order to prove the subsequent approach to be taken.

7.4 Another approach to prove emerging solutions might involve non-IT exploration of possible solutions, for example by the manual conversion of an individual account from sterling to euro. This approach provides a useful check on whether a proposed solution would deliver any inaccuracies or anomalies.

BOX 8 - TOOLS EVALUATION EXERCISE

Conversion tools

8.1 The decision whether to utilize conversion tools may be dependent on the size and complexity of the systems identified in your systems inventory. Experience from the euro area indicates that tools were not used and in many cases the changes were made using the traditional development methods and building upon the experiences gained during Y2K. However, you need to consider carefully whether it would be a viable option to perform euro remediation without additional or specific euro tools.

Recommendation

8.2 If your systems are currency related and your lines of code are numbered in millions, it is strongly recommended that a tools evaluation exercise be undertaken as the process should probably be partially or fully automated.

Partially automated process

8.3 These may be where scanning and data flow tracing tools are utilized to identify the code that needs to be changed. In this instance traditional techniques can then be employed to carry out subsequent changes to code. (A fully automated process differs only in that subsequent code changes would be undertaken using mass change tools thereby potentially saving vast human resource.)

8.4 Experience has shown that there is no single tool that can be recommended for all code conversion. For example, the structure of VME Cobol systems used by Inland Revenue, DWP and HMCE is different for each department. An exercise undertaken by these departments concluded that a solution for one department would not work equally well for the other departments and that each department's systems need to be considered independently. The assessment did not find one service or tool that could handle the requirements of all three departments without some changes to meet the individual needs of each department.

Stand-alone tools

8.5 Stand-alone tools designed to enable conversion of mainframe generated code such as Cobol generally do no provide direct support for euro conversion on the mainframe. These tools usually require the code be downloaded to some other platform such as a PC or a server. The analysis and code remediation would be conducted on that platform and the source code returned to the mainframe.

8.6 The advantage of using a platform independent of the mainframe means that euro compliance work could be done without impacting present mainframe resources. Also because tools are usually employed independent of the mainframe, the option to contract out this work to a third party becomes viable.

8.7 The disadvantage is that the components that make up the systems would all need to be transferred to the working platform for scanning and remediation and then transferred back again. This is a significant task requiring strong configuration management.

BOX 9 - CONSIDER COMMON ISSUES

9.1 An organization needs to consider and identify the key common issues which may impact upon numerous systems or clusters in order to develop standards that can be widely implemented. Key issues include:

- Screen design,
- Outputs,
- Customer identification.

Screen Design

9.2 A key message here is that screen designs should be consistent across an organization.

Recommendation

9.3 Screen design standards should be developed in tandem with other prototyping work. When developing these standards an organization needs to consider factors such as whether:

- screens can contain mixed currency amounts,
- to use GBP or EUR rather than not £ or € to differentiate between currencies,
- to highlight different currencies in specific font colours etc.

For a detailed example of screen design standards see the Inland Revenue Euro Programme Euro Screen Design Requirements.

Outputs

9.4 This relates to forms, prints and notices. Designs should be consistent across the organisation as customers may deal with the organisation across a number of different services.

For a detailed example of Forms design standards please see the Inland Revenue Forms Design Requirements for New Developing Systems.

Customer identification

9.5 Consideration needs to be given as to how customers would notify an organization that they wish to deal in euro. If the customer deals with an organization across numerous services (e.g. – a company may deal with HM Customs & Excise for VAT, Customs Duty, Landfill Tax and Insurance Premium Tax purposes, or a taxpayer with Inland Revenue for Income Tax, Capital Gains Tax, and Trust Taxation) how would the organization manage the customer's request to deal in euro? The customer would only be required to inform an organization once but the organization must manage this notification to ensure all services identify the requirement.

BOX 10 - ESTIMATES FOR SYSTEM APPLICATION CHANGES

High level estimates

10.1 At this stage you should be able to gauge high-level estimates for all applications which would require modification for the euro. This should enable you to:

- Further develop the approach to conversion,
- Identify required resources,
- Plan the next stages of any single currency programme.

Helpful documents

10.2 Documents previously produced such as the Business Requirements and Migration Strategy, User Requirements, Technical solutions and work relating to testing (together these documents effectively can be viewed as the **Virtual Functional Specification** – see the dotted box into which Boxes 3-6 flow) will all feed into the estimating phases.

Tools and Associated Training estimates

10.3 If tools are to be utilized you should consider whether there is any training required and how their use may impact on the estimates. If tools include scanning and data flow software they may be used to enable more accurate estimating as they are able to identify numbers of occurrences of currency fields in large suites of programmes much more quickly and effectively than by traditional methods.

