

A Primer on OBASHI Framework

OBASHI - A methodology for mapping and developing how IT systems relate to organizational operations (OBASHI stands for Ownership, Business Processes, Applications, Systems, Hardware, and Infrastructure).

The OBASHI methodology provides a framework and method for capturing, illustrating and modeling the relationships, dependencies and data flows between business and Information technology (IT) assets and resources in a business context.

It is a formal and structured way of communicating the logical and physical relationships and dependencies between IT assets and resources (Ownership, Business Processes, Applications, Systems, [Hardware](#), and Infrastructure) to define the business services of a modern enterprise.

The name "OBASHI" is a licensed trademark of Stroma Software Ltd.

Core Principle

OBASHI is based around a core principle that IT exists for one reason, namely, to manage the flow of data between business assets.

Business resources (which include people) and IT assets are either providers of data, consumers of data or provide the conduit through which the data can flow.

The role of IT is to support process and optimize the flow of data to maximize business/organizational performance.

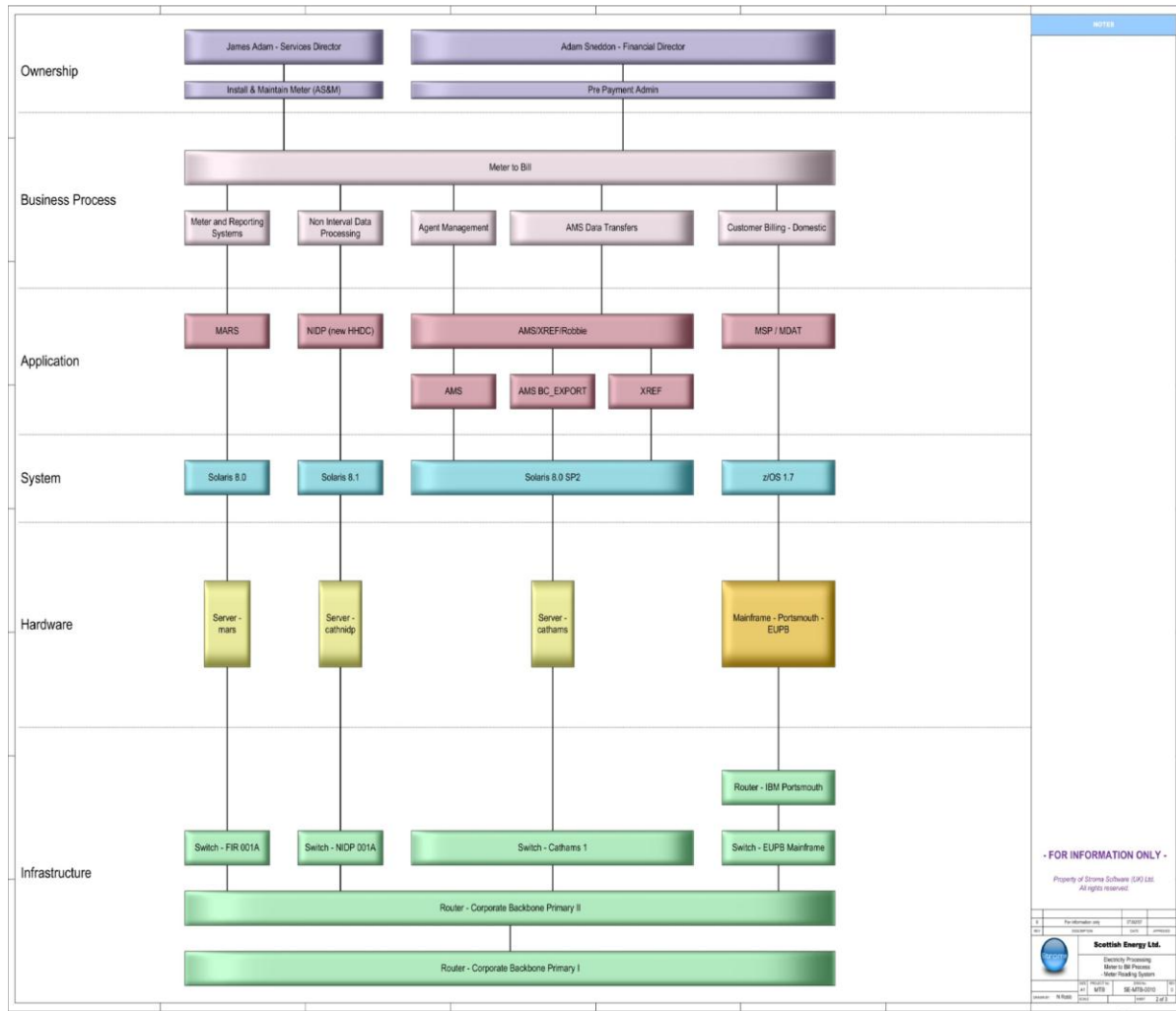
The OBASHI Framework

These Layers provide the framework for organizing the elements that represent individual Business or IT assets / resources. The six layers are:

- [O](#)wnership
- [B](#)usiness Process
- [A](#)pplication
- [S](#)ystem
- [H](#)ardware
- [I](#)nfrastructure

They are collectively known as OBASHI

A Primer on OBASHI Framework



Ownership Layer

The Ownership Layer contains elements representing the person(s) or group(s) that ‘owns’, or is responsible for, business processes portrayed in the Business Layer. Ownership elements can be positioned beneath other ownership elements to create a hierarchy of owners. Example owners could be: "Accountancy", "Planning Manager", "Logistics", "New York", "Purchasing Officer" and "Environmental Health".

Business Layer

The Business Layer contains elements representing the business processes or functions that are being used by the ‘Owner(s)’. These elements are positioned under their appropriate ‘Owner’. Examples could be: "Monthly Balance", "Sales Transactions", "Tank Stock Management", "Production Data" and "Capture Budgeting".

A Primer on OBASHI Framework

Application Layer

