

projectName

**Solution foundations  
System area definition (SAD)**

PHASE 2: FEASIBILITY AND FOUNDATIONS

# Purpose of this document

DSDM handbook 8.4.3.2 System Area Definition; Appendix C 3.3.2  
Document template version 0.6

* To provide a common understanding of the technical architectures to be used during development and deployment of the solution including:
  + Hardware/infrastructure
  + Software architecture
* To describe the target environment for the solution and (if different) the development environment.
* To provide an outline description of anticipated developments in areas such as:
  + Hardware (i.e. the infrastructure, processing, storage, networking etc.) for both development and deployment.
  + Software (i.e. the major software objects or components - both process and data - and their interactions).
  + Information security (eg access policy, access control etc.).

# Quality criteria

* Is the architecture appropriate for the requirements?
* Have the risks of the proposed architecture been properly considered? In particular, are all components of the proposed architecture available and mutually compatible?
* Is the architecture defined at an appropriate level, so that it will not be too vulnerable to change as the project progresses?
* Will migration from the development platform/medium to the target platform/medium be able to occur easily? If not, are all foreseeable problems identified?
* For software projects:
  + Is the outline software architecture sufficiently well-defined to give solution developers a high-level view of the proposed computer system?
  + Has advantage been taken of any opportunities for reuse of existing components?
  + Can the architecture be expected to cope with performance, capacity and resilience requirements?

# Document sign-off

|  |  |  |  |
| --- | --- | --- | --- |
| RACI | Role | Name | Date |
| Responsible | **Technical coordinator** |  | **yyyy-mm-dd** |
| Accountable | **Project manager** |  |  |
| Consulted | Business analyst |  |  |
| Verified | Team leader |  |  |
|  | Solutions developer |  |  |

# Revision history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Version | Reason for change | Status | Date |
|  | 0.1.0 | Initial draft | Draft | yyyy-mm-dd |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 1. System architecture

## 1.1 Architecture overview

***Construct a high level model (diagram or series of diagrams) that clearly identifies all the hardware components of the architecture and illustrates how the software application is overlaid on the chosen hardware. Verify that it is complete in terms of both hardware (identifying all technical platforms and relationships between them) and software (identifying all interfacing systems, all application software components and appropriate middleware and database software).***

***Note: in the description above, the word component refers to a “part of” the architecture or system and not specifically to reusable components that the project may produce or re-use.***

Architecture overview…

## 1.2 Hardware components and relationships

***By reference to the architecture overview above, describe in as much detail as is available:-***

* ***Each of the hardware components of the system that are illustrated in the previous diagram (i.e. describe the ‘entities’ on the diagram – the boxes or other shapes).***
* ***The physical interfaces between them under predicted implementation scenarios eg office access and home access (i.e. the lines on the diagram).***

***Briefly explain how the proposed architecture fits with the organisation’s strategic architecture (if that exists). Identify risks and issues related to deviating from current technical architecture or policy.***

***Note: If the purchase of new hardware or the installation of new infrastructure is required to support the deployment of the proposed system, it needs to be described in sufficient detail here to provide likely hardware specifications and associated costs.***

Hardware components and relationships…

## 1.3 Software components and relationships

***By reference to the architecture overview (1.1) above, describe at a high level:***

* ***The major software components of the system in terms of their purpose (eg application server, data server, web server, transaction manager, etc.)***
* ***How these components relate to each other in terms of software. Describe the involvement of any middleware in the architecture. Describe the interfaces between the software components under predicted implementation scenarios eg office access and home access.***
* ***How the proposed system interfaces with existing applications and any others currently under development.***

***Briefly explain how the proposed architecture fits with the organisation’s strategic architecture (if that exists). Identify risks and issues related to deviating from current technical policy with regard to middleware etc.***

