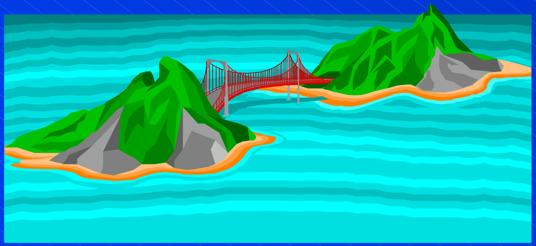
Islands of OOA



Jeff Froggart

froggajn@behp72.marconicomms.com

Shlaer-Mellor User Group Conference

September 1998

MARCONI COMMUNICATIONS

Contents

MARCONI COMMUNICATIONS

Legacy systems and Shlear-Mellor OOA

The Shlear-Mellor method is often perceived to be incompatible with legacy software systems

Typical perceptions :-

- Shlear-Mellor is a "pure" technique
- "Thou shalt not pollute a Shlear-Mellor model"
- Needs a green field development
- Shlear-Mellor systems are slow
- "It's not REAL OOA!"

MARCONI
COMMUNICATIONS

Background

- These are my own views!
- I'm not very up to date with latest OOA
- Our application of OOA has been cautious and practically orientated
- Architecture and Framework.
  - Architecture - used to map OOA to implementation
  - Framework - Defines a set of interfaces and development process for OOA

MARCONI
COMMUNICATIONS

The Interworking problem

The EM-OS project circa 1994

- Large amount of existing (working!) software
- Emerging defined interfaces and system design
- Natural split off between EMs (Element Management) and NCL (Network Level Control)
- Large amount of common functionality
- Desire to partition system
- Desire to make use of common software elements
- Desire to make use of newer implementation technologies

MARCONI COMMUNICATIONS

Initial solution

- Parallel development of a green field Shlear-Mellor replacement EM-OS
- Problems
    - Cost!
    - Low buy in from existing development teams
    - Difficulty in "proving" the techniques
    - Skill base problems

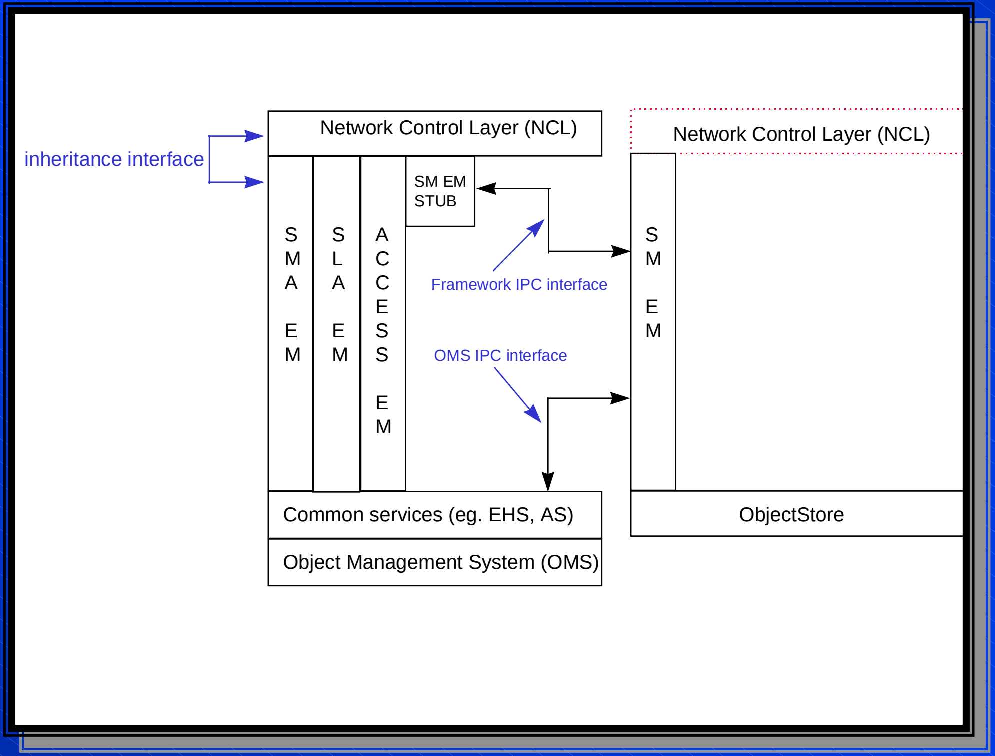
MARCONI COMMUNICATIONS

Current "Framework" solution

- Provides an OOA aware framework as an extension of the OOA architecture
- Provides the possibility of "OOD'ed" domains as an alternative to OOA domains
- Connects these "Islands of OOA" into the legacy system via "Interface domains"
- Interface domains straddle the OOA and legacy system worlds to bridge them

