A Position Paper for Second International Workshop on Component-Based Software Engineering

Strawman section – Principles of Adopting Component-Based Software Engineering (CBSE)

1 Introduction

This paper speaks to section 2 "Practices of Adopting CBSE" of the strawman outline. Most of the topics covered fall within Organisational Issues but one candidate new (sub)section is identified.
Top of the Document

2 Which Components First?

Broadly, components can be divided into two groups (as can most of the rest of human experience, at least by someone). The author proposes:

- Business Components - those that the business itself would recognise - e.g. customer, invoice, account, assembly.
- Infrastructure Components - those which (probably) only the IS community would recognise - e.g. audit (in transaction terms), security, error message handling.

CBSE will really show returns for an organisation when its Business Components are identified and "engineered". Unsurprisingly, there are several issues to be addressed.

Firstly, the business needs to be educated about components and the benefits they bring. Secondly, there may be a cultural barrier to overcome about the role of software viz. the need to see software as infrastructure (notwithstanding the author's classification of components above).

Once these two have been overcome, there is a problem, which has analogous forms in more traditional forms of software development. In an organisation of any size, getting disparate user groups to agree on the definition of some Business Components will be a non-trivial task. For example, in a distribution company or a postal authority, there will be several views on an address or even on a postal code. Similarly, in a railway company, the concept of "train" can be an engine (to the maintenance yard) and the 07:15 Monday-Friday from Sheffield to London (to traffic planners and passengers).

The author does not claim this list is exhaustive.

Infrastructure Components may avoid some of the problems associated with Business Components (e.g. there will undoubtedly be a group of developers in an organisation who will want to bring new technology in) but they have their own traps.

The first of these is the effect on the business users. If too much emphasis is made of the introduction of infrastructure components, the business may see components as the latest technical fad. Thus, the introduction of infrastructure components must be handled carefully. However, if managed well, this approach could show a technical "proof of concept". Also, components which are rightly seen as infrastructure but which do interact with business users (e.g. authentication) should render themselves capable of explanation in a way to help convince the business of the benefits of CBSE.

The next problem for the infrastructure component approach is funding. This is also a problem for Business Components but the difficulties are similar in both cases, and may be more difficult to solve if Infrastructure Components are the first to be done. It is likely that a component will cost more to develop than the same functionality written in more traditional methods. This will be difficult to sell to the first project (and others) that uses this component. A possible model comes from a technology (or product) development budget approach where a central (within the IT function) body carries the "risk". This is discussed in more detail in Costs/Benefits/Funding below.

The current marketplace is populated far more with technical (e.g. ORBs) components than with business components. This could suggest that it makes sense to start with infrastructure components. However, as outlined above, the real returns for an organisation will come when business components are deployed. In the end, the approach must be determined by the benefits sought by a project and/or organisation, and the starting point for the migration to CBSE.
Top of the Document

3 Costs/Benefits/Funding

3.1 Objectives

It is vital to be thorough on investigating costs when putting together the business case for the adoption of CBSE. It is also important not to overestimate the benefits, especially for the return on investment. If the costs and benefit are not carefully researched and presented, subsequent CBSE projects may find it very difficult to get funding, and the adoption of CBSE may fail.

An organisation adopting CBSE must also be very clear on objectives. For example, reuse looks very attractive, and if CBSE is done properly, it will come. However, it almost certainly will not come early on. Similarly, flexibility for change should come from successful adoption, but not immediately. The best way to frame objectives is to find out what other organisations have done, where they have succeeded (and struggled, or even failed) and work out what is best for your organisation from these experiences.
Top of the Document

3.2 Possible funding structure

To a large extent, funding structures for CBSE will depend on how the organisation typically invests in IT change and infrastructure (including people and methods).

What will be hardest to achieve, and least likely to succeed, is to ask individual project teams to deliver components out of their funding. The demands on projects make this an almost impossible goal, from the outset. It is better to have some centrally funded activity, whichever way the organisation decides to start with components. (See discussion in Staff/Organisational Issues below).

If the organisation starts on infrastructure components with a single specialist team, (see also the discussion on transition in Staff/Organisational Issues below) it could charge project teams. The charge to projects must be a little less than the team would have spent developing the functionality in a more traditional way. The other approaches can be funded in similar ways, but crucially relieving the project teams of the burden.

For any of these to work, the CBSE team needs some investment until it starts "making a profit". This should come from a technology or product development budget.

A further aspect of funding to be considered is for maintenance costs. See 5 below for a brief discussion of Service Management aspects of CBSE.
Top of the Document

4 Staff/Organisational Issues

4.1 Education and Training

There may be management resistance to education and training - especially as it will involve users at some stage. Technical staff will need a change of mindset as well as technology education - even if they are already using Object Oriented techniques. Giving this sort of training will bring with it a risk of staff turnover. Management must be convinced that the problem can be managed, but the risk still exists and needs to be managed once the adoption is underway. Project Managers will also need training, because CBSE is different for them, as well as for technical staff.
Top of the Document

4.2 Organisation

4.2.1 An Object View

There are different models of adoption from which a choice must be made. Alan O'Callaghan of De Montfort University in the U.K. has written extensively on the issue of making the transition from traditional methods to Object Technology, and the principles apply in CBSE as well. In particular, one paper speaks to the organisational issues to be faced [1]. Where this paper refers to O'Callaghan's article, no attempt to "convert" to from objects to CBSE has been made. However, the principles still apply where an organisation has not moved to Object technology but is considering moving from traditional structured methods straight to Components. This is not as rare as might be imagined. Object technology has not penetrated the market for administrative or commercial software and many such organisations are still deploying Client-Server applications.

