

Sizing Infrastructure Software (Middleware, Components)



HOME



BACK



NEXT

By
Tony Rollo SMS

Health Warning



Do Not do this for Application Sizes

Why Size ?



- To understand Diversity of environments
 - For estimation
 - For Performance assessment
-
- Different Teams
 - The HCI front end may be a different team to the server
 - The middleware will be a different team that the applications
 - Productivity
 - HCI software can be quite different to the application server
 - For middleware is usually lower than application software

Infrastructure Software?



Software that provides a variety of services to a range of software applications

Provides communications infrastructure between applications

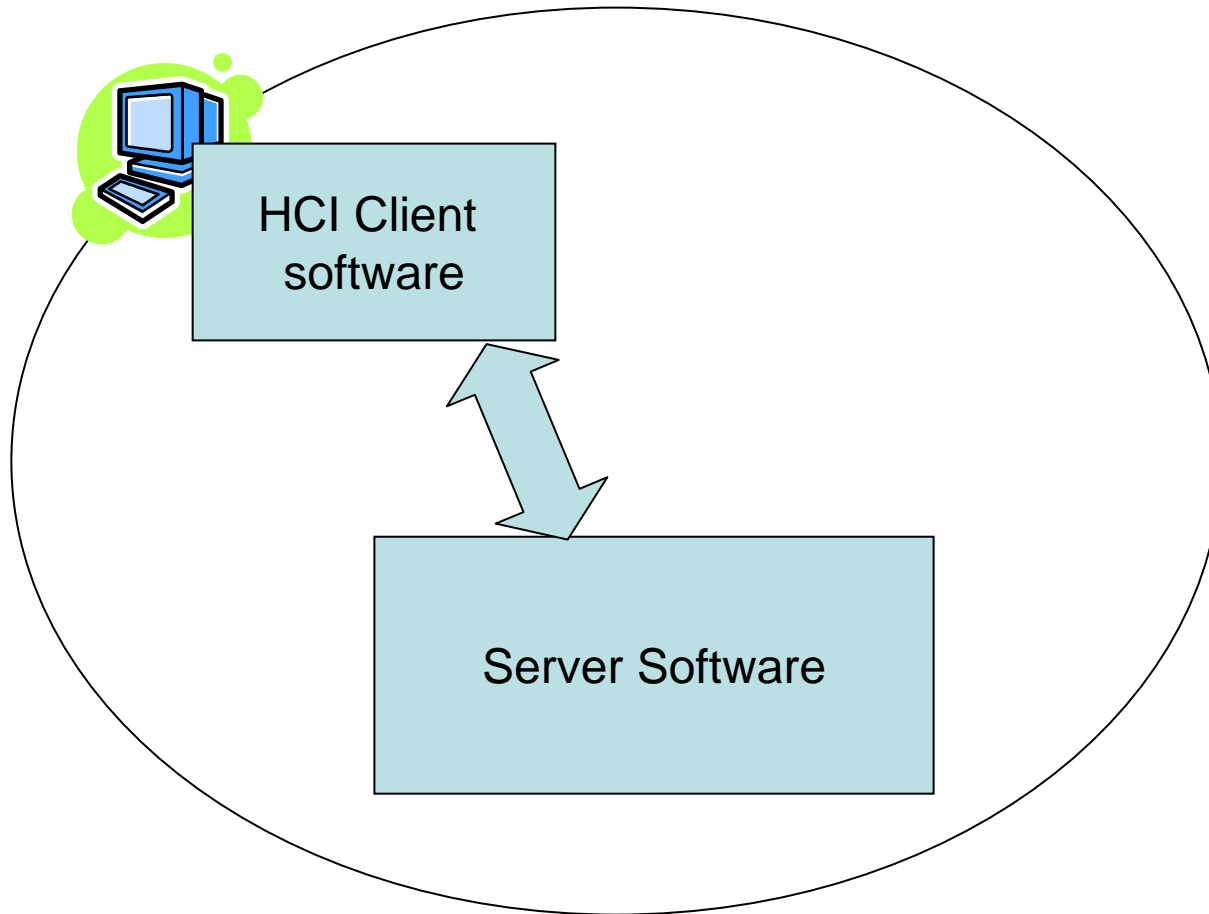
In general it does not interact directly with a user

Essential in service oriented architecture environments

I include major software components – because they can also be sized using similar arguments fro the same reasons