Field Size

10.4 A critical element of conversion is field sizing and all currency related fields would need to be identified. Because sterling is in mathematical terms less ‘granular’ than the euro (on current exchange rates 1 pence in sterling carries a higher value than 1 euro cent) a significant proportion of all currency fields will usually need to be expanded to accommodate the euro.

10.5 While working on systems for euro related reasons, this may be an appropriate time to carry out some system future proofing and expand some other currency related fields that have reached their size limit over an extended period due to inflation and changing business trends.

Recommendation

10.6 Care should be taken to ensure that estimates encompass both work relating to field sizing and future proofing as well as to basic monetary conversion work where the strategy relating to issues such as masking, bridging, wrapping, smoothing and encapsulation would be addressed.

BOX 11 - DATA CONVERSION ESTIMATES

11.1 In preparing data conversion estimates you will need to refer to the Virtual Functional Specification documentation.

Sterling and euro data

11.2 An organization may receive and need to access data held in both sterling and euro, depending on the level of demand for euro services during the transition period. It would be up to the business to decide at what point systems were converted to run with euro as the primary currency; this will be dependent on an individual organization's business requirements.

11.3 Various factors will have been taken into account by an organization when considering how to undertake data conversion. Whatever the solution, it is important that data integrity is maintained and the data remains available to those who require access to it.

For a detailed explanation of the issues relating to data conversion please see Annex 2 of the Euro Compatibility Technical Guidelines.

BOX 12 - SYSTEM AND DATA CONVERSION PLANS

Purpose

12.1 These would provide high level planning information for euro conversion of an organization's systems portfolio. The estimates relating to applications and data conversion together with Business Requirements, User Requirements and Technical Solutions will all need to be taken into account in order to draw up these plans.

Recommendation

12.2 An organisation need not be too prescriptive about the composition of these plans. However the plans should provide timeline-based information that would enable identification of at least the following:

- Effort by system or cluster of systems,
- Implementation Route Map,
- Data conversion requirements,
- Indicative euro key milestones,
- System Cluster dependencies,
- A description of the timeline and planning assumptions and risks.

12.3 Once formulated, the Programme Team should have access to these plans to allow their inclusion in the overall Programme Plan. This will also enable progress to be monitored to ensure that activities are undertaken within the required timescales.

BOX 13 - CHANGE SPECIFICATIONS AT PROGRAM LEVEL AND TEST PLAN FOR SYSTEMS

Change Specifications at Program Level

13.1 If the Business Requirements and Migration Strategy, User Requirements, and the Technical solutions documents are all sufficiently detailed and complimentary, the task of the analyst/programmer will be far more straightforward. Also work on independent tasks such as prototyping, tools evaluation, screen design and trader recognition may need to be taken into consideration when drawing up these specifications.

Recommendation

13.2 To ensure documents are of a sufficient standard Programme Managers should

- Give thorough attention to detail in earlier phases of the programme to help ensure conversion plans are not delayed unnecessarily,
- Consider issuing standard document templates.

See Box 3.2 System Owners to Prepare User Requirement Documents and Example UR template.

Prepare Test Plan for Systems

13.3 Various items must be tested as part of the assurance process. This will include the assurance of the components that comprise the system and/or service and the testing of the system has a whole. All components should be identified as you may not have responsibility for testing all parts of the system individually.

13.4 The test plan is an essential document for each system as it sets out how the testing is to be completed, who is responsible, what criteria will be used for success and failure and how you will handle problems. The test plan must be distributed to all stakeholders so they are aware of their roles and responsibilities.

13.5 It is essential that you define the scope of the testing and what needs to be tested. In addition you will need to state what policies are to be adopted and the method of testing to be used and any specialist techniques required. ***See Box 16 Test for more detail relating to preparation of test plans***.

BOX 14 - DATA CONVERSION SPECIFICATIONS AND TEST PLAN FOR CONVERSIONS

Prepare Data Conversion Specifications

14.1 Careful planning of data conversion in the early stages will save time and effort in the long term. The system and data conversion plans will detail what data is to be converted and when the work is to be undertaken. ***See Box 12 System and Data Conversion Plans***. The plans should have been created using the Business Requirements and Migration Strategy, individual User Requirements and Technical Solutions as their input. This information is key to preparation of data conversion specifications.

Conversion rules

14.2 All data conversions carried out would adhere to EC Regulations covering conversion and rounding using the pre-defined fixed conversion rate. The data conversion specification should therefore reflect these rules.