The models are variants on a centre of expertise being formed which over time effects skills transfer into the rest of the organisation.

"Loading" the pilot team
Here, a pilot project is chosen with some key well-respected developers, along with more junior staff doing well on

other projects. The project is chosen to be important to the organisation's goals, without being one by which the organisation lives or dies. After the project is complete, the team members will "spread the word" as they disperse through the organisation.

Technology Reception Team
Here, the pilot team approach is used in a much more specialised way. As well as the goals above, the team starts out from the very beginning to train in-house "mentors" for the new approach. This needs to be part of a well-formed migration plan for the organisation. The plan will include not only how the reception team will function, but also the technology (in its widest sense) transfer mechanisms to spread the new methods into the rest of the development (and business) organisation. This transfer, in turn, will need to define what skills and competencies will be needed not only for the staff who work in the new way, but also (perhaps crucially) for those who will effect the transfer. The best technicians might not make the best mentors!

The Object Centre approach
Here, a specialised group is set up as a centre of excellence in the new technology for the organisation. As well as getting the new approach started, over time staff will rotate through the Object Centre. This will spread the new skills etc. through the organisation.
Top of the Document

4.2.2 The "Component Centre"

The CBSE analogue of the Object Centre is the "Component Centre", and the analogy is fairly direct. When considering CBSE specifically, the Component Centre may also "harvest" and "mine" new (even legacy) applications. This takes the burden off the project team to think too hard about components. Instead, the Centre works alongside them, or even slightly after deployment, and examines the application for potential components. The Centre then carries out the work to convert the identified element to a component.

One way of achieving this is to put Component Centre staff into the project team at the equivalent rank of Project Manager and giving them a brief both for harvesting code for reuse and for spotting opportunities for reusing existing code. While the project team itself concentrates on meetings its own application/system deadlines, the CBSE person offers to "buy" suitable components for rework to make them reusable, and to "sell" existing repository components to the project. The implication is some kind of cost centre relationship between the project and the CBSE team.

In addition, the Component Centre can take on the management of the in-house component repository. Indeed, during the transition, it should do so, and there are reuse exemplars that speak to this.

The Centre will keep an eye on the component marketplace so that components can be bought, and, where appropriate, sold for the organisation. This runs alongside the Centre's role in ensuring the order in which an organisation seeks to source each "new" component. Acquisition, then adaptation of existing components should always be considered as preferable to construction.

Top of the Document

4.3 Staff Sensitivities

Some existing roles will almost certainly change, while new roles will be created. People dislike change (not a new insight, the author wishes to point out) so these new structures and practices must be sold to staff. Watch out, for example, for the impact of the Component Centre member seconded into a project team as described in 4.2.2 above.

Life is never simple, however, so organisations making this change may well find that staff feel uncomfortable about change, and yet jealous of those being given the opportunity of new technology. This is, however, not a new

problem. Management experience shows it recurs, with, perhaps, only the backdrop being new.
Top of the Document

5 Support Issues

The Service Management (in its broadest sense – for example as defined in the IT Infrastructure Library) function will be concerned. There are issues about how defects in production components are dealt with when the components are bought in. This problem differs from the issues connected with supporting a bought-in package.

A model for the maintenance costs of components needs to be developed before any components go into production. This will be complicated by the concept of ownership of components, especially once they start to get reused.

A further complication arises in the field of Change Management. Most production defects can be attributed to change in some shape or form, so organisations thinking about Service Management in any depth are trying to understand and manage change. Change will be more difficult to understand, as components get more and more widely used. And bought in components may, at first inspection, make it even more difficult. As with all development methods, this just serves to underline that so-called non-functional requirements need to be considered as early as functional requirements in any project.

The Strawman outline would benefit from a Service Management section.
Top of the Document

6 Project Management

Other workshop participants' papers under the Project Management section may discuss these issues. The issues the author sees with CBSE adoption here result from the fact that the paradigm is so new. There will be a need for new "rules" for managing software projects because there will be new roles and some merging and mixing of existing roles.

The key point here is that Project Managers will need to be convinced that CBSE is different, and then they will need the relevant training.
Top of the Document

7 Metrics

There is also an issue (not that the software industry has got this right yet!) in estimating. By its very nature, estimating is only good when there is a good body of data from which to extrapolate. By definition, a new paradigm cannot have such data, especially within one organisation. Some mechanism for sharing estimates (and actuals, of course) across the industry must be found.

Having said that, some existing techniques may still be valid with, perhaps, just a change of scale. Also if an organisation does construct small components, a database of useful past experience may be built up quite quickly.

Top of the Document

8 References

[1] “Organising for the ‘Personal Shift’”, Alan O’Callaghan, Object Expert 1(3)

Chris Woodhouse
Post Office Research Group
The Post Office
Manor Offices
Old Road
Chesterfield
S40 3QT
U.K.

Tel: +44 (0) 1246 214830
Fax: +44 (0) 1246 523325
Email: woodhouc@postoffice.co.uk

(End of document) Top of the Document