Issues Related to a Soft Transition of Existing Archives to a Preservation Environment

Fabio Nanni[1], Giorgos Flouris[1,2]

[1] Istituto della Scienza e delle Tecnologie della Informazione (ISTI), CNR, Italy
[2] Institute of Computer Science (ICS), FORTH, Greece
f.nanni@ao-pisa.toscana.it, fgeo@ics.forth.gr

Abstract. This document addresses the problem of ruling a soft transition of an existing archive to an externally managed preservation environment. When such a transition must take place, the archive management has to decide whether to hand over the complete custody of its data holdings and/or its procedures to the preservation system, or to retain control of any of these, and, if so, to what extent. We are motivated by the admittance that most archives will probably be quite reluctant in abandoning their data, as well as their well-established procedures, protocols and interfaces, in favor of something external to the archive. This admittance raises a number of issues related to how to make the transition of the archives into the preservation system as smooth as possible and how to facilitate the interaction between the two systems.

Keywords. Information Preservation, Digital Preservation

1 Introduction

One of the most challenging problems being faced by modern archivists is the rapid obsolescence of large volumes of digital (especially "born-digital") information. This problem is being addressed in the research field of information preservation; an interesting analysis and introduction to the problem appears in [11]. Preserving digital information is a challenging problem, not fully understood to date; this stands in contrast to its importance for librarians and archivists. Recently, several international research efforts have dealt with the problem (examples being [1], [3], [6]), leading to a number of methodologies that allow the preservation of digital material, such as [2], [4], [5], [6], [7], [9], [10]; nevertheless, the field is still in its infancy.

The task of preserving large amounts of digital material is a demanding one in terms of resources, technology, know-how and incurred costs. Moreover, all related costs will have to be afforded by current users, whereas the preservation activity targets future users; this is one characteristic of preservation which makes funding of preservation activities quite difficult, despite its tremendous importance [8]. One possibly attractive strategy for archives and libraries in order to minimize the necessary investment could be to "outsource" preservation, i.e., to use some external organization, with the expertise, know-how, resources and systems available to carry out the task of preserving a digital archive. For example, the ongoing project CASPAR [1], is

in the process of implementing such an experimental system, based on the OAIS model [6], [7], which is a high level reference model and an ISO standard.

Switching from a standard archive system to an externally preserved archive system using the above method is not a task to be taken lightly. We acknowledge that existing archives may have already in place strategies, workflows, as well as human, hardware and software resources which help them perform their duties, such as browsing, accessing and delivering services, and such procedures are not likely to be abandoned just for the sake of preservation. Moreover, the archive may wish to retain custody of the data to be preserved, or may want (or be legally bound) to handle preservation activities itself, under the supervision and consultancy of the external preservation system, rather than allowing the preservation system to modify and access the data itself.

On the other hand, a preservation system will require full access to the contents of the archive and will need the maximum possible degree of control over the data itself and the preservation activities in order to be successful in preserving the data. Moreover, the capabilities of the services it will offer, in terms of access, searching, retrieval etc, will probably be low-level and less specialized than the ones offered by the archives, so such services are unlikely to be found acceptable by the archive users.

Therefore, a balance should be struck among the conflicting requirements of not changing the well-established and exhaustively tested archive's procedures on the one hand, and, on the other, handing enough control of such procedures (and data) to the external preservation system so as to be able to perform preservation. The purpose of this paper is to identify the related issues and evaluate the various tradeoffs involved, in order to help the management of archives in their task of determining the best solution for their case, depending on the peculiarities of the environment at hand.

Towards this aim, we identify three interrelated dimensions (axes) along which such issues need to be studied, namely, the data, the services and the preservation. The data dimension studies where the data should be physically stored; three options are possible, scilicet storing the data in the archive (only), in the preservation system (only), or in both. The services axis evaluates the effects of the transition as a function of the amount of integration of the services offered by the archive with the services offered by the preservation system, where by the term services we mean all the procedures, protocols, interfaces, search and retrieval facilities etc of the archive. Several different options can be imagined in this respect, ranging from the archive not using the preservation system's services at all, to completely abandoning its processes in favor of the processes of the preservation system. Finally, the preservation dimension determines the details of the actions that need to be taken when the data is close to becoming obsolete; related issues include where and by whom such actions should be implemented, and what effects these actions would have to the related services.

