
Volume 8 - Issue 4, 2008 - Cover Story: Cost-Effectiveness in the Imaging Department

Create Custom Software for Your PACS: Increase Functionality, Decrease Costs

Author

Dr. David Harvey

Medical Connections

Swansea

Wales, United Kingdom

dave@medicalconnections.co.uk

Since the introduction of PACS, vendors have gone to great lengths to ensure that the products they supply provide the users with the functionality they most need, from both a clinical and administrative perspective – and generally they succeed. However, no two hospitals are ever exactly alike, and some of the specific requirements for individual hospitals may simply be too small and esoteric for the mainstream vendors to get involved.

In this article, I explain how the creation of a simple and easily built custom software solution can increase functionality in a cost-effective manner. In many cases, functionality errors or problems relate to the interfacing between equipment from different suppliers, and they commonly result from minor errors or discrepancies in the interpretation of standards e.g. DICOM, between different vendors.

Common Interoperability Problems

Problems may not have shown up when the vendors did their own internal testing against their own systems, but are found by users in mixed vendor environments.

For example:

- Some imaging equipment, quite legitimately, sends images to PACS using a separate association for each image, but a PACS from a different vendor misinterprets the end of an association as the end of a study, and handles subsequent images inappropriately.
- Likewise, whilst DICOM does not specify the order in which images should be sent, as ordering is defined by internal attributes, some workstation/PACS combinations insist on displaying images in the order received or even worse, sorted by inappropriate keys such as their UIDs.
- Some systems expect to find only their own defined terms in fields that are allowed to have variable text.
- Some systems fail to complete mandatory DICOM fields, or, equally badly, expect to find data in optional fields, and fail if it is missing.

In all these cases, and many more, the end result is that users are unable to use the images or other data, such as radiotherapy records, in the way that they would wish and expect. Large suppliers can be very slow to admit that they have made mistakes in their implementations and sometimes even try to “justify” their departure from standards. Even if they do admit that they have a problem, they are unlikely to be able to produce a timely fix, so what can a user do?

What is the Answer?

A solution used by an increasing number of hospitals is to write or buy a small simple custom solution for their own needs, addressing just the problem at hand. Due to the way DICOM works, this often fits into the workflow relatively easily by inserting a “modifier” programme between the items of equipment.

This receives the images or other objects from one system, and makes the changes required before sending them on to the final destination. In my experience, the following features impact on the success and acceptability of the project:

1. Use an Established Software Toolkit

© For personal and private use only. Reproduction must be permitted by the copyright holder. Email to copyright@mindbyte.eu.

Whilst the changes to be made to the data are normally very minor, or even zero in the case of applications which simply rearrange images onto different associations or a different order, the whole complicated DICOM protocol needs to be handled.

Writing such a project from scratch would therefore be a very large undertaking, and it is important to use an established software toolkit (proprietary or open-source) to handle those aspects, leaving the writer to concentrate on the required functionality.

2. Use the Resources at Hand

Of course, the writer needs to have the required skills to understand exactly what they are doing and why. In many cases these skills can actually exist in-house, often in the medical physics department, and such a project then provides an excellent staff development project as well as a solution. Where such skills are not available, then there are plenty of small software houses who can do the job for relatively small cost.

3. Keep the Project Within Reasonable Limits

It is important that the scope of the project should be well defined. All too often, once people realise that a facility like this is being built, they find other "nice features" that they would like to add, and if care is not taken, then the scope can grow unmanageably.

4. Quality is Key

There needs to be a suitable quality assurance process to ensure that the resulting data is as expected. Of course, not all projects fit so easily into such a simple workflow, and there are many other project types for which custom applications can be used, of which the most common involve unusual combinations of equipment.

These situations typically occur where there has been reorganisation, sharing or merging of services, such that information flows may not follow conventional pathways. For instance, it may be necessary for a workstation or acquisition modality to connect to more than one PACS, or even to more than one RIS, either simultaneously or as alternatives.

Likewise, it may be necessary to connect one PACS to another for data migration, either due to replacement, or because of consolidation of departmental systems into a single PACS.

Such unusual data transfers can be very straightforward when viewed from a logical perspective, and can be equally simple at the technical level as they can normally be achieved using standard interfaces on the existing equipment.

However, every single such application is likely to be different, and small but significant data transformations (e.g., patient ID, accession number, etc.) commonly need to be incorporated into the process.

Such projects are ideal candidates for custom software development, and provided that the above constraints for project scope and management are followed, then the same benefits can be obtained.

Assessing the Cost of Custom Software Development

Whilst custom software development sounds as if it would be expensive, the costs are, in practice, relatively small in comparison to the equipment with which it is used, and most projects should end up costing less than 20,000 euros, including a software toolkit, the inevitable "extra PC" to run them on, development time (whether in-house or external) and testing.

Conclusion

In conclusion, a "standard" PACS installation, with no unusual requirements, no legacy systems, and perfect standards implementation by all vendors should work "off the shelf" with no need for custom software, but if you find that you do have specific needs which cannot be met by your PACS or modality vendors, then the option of custom software development should certainly be considered.

Published on : Wed, 1 Oct 2008