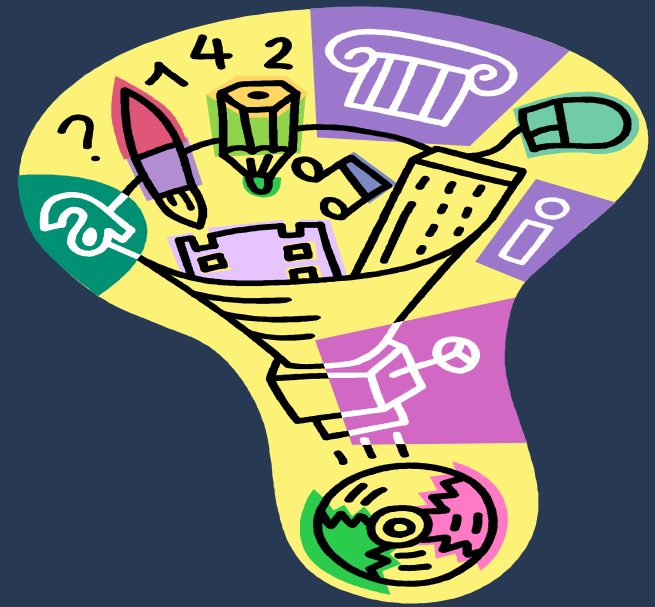
Some forms are often called middleware

What are Components?



Major Software Components

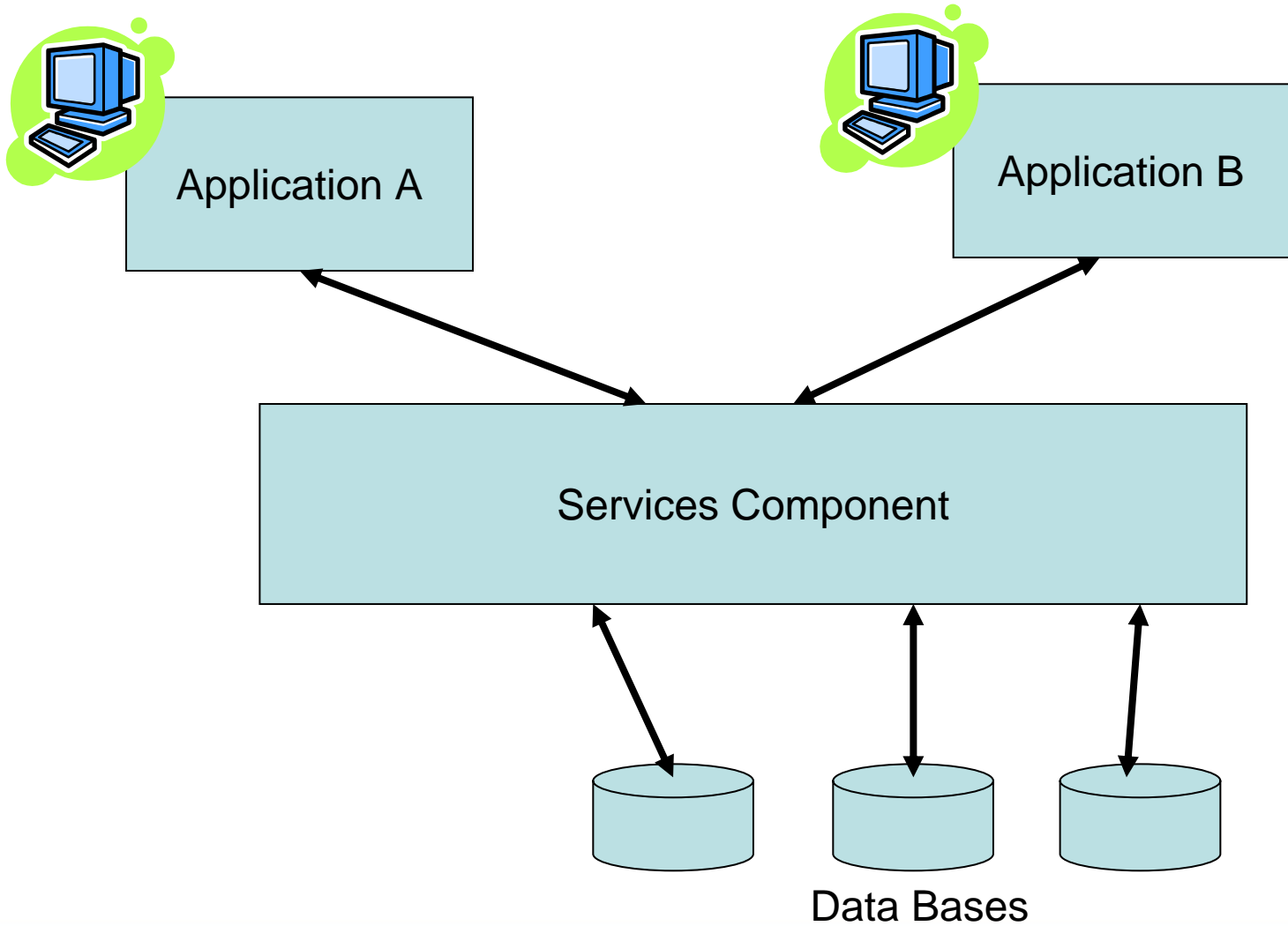
Understanding modern development environments



it is sometimes difficult to understand the variation in the productivity of the multiple environments encountered within one application

- several different technologies may be being employed within a single application
- development often uses different teams or Indeed companies.

