

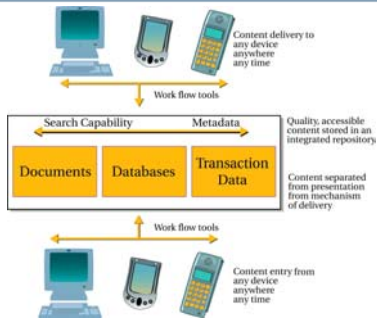
Utility Technology Solutions

Design & Architecture Principles for Performance IT



Empowering Business for Today and Tomorrow

Eliminate the Ph.D.



Empowering Business for Today and Tomorrow

Designing the Knowledge Worker Assembly Line

- IT is the assembly line for knowledge workers
- Design what we make then design the assembly line
- Cost, quality, timeframe, how many people
- Assure quality of hand-off
- Standard monitoring, assurance & reporting



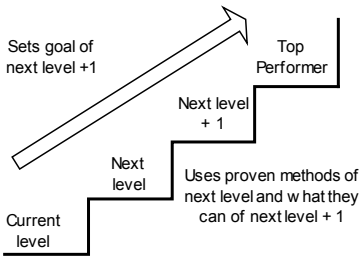
Empowering Business for Today and Tomorrow

Market Forces Drive Design

- One size does **NOT** fit all
- Mature market is a highly standardised offering
- Evolving market rapid design changes
- Know your product offering
- Differs between & within an organisation



Market Forces in Solution Design



- Model for IT optimisation
- Migration path
- Proven, simple, effective
- Business optimisation model



Rigorous Design & Engineering

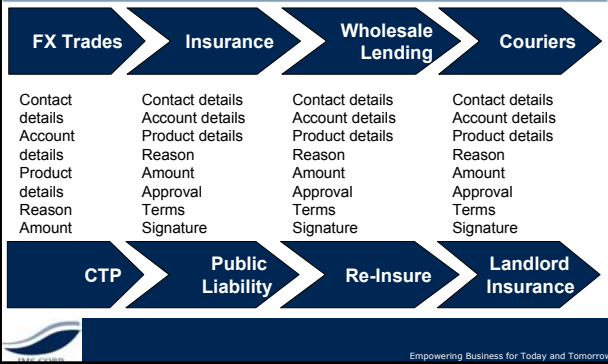


Proven Design Principles

- IT is the assembly line for knowledge workers
- Market forces drive solution design
- Define the product
- Design is an iterative process
- Standardised construction and deployment
- Agile methods – another excuse not to think?



Common Process - Refunds



Elements of the Solution

- Consolidated Repository
- Relationships between information
- Information with work-flow
- Integrated search
- Presentation
- Messaging
- Security



Designing Solutions for the Information Age

Common Solutions Extending From the Interface to Back Office



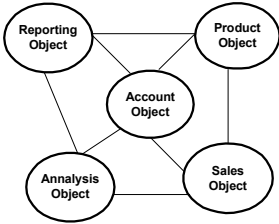


Standard Iconic Interface

- Standard objects
- Traps work-flow
- Easy to use
- Business defined
- Auto change for task
- Eliminate application specific knowledge
- Business sees same interface as customer
- Search
- Context sensitive help
- Multi-lingual
- Customisable
- Messaging
- Functionality integration



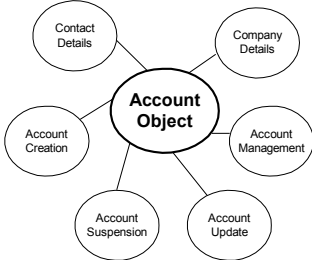
Selling Activity Objects



- Define standard objects for all activities
- Activity objects extend from interface
- Objects define work-flow & information
- Images, security, business logic & processing rules



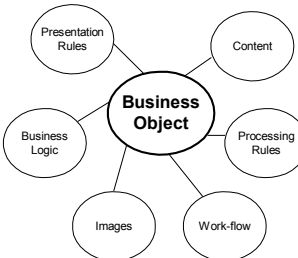
Object Hierarchy



- A hierarchy of activities - recipes
- Objects within objects becoming more specific
- Inclusive & authoritative
- Easily defined
- Easily documented



Object Structure



- Object contain information, relationships, content, business rules & logic
- Define the objects - define the software
- Standard structure
- Dynamic interface
- Separation of elements
- Recyclable, standardised, dynamic, scalable



Underlying Databases

- Information databases
 - Information relationship databases
 - Rule processing databases
 - Activity object databases
 - Exceptions databases
 - Messaging databases
-
- Accessed from virtual consolidated repositories



Activity Approach

- | | |
|---|---|
| <ul style="list-style-type: none">• Elements <ul style="list-style-type: none">• Set of standard objects• Rendered in a dynamic framework• Separation of rules, logic, data etc.• Virtual, consolidated contextual databases• Configurable standard interface | <ul style="list-style-type: none">• Advantages <ul style="list-style-type: none">• Business driven• Comprehensive• Intuitive• Across device support• Eliminates information access issues• Eliminates application issues• Customisable & configurable• Multi-lingual |
|---|---|



Utility Infrastructure

Moving From Solutions to Utilities



Industrial Strength Utilities

- Redundancy
- Failover
- Load bearing capacity
- Kept simple
- Not the latest everything
- Standardised & modularised



Utility Characteristics

- Accommodate change
- Assume failure
- Standardisation
- Best of breed
- Emergent behaviour
- Scalability through automation
- Bring together simple solutions



Unified Approach to Utility Information Technology

Proven Principles for Design, Engineering & Architecture



Questions