14.3 It is generally recognized that conversion from one currency to another may produce large, or even infinite, numbers of decimal places. It is for this reason the conversion rate would be set to six significant figures.

For a detailed explanation of these rules and other issues relating to conversion and rounding please see Annex 2 of the Euro Compatibility Technical Guidelines.

Prepare Test Plan for Conversions

14.4 A number of issues may need to be considered here depending on what solution an organization is implementing. For example, during transition, and possibly after a potential E-day, an organization may be required to hold data in both sterling and euros.

Recommendation

14.5 Where data would be held in sterling and euro, you should test that stored data can be retrieved in the correct/expected currency. Even where the system would be euro only following conversion, this check may still be relevant if stored data is to be made accessible.

Data integrity

14.6 Tests should be carried out to ensure that data items stored on the database are consistent, are stored in the correct currency and are retrievable by the system. Tests of data residing in a database should also check that stored rounded amounts are as expected and are represented correctly when they are retrieved. Similar tests should also be applied to confirm integrity of stored amounts is retained.

14.7 If your solution requires database conversion from sterling to euro, before and after images should be checked to ensure that the currency conversion has been applied successfully. In these circumstances, financial audits may be required to ensure that the newly converted database balances with the old database.

Database fields

14.8 It is also worth checking that database fields are large enough to store potential euro amounts and that no truncation could take place. (Field sizes should have been checked and increased as necessary as part of the work carried out preparing estimates for system application changes – ***see Box 10 Estimates for System Application Changes***.)

Test Data

14.9 Test data must be made available or created to allow functional tests to be carried out. In the latter case, you must consider how you are going to create the data.

14.10 It may be a requirement that test data be aged to mimic the application being run in a live environment. If so, you need to consider how data ageing takes place, either by an incremental process or by a big bang approach to a date beyond cutover.

14.11 The system owners will be responsible for assuring that test data is satisfactory for testing purposes. You should also decide who will be responsible for upkeep of data and if it needs to be made available to other systems for the testing of interfaces etc. ***See Box 13 Change Specifications at Program Level and Test Plan for Systems.***

BOX 15 - CODE AND DATA CONVERSIONS

Plans

15.1 This stage of work will be less complicated if the system and data conversion plans have been satisfactorily drafted. ***See Box 12 System and Data Conversion Plans.*** The system and data conversion plans will detail what data is to be converted and the timetable for carrying out the work.

Data Conversion

15.2 The data conversion itself should be carried out in accordance with the data conversion specifications that follow EC Regulations on conversion and rounding using a pre-defined fixed conversion rate. ***See Box 14 Data Conversion Specifications and Test Plan for Conversions***.

Code Conversion

15.3 Similar comments apply to this. Code changes will be more straightforward to carry out if the program specifications are well written and contain sufficient detail. ***See Box 13 Change Specifications at Program Level and Test Plan for Systems***.

15.4 Effective code conversion will also be dependent to some extent on the system and data conversion plans (***See Box 12 System and Data Conversion Plans***) as they will detail data conversion requirements which will feed into the program specifications.

BOX 16 - TEST

Environment

16.1 A test environment should be created and maintained in which changes to systems can be tested or even to test any existing systems to identify any problems.

Platform

16.2 You will need to ensure that the platform required is similar or preferably identical to the one on which operational systems are currently running. If the system is due to be supported on another platform, then it is essential that all likely platforms are used to test the system.

Regression testing

16.3 Regression testing is the process of allowing system changes to be released into the live environment (after unit testing) to enable the support team to let users test the changes for normal business functionality and integrity. Regression testing must be considered to enable the maximum amount of time for testing to be taken. If this is not possible, you should plan when the system can be implemented.

Configuration management in development phase

16.4 At any point in the changeover process time, it should be possible to identify which parts of the system have been changed and their status. It is therefore essential that a configuration management system is established to monitor the process of auditing, fixing and testing. The changes made as part of this process should still be included as part of any configuration management procedures.

Interfaces

16.5 Interfaces between systems and differing modules of the system may be manual or electronic. It is essential that all interfaces are tested to ensure that both parties, importing and exporting data, are satisfied that the information supplied is correct and that the format cannot lead to ambiguity of information.

Test Scripts

16.6 These will have to be written for all differing scenarios. These test scripts should be utilised by the users during the testing phase. This will enable users to coordinate their activity and ensure that:

- there is no duplication,
- the right tests are being undertaken with the correct data,
- the testing process is properly documented.

16.7 The test scripts should document the test and the expected result. If the expected result is not achieved then this should be logged and recorded.