Infrastructure



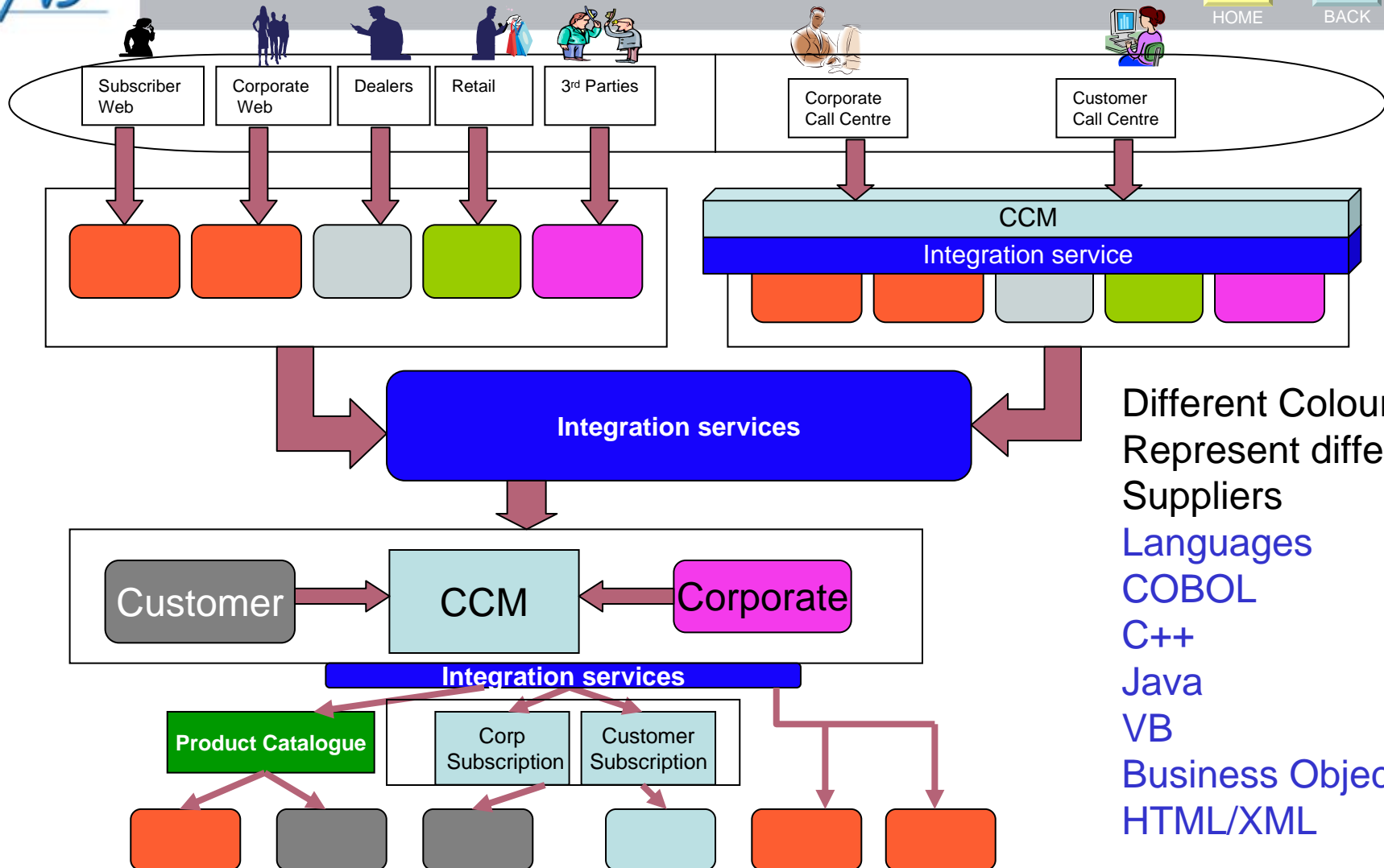
Integrated Software



It can be difficult to separate applications

- what were once distinct and clearly separate can become highly integrated and identifying a particular application may be difficult.