MARCONI COMMUNICATIONS

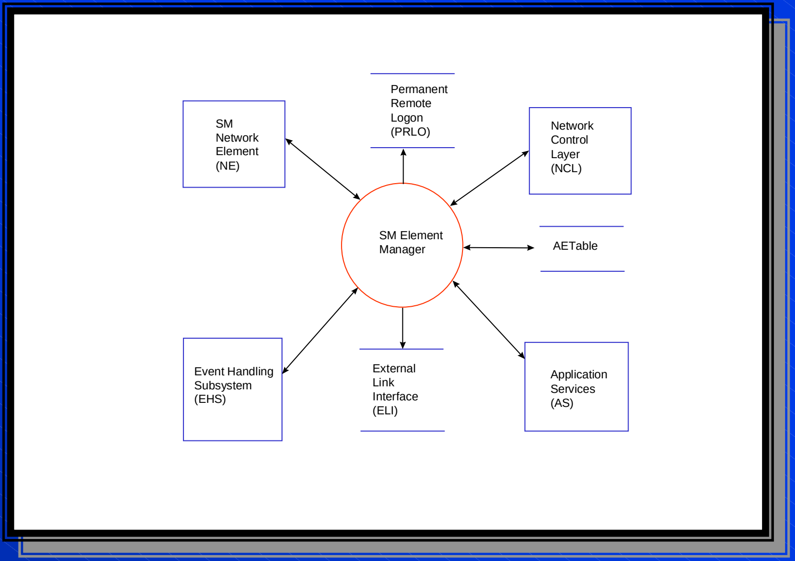
EM-OS block diagram



inheritance interface
Network Control Layer (NCL)
SMA EM
SLA EM
ACCESS EM
SM EM STUB
Framework IPC interface
OMS IPC interface
Network Control Layer (NCL)
SM EM
Common services (eg. EHS, AS)
Object Management System (OMS)
ObjectStore

EM context diagram

The external interfaces with which the SM EM interacts are shown below. This diagram is typical for all the element managers within the EM-OS system


SM Network Element (NE)
Permanent Remote Logon (PRLO)
Network Control Layer (NCL)
SM Element Manager
AETable
Event Handling Subsystem (EHS)
External Link Interface (ELI)
Application Services (AS)

MARCONI COMMUNICATIONS

Framework concepts

- Database/process mapping
- Signals and Event/Response pairs
- Counterparting
- Sync Services
- Domain Interface (DIF) Objects
- Domain Stubs
- Interface Domains