16.8 This process will also assist with incident management. All unexpected results can be logged and passed back to the support team at the end of the cycle.

Approach

16.9 System changes and testing can be undertaken in a variety of different ways e.g. bottom up, top down, big bang or critical first.

Recommendation

16.10 It is recommended you complete a bottom-up testing process and test components in isolation then complete further tests in groups on modules which have common business functionality before completing a fully integrated test.

Business requirement assurance testing

16.11 One of the objectives of assurance testing is to make sure that the system still conforms to the business requirements and objectives after any changes. As a general rule tests should conform to the requirements of the business processes involved and the operational cycles. This has the benefit of making the tests as realistic as possible. Typical cycles for this testing could include:

- Collection and validation of data;
- End of day/overnight batch operations;
- End of week, month, year, century; and
- End of financial years.

BOX 17 - CERTIFICATION PROCESS

17.1 This is a key factor of your euro changeover programme. Following completion of testing, those who have been assigned with euro responsibility should be tasked to certify the euro compliance of their system. This certification must include a statement relating to the compliance of key components and all internal and external interfaces.

17.2 Completed certification should be collated by the Central Programme Team to enable the monitoring of overall euro compliance against the organization’s inventory. ***See Box 2 Inventory and Allocation of Responsibilities***.

Failure to assure

17.3 Where those who have been assigned with euro responsibility are unable to give an appropriate assurance then the matter should be escalated at the earliest opportunity through to the Central Programme Team and the appropriate authority advised accordingly.

BOX 18 - RELEASE

18.1 Having carried out exhaustive testing and received certification from the appropriate authority that a system is fit for purpose, deployment within the live environment can then be considered.

Configuration management in live system

18.2 Configuration management is crucial not only within any development, but also within the live system itself. There is little point burning up a substantial proportion of the allocated resource on testing, and subsequently requesting assurances from system owners relating to compliance, if an audit trail is unable to confirm that the product tested and assured is exactly the same as the one that is to be used in the live environment.

Recommendation

18.3 With the high incidence of change to be encountered within an organization’s euro changeover programme it is vital that you ensure everyone involved knows exactly what status everything has, or had at a given point in time, so that a known, safe state can be reverted to when a change is discarded or goes wrong.

BOX 19 - MONITOR

19.1 Monitoring effectiveness of euro converted systems would be of significant importance. Once systems go live, they should be closely monitored to ensure their operation is as anticipated. If an organization has employed the following measures in earlier stages there should be fewer problems than there might otherwise be:

- Thorough testing of system with full user involvement - system changes having been released into the live environment enabling users to test the changes for normal business functionality and integrity. ***See Box 16 Test***

- Effective configuration management with strong change and release control procedures in place.

Errors

19.2 As with any software related exercise, there are always going to be some mistakes and errors found, some of which may not be directly under your control. An organization should therefore ensure that procedures are in place so that any errors found can be addressed and rectified accordingly. These procedures should be in place anyway but they should be scrutinized to ensure they are robust.

Continuity Planning

19.3 Wherever possible an organization should ensure that their normal fallback procedures and business continuity plans take account of any euro changeover programme as appropriate.

BOX 20 - DEVELOPING SYSTEMS

Euro compatibility and value for money

20.1 When developing new systems, euro compatibility implications should be considered at the earliest opportunity. All new or upgraded systems should already now incorporate euro compatibility where this represents value for money.

Timing

20.2 You might need to bear in mind the timing of any new developments against the organization's key service provision and operational milestones though this should have been covered within the planning phase.

Currency neutrality

20.3 A system would be easier to convert if the euro becomes the chosen currency of the UK if it has been designed as currency neutral. To facilitate full euro compatibility, all new systems should therefore be designed as currency neutral. Therefore, where thresholds are required, the threshold amount should be passed to the program via external parameters or tables and not hard-coded. Likewise, euro glyphs and £ signs should not be hard-coded. Also, field sizes should be designed big enough to accommodate euro amounts (e.g. a 3 digit field which can accommodate a figure of £999 will need to be designed as a 4 digit field to accommodate the equivalent euro amount).

Factoring in euro compatibility

20.4 Where appropriate, it is the responsibility of the project owners to ensure that specifications or contracts for system supply have euro compatibility built in. It represents good value for money to build in currency neutrality into all new systems as subsequent conversion to the euro, should it be required, would be both cheaper and easier to undertake.

For further guidance on building in both currency neutrality and basic elements of full euro compatibility see Annex 3 of the Euro Compatibility Technical Guidelines.