Multiple Technologies in an Application



Outsource Service Level Agreements



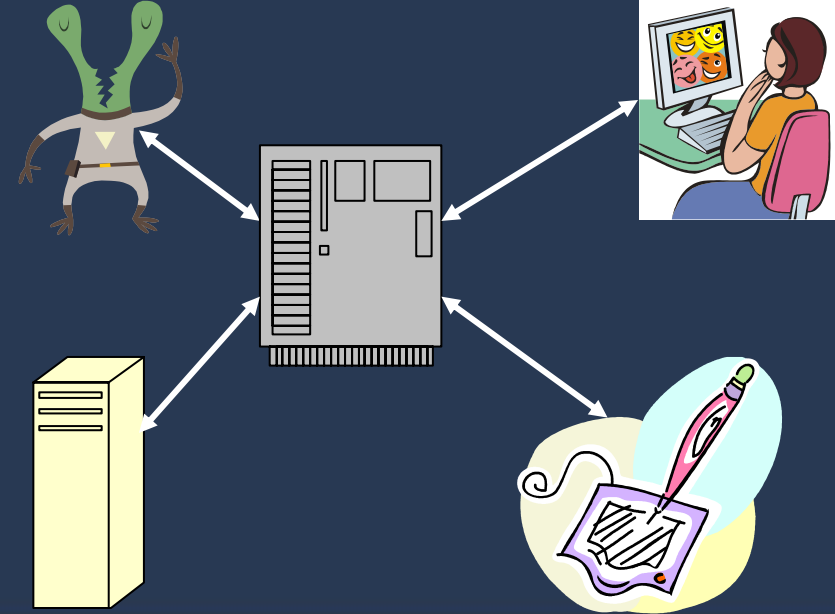
- Some outsource contracts have SLAs which are
 - based on the use of Function points
 - determine if productivity or cost goals are being met
- These contracts start by
 - measuring a range of projects
 - arrive at the baseline cost/productivity data
 - On which the targets are based
- common feature of such baselines
 - analysis of too few projects
 - restricted range does not always reflect
 - the diversity of enhancement environments that may be encountered within a single application

IFPUG CPM R 4.2 Rules



Does anything in the CPM prohibit the sizing of middleware?

User



Any person that specifies Functional User Requirements and/or any person or thing that communicates or interacts with the software at any time.



Boundaries



The following rules must apply for boundaries:

- The boundary is determined based on the user's view. The focus is on what the user can understand and describe.
- The boundary between related applications is based on separate functional areas as seen by the user, not on technical considerations.
- The initial boundary already established for the application or applications being modified is not influenced by the counting scope.

The Users View



A *user view* represents a formal description of the user's business needs in the user's language. Developers translate the user information into information technology language in order to provide a solution.

Is a description of the business functions

- Is approved by the user
- Can be used to count function points
- Can vary in physical form:

catalog of transactions,
proposals,
requirements document,
external specifications,
detailed specifications,
user handbook





Data Functions

- Infrastructure handles data and references data
- So we can identify logical files as for any application

Transactional Functions

- Transactional function exist in the intercommunications between the infrastructure and other software components
- We can identify EI, EQ and EO among theses interactions

Practical Investigation



At the request of a client an investigation was carried out to determine if the inclusion of the FP size of a middleware component would provide a better fit with the effort expended.

Considerations



The purpose was to size the requirements of an enhancement - added, changed or deleted functionality

The most difficult assessment dealt with the boundary definition.

In this discussion it was decided to include the two Function Point methods – Cosmic and IFPUG FP.

The Cosmic defines – Boundary and Layers. Middleware is a layer and a boundary exists between layers.

For the IFPUG method an assessment needed to be made to see if a boundary could be defined



- Biggest issue is the identification of the middleware as a separate boundary.
- This needs to be done with care.
 - If the middleware has only been introduced as a technical interface between front end and back end, and is not recognized by the user.
 - Not having maintained data that are recognized as other than copy or images of the data stored elsewhere it should not be identified as a logical boundary –
 - but purely a technical implementation. In the study this was not the case.

Results



Measurements were conducted and the agreed results are as follows

- IFPUG FPA Size is 177 FP
- COSMIC CFP size 192 CFP
- This amounts to a difference of some 15 FP which is under 8% this is within the generally accepted level of accuracy of these methods (10%).

Conclusions



The authors concluded that there was no reason why middleware applications should not be counted

We recommended that the IFPUG method should be used – there is no point in having two methods

Consequences



A study of the effect on project costing would need to be undertaken by the client and outsourcer.

Probably baselines would need to be recalculated – though this could be done by accretion 2 such projects have already been sized.

Initial indications are that this accounts for a significant proportion of the variation apparent on the basis of the baseline data.

Thank you for listening



Any Questions?