The decisions taken with respect to each of the above axes affect the quality of the interoperability between the archive and the preservation system, the amount of integration required, the smoothness of the initial transition and the success of the preservation process itself. Among the several different options available, certain combinations make more intuitive sense than others and are described in section 6. The best set of options depends on the peculiarities of the archive environment, so this paper makes no commitment related to "best practices"; in fact, we argue that there is no "best practice" for all applications.

2 Preliminaries

In the following, we assume that an archive system (hereafter denoted by AS) decides to use an externally operated, specialized preservation system (hereafter denoted by PS) in order to avoid the obsolescence of its data. We assume that all decisions related to how and when the preservation activities will take place are dictated by the PS. In some cases, an AS may have developed its own preservation mechanisms; such mechanisms will have to be abandoned, or it makes no sense to hire an externally managed preservation system. The case where preservation is handled internally by the AS does not present the difficulties described here and will not be further considered.

In addition, we assume that end-users do not interact with the PS, but with the AS; thus, search, browsing, retrieval etc requests are passed to the AS which, depending on the case, may be able to process them internally, or may have to pass them over to the PS to get the related answers or data. This assumption is motivated by the admittance that the PS is designed to handle preservation, therefore its interfaces are probably inadequate to handle the diverse needs of different communities of users, interested in information from various domains. Moreover, allowing end-users to interact with the PS would introduce a number of complications related to its interoperation with the AS, and/or deprive the AS from any useful functionality.

3 Location of Data and Related Issues (Data Dimension)

In general, we can assume that the PS will not just store the data in a “raw” manner, as is done in a standard AS or information system, but will include some type of additional data-dependent information that will be used for preservation purposes. Moreover, it is likely that the PS will reorganize the data in a manner more suitable for its needs. For example, a PS compliant with the OAIS specification [6] will internally package the preserved data using a specially formatted structure, called the AIP.

This has two effects on the interaction between the systems. The first is that when the AS decides to initiate the interaction with the PS, it must necessarily transfer its data to the PS, regardless of where the data will be ultimately stored. Then, depending on the case, the PS may store the data itself inside its preservation structures (packages), or delete the actual data and store, in each package, an external “pointer” to the associated data (in the AS), rather that the data itself.

The second effect is that each block of data in the AS (database record, digital object, multimedia file etc) will get a new, internal associated identifier (ID) in the PS’s storage medium, which is the ID of its associated package; this ID would exist instead of, or in addition to, the one it has in the AS. To achieve cooperation, one (or both) system(s) will have to expose the IDs associated with any given block of data to the other and maybe also some related internal information (metadata).

The above discussion gives rise to two different issues: the first is where the data is physically located (AS, PS, or both); the second is who exposes its IDs and related information (AS, PS, or both). Of course, not all combinations make sense. Firstly, if the data is stored in one system only, then this system will necessarily have to expose

the IDs associated with its data, otherwise the other system will have no way to retrieve the required data for preservation purposes (if it's the PS) or for accessing, retrieval etc purposes (if it's the AS). Secondly, if the PS's algorithms require significant transformations upon the data, rather than a simple "repackaging", then storing the data in the AS only is not really an option, as it will not be possible to associate the transformed data to the original using simple pointers.

3.1 Available Options for the Data Dimension

To decide on the optimum strategy to follow, the AS management must first decide whether the AS needs (or wants) to remain the main custodian of the data. If so, then the AS will be responsible for access, searching, retrieval etc activities and it can largely continue to use its well-established procedures with minimal changes to its software. Since the AS still has control of the data, all services could be directly handled by it rather than using the PS's API.

One drawback of this option is that the AS will have to communicate each change it performs on its data to the PS, either in order for the PS to perform the same change upon its own version of the data (in the case of replication) or to determine the effects of such a change to its packages' structures (in the case of no replication). Another responsibility of the AS in this case is to expose its internal structure and IDs to the PS. This information will be used by the PS in its own internal packages to refer to the actual data; notice that this is true even if the data is replicated, because, e.g., in the case of updates, the system must know which block of data changed in order to correctly reproduce the effects of the change in its own copy.