Software components and relationships…

## 1.4 Non-functional requirements

***Describe the key areas of non-functional requirements that are likely to be important to the project and how the architecture described above supports the non-functional requirements. If possible prioritise these on a scale of 1 to 3 as indicated in the table. Non-functional requirements were first considered in the technical constraints section of the feasibility report so refer back to this as a starting point.***

***Note: At this point it may not be possible to describe the non-functional requirements in the detail indicated in the comment text within the table. Don't worry too much about this but be sure to make a note in the table that further investigation of this requirement is needed during the functional modelling phase.***

***Use a priority scale of 1. Critical, 2. Important, 3. Unimportant.***

|  |  |  |
| --- | --- | --- |
| **Non-functional requirement** | **Priority** | **How is this supported by the architecture?** |
| Usability |  |  |
| Performance |  |  |
| Capacity |  |  |
| Scalability |  |  |
| Security |  |  |
| Availability and peak usage |  |  |
| Resilience and recovery |  |  |
| Disaster recovery |  |  |

***Usability***

***If there are any specific usability drivers for the system describe the required design interventions here. The need for minimal user training, use by the visually impaired, operation by inexpert or ‘computer illiterate’ users etc. are example drivers for specific usability requirements***

***Performance***

***What are the expected user response times for real time systems, run time constraints for batch processes (e.g. must complete between 1.00am and 3.00am) etc. and how does the architecture support these requirements especially at peak usage times?***

***Capacity***

***What are the likely number of simultaneous users (by channel where appropriate) and how much data is likely to be flowing between hardware components. How does the architecture deal with this especially at peak usage times?***

***Scalability***

***Over time, it may be the case that numbers of users, data volumes etc. will increase. There are one or two ways to deal with this. The first is to build the solution to exceed current requirements and meet possible future requirements. The alternative is to create a smaller system now that can be scaled up to deal with increased capacity and performance demands. If a scalable solution is most sensible, describe how the architecture will accommodate the increases as and when required.***

***Security***

***Describe any security requirements and how these are accommodated by the architecture.***

***Availability and peak usage***

***Describe normal hours of operation including anticipated peaks and troughs in system usage over explicit periods of a day, a week, a month or a year, e.g. peak usage 8.00 to 9.00am, 12.00 to 2.00pm or Sunday night or last working day of the month or tax year-end. If appropriate, describe any appropriate provision within the architecture to deal with peak usage***

***Resilience and recovery***

***Based around the criticality of availability (i.e. how long the business can be without the system without suffering loss or damage) describe how the architecture meets this need. Describe how recovery after failure will be handled and how this is accommodated by the architecture.***

***Disaster recovery***

***Describe how the architecture provides for disaster recovery, eg does the solution require cross-site mirroring of data? If so, does the mirror copy need to be up to the minute, hour, 4 hours etc. or is last night’s off-site backup sufficient in a disaster recovery scenario?***

## 1.5 Maintainability considerations

***The system life expectancy and maintenance strategy section of the business area definition considers the options for how the system will be constructed. Describe how the architecture deals with these considerations, for example how a modular architecture will allow modules may be removed or replaced and how new modules may be added.***

***Note: The impact of the chosen approach must be fully considered and costed and critical decisions made around short term or long term deviation from the architecture described above. These should be considered when creating the delivery plan.***

Maintainability considerations…

# 2. Technical environments

## 2.1 Development platform

***Identify the software tools that will be used during development beyond the standard desktop environment (software tools include compilers, modelling tools, configuration management tools, testing tools, etc.).***

***Define the different environments that will be used by or created for the project eg development, testing, user testing environments.***

***Note: The description of the development platform should provide the project with sufficient information to identify:***

* ***the technical skills needed by the developers.***
* ***purchases needed to be made for development activity to take place.***

Development platform…

## 2.2 Target platform

***Identify the hardware and software that will need to be in place for the system to be delivered?***

***Identify the tools needed for staff performing support and maintenance activities?***

***Note: The description of the target platform should enable planning later within the project for migration and cutover.***

Target platform…