The Application Layer contains elements representing software applications. These are positioned beneath the business processes that utilise them. Examples could include: "Excel", "Oracle", "Sage", "SAP" and "PeopleSoft".

System Layer

The System Layer contains elements representing the operating systems on which the applications run. These elements are positioned beneath the appropriate applications. Examples could be: "Windows XP", "Unix", "Solaris", "Linux" and "Vista".

Hardware Layer

The Hardware Layer contains elements representing the computer hardware on which the operating systems run. These elements are positioned beneath the appropriate operating systems. Examples could be: "Workstations", "Servers", "Laptops", "Tablet PCs", and "Mainframes".

Infrastructure Layer

The Infrastructure Layer contains elements representing the network infrastructure into which the hardware is connected. Infrastructure elements can be positioned beneath other infrastructure elements to create a hierarchy that supports the business. Examples could be: "Switches", "Routers", "Multiplexers", "Bridges" and "Hubs".

A Primer on OBASHI Framework

Rules

The OBASHI Framework comprises the following rules which govern the implicit and explicit relationships between elements.

An element placed beneath or above another element has an implicit relationship with that element.

All elements within the same layer have an implicit relationship to each other.

Connected elements have an explicit relationship to each other, with the following rules governing connectivity:

A connection is a bi-directional relationship

An Infrastructure element may be connected to one or more Infrastructure or Hardware elements.

A Hardware element may be connected to one or more Infrastructure or System elements.

A System element may be connected to one or more Hardware or Application elements.

An Application element may be connected to one or more System, Application or Business elements.

A Business element may be connected to one or more Application, Business or Owner elements.

An Owner element may be connected to one or more Business or Owner elements.

A dependency is a uni-directional relationship i.e. element X may be dependent on element Y, but element Y might not be dependent on element X.

An element may have one or more instances within a layer.

An element can exist on more than one OBASHI diagram.

A dataflow comprises two or more connected elements.

A dataflow can contain one or more dataflows, enabling a hierarchy of dataflows.

A dataflow may span multiple Business and IT diagrams.