The net effect in this case is a loose coupling between the two systems. On the one hand, the AS still holds the data so it can, to a certain extent, continue doing "business-as-usual", without being restricted by the existence of the PS. On the other hand, the PS can freely perform its operational activities (like replication, repackaging, refreshment, transformations etc), without necessarily informing the AS, so long as the IDs and the structure of the data remain intact; notice that the latter restriction is not negligible, as some PSs may require complete control over the data to guarantee preservation.

The alternative for the AS is to decide that the PS will be the custodian of the data. This option may not be acceptable in many cases because it implies that the PS will have to perform a significant investment in terms of suitable media/apparatus to be used to store the data; this is quite likely to be of prohibitive cost, as a typical modern AS contains large amounts of data. Even if this cost is compensated by the AS, as part of the cost of preservation, this is unlikely to be accepted, given that the AS has already invested in storage systems holding its data. However, such an option would make sense for a new AS that could choose whether to invest on local storage, or to outsource it (as part of preservation). Another possible problem stems from the fact that the AS may be legally bound not to hand over its data, e.g., because of non-transferable rights on the data.

Putting aside the financial and legal concerns, this option has a number of technical merits and drawbacks. First, it causes a tighter coupling between the two systems, implying more extensive changes in the AS's software. An important advantage, from

the preservation point of view, is that it gives enough flexibility to the PS to perform any type of transformation and repackaging it may deem necessary for its purposes. The AS will have to either follow up on the changes, in order for the replicas to remain compatible, or abandon its data.

If the AS abandons its data, then the PS's internal structures, including package IDs, will have to be exposed to the AS. This way, the AS reduces itself to the role of a mediator between the end-users and the data (which are now held by an external system, acting as a back-end). End-users will interact with the AS, which will be a kind of a front-end implementing its services and functionality through the PS's API which provides access to the data. This implies a number of changes in the lower layers of the AS's software implementation, but, depending on the quality of the implementation, this could be done with minimal impact on the AS's upper service layers, so with minimal impact on the users' perception of the AS's services.

If the AS decides to follow up on the changes, a significant amount of interaction between the two systems would be required in order to verify that they both contain the same restructured data. Moreover, some of the services offered by the AS may not be compatible with the new data format, so a significant amount of redesigning may be required. In addition, this option implies the same general problems that appear if the systems decide to replicate the data to begin with: all changes incurred in any system for any reason, will have to be communicated to its counterpart; both systems will have to change their software so as to handle changes in the other system; both systems will have to expose their internal structures and IDs; and a significant amount of interaction will be required for synchronization purposes.

3.2 Discussion on the Data Dimension

Judging the tradeoffs described above, we conclude that, from the AS's point of view, the best option would be to hold the data itself, sending it to the PS only for the initial preservation-related processing. This implies the least changes in the AS's software and the less training, development, testing etc costs for the transition; on the other hand, this option may not be viable if the PS requires complete control of the data.

Alternatively, the PS may choose to replicate the data, but this implies increased initial costs, so a simple pointer to the data's original location may be a more sensible strategy. Finally, the case of the PS being the main custodian of the data should be avoided, because it implies several difficulties regardless of whether the AS keeps a copy or not. However, such a strategy may make sense for a new AS and/or when the PS requires complete control over the data to guarantee the preservation of the digital material.

4 Searching and Retrieval Facilities (Service Dimension)

Regarding the services' dimension, the AS has several options, ranging from abandoning its own procedures and using the procedures of the PS, to totally ignoring PS's procedures and retaining its own; of course, many intermediate possibilities exist as

well. There are certain basic considerations that guide our choices with respect to this dimension, outlined below.

First, the PS's software is highly unlikely to provide any good, high-level services for the user; most probably, it will just provide a generic API, and, possibly, but not necessarily, some very basic user interface. Notice that this is not a shortcoming of PSs, as different archives have totally different and highly specialized needs and procedures, many of which are domain-dependent. As a result, we cannot expect to get the quality of service provided by a standard AS if we use the PS's interface instead.