MARCONI
COMMUNICATIONS

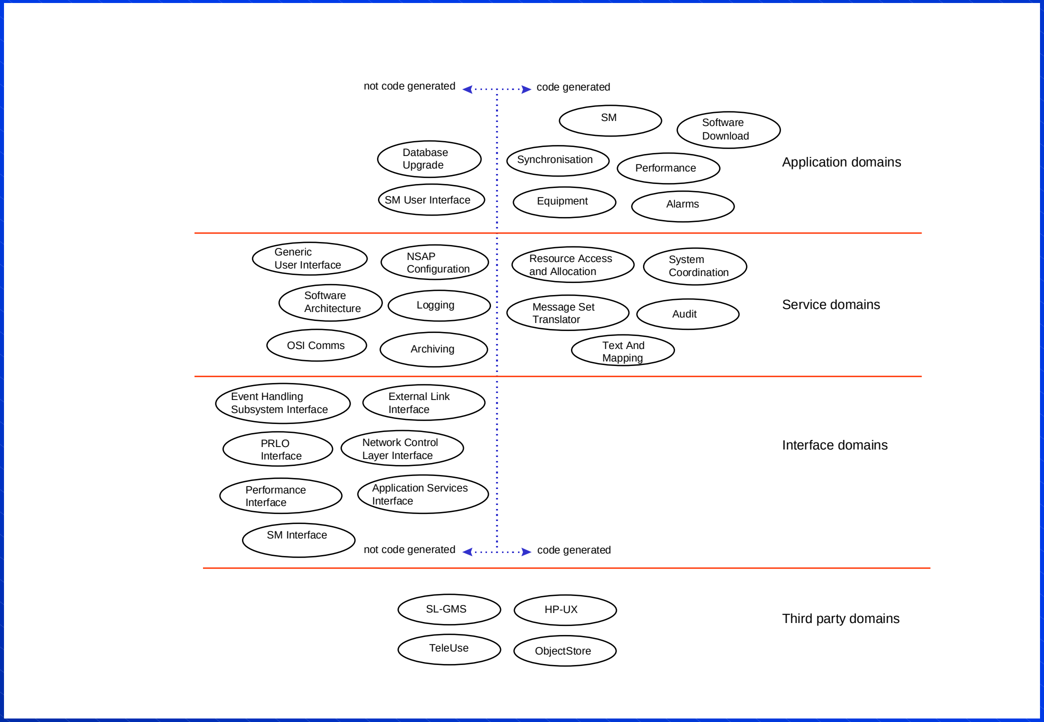
Framework domain variants

- "Full" Shlear-Mellor OOA domains using I-OOA and ASL
- Skeleton based OOA domains, implemented with C++ actions
- Full C++ OOD domains
- User interface OOD domains
- Interface OOD domains

Domain Charting

- Bound the domain chart to the legacy software's requirements
- By interactions between interface domains and the legacy interfaces
- By defining any implementation domains from the legacy software - these can then have either
    - a direct bridge mapping
    - an associated interface domain

Example domain chart



MARCONI COMMUNICATIONS

Bridging to non-OOA software

- Interface domains provide the services required to support the bridging between the OOA and legacy software.
- They map the OOA concepts of the Framework Architecture to the legacy's IPC and C++ method call interfaces.

MARCONI
COMMUNICATIONS

Counterparting to non-OOA software

- Object instances must be mapped from the legacy software's naming method (text based) to the Framework Architecture's numerical counterpart ids.
- The naming service is one of the functions provided by an interface domain.
- Creation threads need reporting and may need synchronising
- Uniqueness of counterpart ids must be maintained
- Identifying attributes in the legacy system must be defined

MARCONI COMMUNICATIONS

Mapping Bridging modes

- Often need to map synchronous blocking calls in legacy system to asynchronous event/response pairs (non-blocking) in the Framework domains
- Signals from Framework domains are often converted to closed calls in the legacy system. The interface domain collecting the returned information for the naming server

MARCONI COMMUNICATIONS

Handling failures and errors

- Failure modes may be different between the legacy software and the OOA architecture (Framework)
- It may be necessary to convert error status values to Framework error events
- It may be necessary to convert error signals or traps to reply statuses on event responses in the Framework domain

MARCONI COMMUNICATIONS

Conclusions

- Shlaer-Mellor OOA can be incorporated into existing legacy software systems. The EM-OS OOA "Framework" is one such solution.
- The legacy software can make use of OOA through "Islands of OOA" sitting in a "Framework" sea.
- A more flexible approach to Shlaer-Mellor domains can allow OOA to "see" legacy software through "Interface domains".

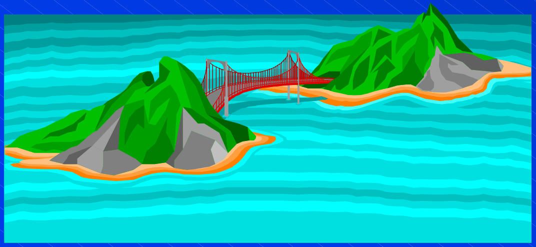
MARCONI COMMUNICATIONS

Conclusions

- If the OOA to implementation translation would lead to unacceptable performance then the use of OOD within an OOA Framework can be used as an alternative to hand coding or providing multiple translations.
- Investment in architectural software can be exploited not only in Shlaer-Mellor OOA based software but can also act as an infrastructure for OOD developed "Domains".

MARCONI COMMUNICATIONS

Islands of OOA



Jeff Froggart

froggajn@behp72.marconicomms.com

Shlaer-Mellor User Group Conference

September 1998

MARCONI COMMUNICATIONS