Second, AS users are used to interact with the AS in a particular way; any changes to the interfaces, procedures and related workflows and, especially, any reduction in the quality of the provided services, are likely to cause problems in the users interaction (as well as costs related to personnel and users training) and should thus be avoided or minimized.

Third, a desirable feature for the PS is that the services provided, and especially those that deal with changes in the data, are implemented through the PS's API. This way, the PS will be able to control accesses to the data and overrule or otherwise react to any action that could jeopardize preservation activities. Therefore, even though overriding PS's API may be possible in some cases, it is generally not recommended.

In conclusion, the most rational choice, in general, is for the AS to retain its public interfaces intact, while using the PS's API. Whether this is possible depends on several factors, including the modularity of the AS's software, the quality of the PS's API and the decisions on the location of the data.

4.1 Factors Affecting the Available Options for the Service Dimension

The available options are severely restricted by the decision on where the data is located. If the data is stored uniquely in the PS, then the AS will necessarily have to use the PS's API for its services. Depending on the quality and generality of said API, and the modularity of the AS's software, this restriction may be transparent or semi-transparent to the end-users, because using the API should just affect the lowest layers of the AS's software. Of course, in practice, many problems may occur, so we generally expect at least a slight, but observable, degradation in the provided services.

But even if the data is stored in the AS, the PS may require that all access (searching, retrieval etc) to the data passes through its own API, rather than being invoked directly by the AS; this would allow the PS to obtain significant control over data accesses that could affect its preservation-related operations. This case is, in essence, the same as the previous one.

Finally, if the data is stored in the AS and the PS does not require AS's operations to use its own API, then the AS may be able to continue accessing the data as before and provide the same level of services to the user. Nevertheless, the PS will require to be notified for any change in the data which could affect its preservation processes; this could be automatically handled by the PS API, if used, so it should not be a major problem in this case, but if the API is overridden, some similar functionality will have to be implemented in the AS's software.

Another parameter that may affect the AS's procedures and workflows is related to the reorganization and restructuring of the data which may occur when the AS initi-

ates its interaction with the PS, as well as whenever the data is close to becoming obsolete. In such a reorganization, we cannot be certain beforehand that the new structure of the data will be compatible with the AS's software and this could have a number of implications, ranging from rewriting the software to redesigning the interfaces and workflows.

4.2 Service Types and Related Interoperability Issues

The services offered by an AS can be roughly divided into two broad categories, namely the searching facilities (which include services like querying, browsing, previewing, reporting etc) and the retrieval facilities (which include services related to the ordering of items, such as cost evaluation and negotiation, data packaging and retrieval, submission, cancellation and postponement of deliveries and possibly even triggering facilities that allow deliveries to be triggered by certain events).

As far as the searching facilities are concerned, it is important to note that these depend on a set of indices and other descriptive information; different information may give different search results (and of different quality). As a consequence, we will have to make sure that only one set of indices is used whenever the data is queried, namely the one in the AS. If the AS decides to abandon its searching facilities and use the PS's API for this job, it may have to transmit its own indices for use by the PS, so that the transition is not perceivable by the end-user.

The retrieval facilities have both internal implications (the AS's management deals with orders) and external ones (users place orders and are interested in orders' history and in tracking down their progress); moreover, the related information is managed diversely among different archives. In order to allow the AS's processes' integration with the PS's API to run smoothly, said API should expose as much as possible of the ordering mechanism, so as to ease the mapping of the AS's internal ordering workflow. In a different case, the AS may choose to just ignore the API, which is an option that should be avoided in the long-term; in addition, this option may not even be available in certain cases, e.g., when the data is stored inside the PS.

4.3 A Proposal for a Smooth Integration of Services

Given the large number of interactions between the AS and the PS, we argue that, in general, it is quite difficult to keep interfaces intact for a long time. In fact, we imagine a first phase in which the AS retains most of its existing services (procedures, protocols, etc), in effect renouncing at taking advantage of the services offered by the PS's API and making only the really essential amendments to its processes. This presupposes that the data is stored locally (in the AS), that the PS does not require the AS to use its API and that the reorganization imposed upon the data by the PS is small enough so that it does not invalidate the standard accessing and retrieval procedures.

Following this initial phase, we expect a slow, but steady adaptation of the AS's procedures, workflows etc into the new reality of preservation, in the sense that the AS will gradually start using the PS's procedures more and more and integrate them with its own; this includes the use of the PS's API. If this process is performed in a

smooth enough manner, it may have minimal effects on users' interaction experience with the AS. Once the integration is complete, we can expect the combination of the two systems to run more smoothly, as the PS's API and internal procedures will guarantee preservation, while the AS's own procedures will be concentrated on exploiting (i.e., sitting atop) the basic and general-purpose processes provided by the PS in order to provide the best possible services to the end-user.

5 Issues on the Preservation Facilities (Preservation Dimension)

Some of the issues related to the preservation dimension have been already discussed as a consequence of choices in other dimensions; however, there are other issues involved as well. As already mentioned, we expect the PS to be in charge of any preservation activities, determining when and how such activities will take place. One open issue is who implements those activities. Of course, this depends on who holds the data; however, if the AS is the data holder then it might be possible that it is not willing to perform the actions indicated by the PS, because, for example, of operational costs caused by such an activity. This could be considered as a good opportunity for the AS to finalize its migration into the preservation environment by handing over the data and taking up the role of the mediator between the user and the data (which is now stored in the PS).

If the data is replicated in both systems, and only one of them reorganizes its data following a notification by the PS, then this could cause the obvious problem that the two replicas are no longer compatible. Such an option should, in general, be avoided, except temporarily.

In the following, we assume that whoever has the data performs the reorganization: if it is unwilling or unable to do so, it has to either pass over the data to the other system, or to abandon the data; of course, the latter option makes sense only in the case of replicated data. Whoever retains the data after this "preprocessing" phase is assumed to perform the restructuring under the instructions of the specialized component of the PS in charge of preservation activities.

If only the AS holds the data (and reorganizes it), then all changes made to the data will have to be considered by the PS so as to upgrade the relevant descriptive information (packages) accordingly. Of course, there is the issue of how the instructions related to the preservation activities, issued by the PS, are passed over to the AS, and what kind of software will implement such instructions. This software may be provided either by the PS, as part of the interoperation process, or it may be an application-specific specialized software, developed by the AS itself.

If only the PS holds the data (and reorganizes it), then, provided that the reorganization does not result to a severe restructuring of the data, the process is more-or-less transparent to the AS and the end-users, as the AS is assumed to access it through the PS's API. Of course, this option implies that the AS will have to abandon its data.

The case of both systems applying the reorganization is rather problematic in the sense that such a reorganization would be rather costly in terms of resources required, so its replication should be avoided, if possible. Of course, this is an inherent problem in replication and should be compared against the option of having one system do the

job and then communicating the changes to the other. Another disadvantage of the simultaneous approach is that, even though theoretically they should both end up with an exact copy (replica) of the new version of the data, it is difficult to guarantee that this will actually happen in practice.

In all the above cases of reorganization, extreme care should be exercised in order to avoid the various processes (services) and interactions to become obsolete as a result of the reorganization. This is less of an issue when the interactions are performed through the PS's API, but it's an issue in both cases nonetheless. In some cases, this may be unavoidable, so some modifications in the relevant software and processes may be necessary.

6 General Scenarios for Interoperability

The above issues give rise to a multitude of possibilities regarding the interoperation between the AS and the PS. Here, we explicitly study three different scenarios, each described in a different subsection below in increasing order of required amount of integration between the two systems.

6.1 The Preservation Consultant Scenario

This scenario applies when the AS does not want to hand over either the processes (services) or the data to the PS. In this case, the balance is tipped over to the AS side, in the sense that the AS makes minimal compromises and commitments in its interaction with PS; this, in turn, has the cost of making preservation a little more difficult to guarantee for the PS.

This scenario dictates that the PS gets the data, as part of the initial interaction, so as to build the relevant packages and create the various descriptive information required for preservation purposes, but then discards the data, storing only a pointer to the original copy, held in the AS; the AS will have to expose its IDs, internal structure and other relevant information about its storage system, as well as to notify the PS on any relevant changes to the data. In this scenario, the AS does not use the PS's API, but continues to perform its operations as it used to. Whenever the time for a reorganization comes, the AS is responsible for implementing such a reorganization, using the instructions and know-how of the specialized external organization (PS).

This scenario has the advantages of a loose coupling between the systems, minimal changes for the AS's services and procedures, and transparent, or almost transparent, transition to the new environment, as far as the end-user is concerned. One disadvantage is that the PS has several restrictions as to the available ways of reorganizing the data, and no control upon it; this may cause difficulties in guaranteeing the preservation of the digital material and may even make this impossible for certain cases (depending on the PS's algorithms).

The "preservation consultant" scenario makes more sense for an AS with very specialized and difficult-to-capture procedures, an AS legally bound by non-transferable rights on the data, or in cases where the PS is not fully trusted (e.g., because it is the result of an experimental research project rather than a commercial product). How-

ever, the various advantages of this scenario related to the smoothness of the transition for both the AS and the end-users make it attractive for other cases as well.

6.2 The Standard Commercial Scenario

This scenario attempts to strike a balance between the AS's needs for a smooth transition to the preservation environment and minimal changes in its procedures, and the PS's needs for complete control over the data.

According to this scenario, the AS hands over the data for the initial packaging, as usual, but remains the main custodian of it, so the data is retained locally as well. The PS may replicate the data following the packaging, or it may just keep a pointer to its original location (the latter is probably better for most cases, due to lower storage and handling costs). All services provided by the AS will have to pass through the PS's API (despite the fact that the data is held in the AS), which allows the PS to monitor operations over the data. Preservation activities are performed by the AS, under the consultancy of the PS, as usual.

We argue that this scenario is what most PSs will impose upon their AS customers, as it gives a significant amount of flexibility for both the AS and the PS, allows the AS to retain control of the data, while the PS can monitor any operations upon it to the end of reacting to any operation that could jeopardize the success of its preservation activities. It may require a number of changes in the AS's processes and workflows, but the effects of these changes can be smoothed if the two organizations agree to apply the "preservation consultant" scenario for the initial period of their interaction, then gradually passing over to the "standard commercial" scenario.

This scenario is probably the most suitable for nearly all cases, as it strikes a balance between a number of conflicting requirements. It is especially suitable for a new AS, in which case the various processes could be designed with the PS in mind right from the start; in this case, the option of handing over the data as well could be considered, as there is no already-made investment on storage media that will be made void by such a decision.

6.3 The Mediator Scenario

This scenario reduces the AS to the role of a mere mediator between the end-user and the data (which is stored in the PS). It gives maximum flexibility to the PS, giving it total control over the data and the procedures and allowing it to perform any operations necessary to guarantee the preservation of the data, but requires a number of significant changes in the AS to achieve integration; moreover, further changes may have to be made following the reorganization of the data by the PS.

This scenario dictates that the AS hands over all its data to the PS, which is totally free to perform any kind of packaging and reorganization necessary in order to guarantee the preservation of the data. Following this packaging, the AS's processes are redesigned in such a way as to enhance the general-purpose processes (and API) of the PS, to the end of improving the users' interaction experience.

The main disadvantage of this scenario is that it is very likely that both the AS's processes and workflows, and the users' interface with the AS may have to be redesigned to some degree. Also, the reduction of the AS to a mediating role and the handing over of its data may be unacceptable, illegal or impossible in certain cases. However, this scenario implies the best possible preservation guarantees to be given by the PS as it has complete control over the data.

This scenario would make sense for an AS without any specialized internal procedures, because non-specialized procedures are likely to remain unaffected by data reorganizations and will anyway be adequately supported by the generic API provided by the PS. This scenario, like the previous one, is also suitable for a new AS. Finally, it should be considered in cases when the maximum possible guarantees regarding preservation success are required.

7 Concluding Remarks

We studied the case of an archive which decides to preserve existing digital material by outsourcing preservation activities to an externally managed, specialized organization. Such an option would make sense in order for the archive to take advantage of the preservation organization's specialized resources and know-how, in order to perform a more successful, efficient and cost-effective preservation.

We identified a number of issues associated with this decision, as well as the various tradeoffs involved. We argued that the related choices affect the quality of the preservation activities, the effects of the transition upon the archive's well-established and tested procedures, workflows, strategies and other processes, as well as the effects on the quality of services and related interfaces provided to the end-users. The most viable set of choices depends on the application at hand, so a number of scenarios were presented that make good intuitive sense for particular cases.

Our study indicated a number of good practices related to the interaction between archives and preservation systems. More specifically, new archives should try to undertake preservation activities as soon as possible. This will guarantee that all processes, workflows, software etc will be designed with the peculiarities and the difficulties introduced by their integration with the preservation system in mind right from the start, and allow the training of users and personnel directly for the preservation-enhanced system, making the whole process much cheaper and more efficient.

Old archives are likely to encounter problems in their effort to integrate their operations with an external preservation system; despite that, given that most digital archives are in urgent need for preserving their material, preservation is a venture that needs to be undertaken as soon as possible. To facilitate transition, archives could consider an initial phase of retaining their data and services (as in the "preservation consultant" scenario), while gradually upgrading their software and internal processes so as to use the generic processes provided by the preservation system (as in the "standard commercial" scenario). Such an upgrade should be made as transparently as possible, and should avoid any degradation of the quality of services provided to the end-user.

On the other hand, preservation systems should design their preservation-related activities and algorithms in such a way so as to have minimal (to the extent possible) effects on the structure of the original data; intensive reorganization activities overrule some of the more attractive choices for interaction between the two systems and may cause significant changes on the archive's provided services. Moreover, great attention should be paid to the API a preservation system provides, because using a well-designed API for data access will imply minimal effects on user interfaces and archive's services, cleaner preservation procedures, increased control over the data for the preservation system and, consequently, more successful preservation activities.

Acknowledgements. This work was partially supported by the EU project CASPAR (FP6-2005-IST-033572) and was carried out during the second author's tenure of an ERCIM "Alain Bensoussan" Fellowship Programme in CNR-ISTI. Part of this work has benefited from helpful discussions with Francesco Furfari.

References

1. CASPAR: Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval. EU funded project (FP6-2005-IST-033572). http://www.casparpreserves.eu
2. Cheney, J., Lagoze, C., Botticelli, P.: Towards a Theory of Information Preservation. In Proceedings of the 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL-01) (2001) 340-351
3. ERPANET: Electronic Resource Preservation and Access Network. EU Funded Project (IST-2001-3.1.2). http://www.erpanet.org/index.php
4. Factor, M., Naor, D., Rabinovici-Cohen, S., Ramati, L., Reshef, P., Satran, J.: The Need for Preservation Aware Storage: A Position Paper. In ACM SIGOPS Operating Systems Review, 41(1) (2007) 19-23
5. Flouris, G., Meghini, C.: Steps Towards a Theory of Information Preservation. To appear in Proceedings of the International Workshop on Database Preservation, Invited Talk (2007)
6. ISO 14721:2003: CCSDS 650.0-B-1: Reference Model for an Open Archival Information System (OAIS). Blue Book, Issue 1 (2002), available at: http://ssdoo.gsfc.nasa.gov/nost/isoas/ref_model.html
7. Lavoie, B.F.: The Open Archival Information System Reference Model: Introductory Guide. DPC Technology Watch Report 04-01 (2004)
8. Lesk, M.: Data Preservation: It's a People Problem. To appear in Proceedings of the International Workshop on Database Preservation, Invited Talk (2007)
9. Lynch, C.: Canonicalization: A Fundamental Tool to Facilitate Preservation and Management of Digital Information. D-Lib Magazine 5(9) (1999)
10. Mellor, P., Wheatley, P., Sergeant, D.: Migration on Request, a Practical Technique for Preservation. In Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries (ECDL-02) (2002) 516-526
11. Pace, A.K.: Coming Full Circle, Digital Preservation: Everything New is Old Again. In Computers in Libraries, 20(2) (2000), available at: http://www.infotoday.com/cilmag/feb00/pace.htm