Structured Information Management for Large Defence Projects

Engineering Corporate Knowledge

William P. Hall, PhD
Integrated Logistic Support
ANZAC Ship Project
Williamstown, Vic.
bill.hall@tenix.com
Overview: Assembling corporate information into products

◆ Documentation management issues for large defence/industry projects
◆ What is structured information management and why does the world need it?
◆ Tenix’s implementation project
  • What we got
    – 80% reduction in number of documents managed
    – 98% reduction in documents delivered
    – further 50-70% reduction in text down the line
  • How we got it
Background: ANZAC Ship Project

- Project to design, build & support 10 frigates
  - 1989 $A 5 BN
- Tenix a privately owned company
- Fixed price contract including support
- I have filled document production and content management roles from the time the contract was signed covering the entire project cycle
Documentation cycle for a large project (prime contractor’s view)

- Short List
- Award
- Subcontract
- Deliverable Documents
- Design Study for the Next New Project
- Requirements & specifications
- Supplier source data
- Review/edit/sign-off & deliver
- Negotiate & Amend
- Review/ed it & Sign Off!
- 20 - 30 year lifetime!
Most corporate knowledge is recorded in documents not data

Product goals
- Correct and up-to-date
- Consistent and legible
- Available to who needs it, when needed

Process goals
- Faster (minimise cycle time)
- Better (improve/assure quality)
- Cheaper (reduce labour)

Automation can help
Key vision: Documentation transforms data and text into knowledge and power

TRANSFORMATIONS THAT ADD VALUE TO TEXT

"INFO WARFARE"

After a diagram by Ian Coombe
(ian.coombe@knowl.com) from the Australian Army Information Management Manual (Ver 2.0)
Key vision (2)

- Unstructured data or text has no meaning to a computer
- Document structure encodes context and semantics
- Content management helps with awareness and speeds decision and action = knowledge management
Structure text, don’t format it

- Focus on document structure and content, not paper format
- Author documents with a defined and controlled structural logic.
  - Unique labels
  - Consistent rules
  - Authors create text and structure, not formats
  - Computer systems, not authors, provide formats
  - Computer systems can process structured documents to add value like databases add value to tabular data.
Tools to engineer documentary knowledge

- Input material: knowledge expressed in text
- Eng. standards: SGML (ISO 8879 - 1986), XML, etc.
  - Requires development of document designs (DTDs)
  - Output format specifications
- Text processing (machine) tools produce/maintain structured text according to engineering standard
  - Parsers
  - Editors
  - Formatters/publishers
  - Processing languages
  - Content managers/databases
Tenix’s history with structured information management
Trial solutions for maintenance documents

  - Merge tables provided “structure” from the outset
  - Produced more than 20 different outputs from “single source”
  - Delivered data files and electronic documents to Client’s Oracle-based ANZAC Maintenance + Planning System (AMPS)
  - Validated data used for AMPS relational links automatically
  - Information was structured but content not controlled
  - Versioning, release, applicability, effectivity all manual
  - Data delivery capability at risk from obsolete host application
Tenix Defence Systems

Trial solutions

- Tenix explored several alternatives to replace WordPerfect
  - SGML for equipment overhaul specifications (1994→)
  - Relational database solution for amphibious landing ship maintenance routines (1996-1997)
  - MS-based reworks of WordPerfect technology
- They weren’t good enough
Provide structured authoring and content management

- Implement concept of structured documents
  - Semantic markup (SGML, XML)
  - Hyperlinks (live links that turn information into knowledge) - HTML, SGML, XML
- Manage documents (container status & versioning)
- Manage content (the contained knowledge elements)

ANZAC maintenance routines a good prototype for many corporate needs

Help transform: Data → Knowledge → Power
Tenix funded a year-long R&D program to evaluate structured authoring technologies

1. Develop preliminary requirements & survey technology
   - SAIC/SDSC evaluation - http://www.sdsc.edu/DOCT/Publications/b4-1/b4-1.html
   - Allette Systems provided advice & assistance in developing internal business case to proceed

2. Develop detailed SOR and draft contract
   - Initially preferred suppliers helped analyse requirements
   - Ensured we expressed our requirements in suppliers language

3. Two rounds of bidding against RFQ (not RFT)
   - CSIRO Mathematical & Information Sciences’ Text Information Management reviewed process and technical evaluation
What we decided we needed:

- Hold data in non-proprietary format (SGML/XML)
- Validate critical data against master sources
- Manage applicability to configuration items
- Link document effectivity to engineering change orders
- Maintain client-specific languages
- Register and link to source documents
- Link deliverable text elements to source documents
- Manage workflow processes
- Manage and reuse content objects (e.g., graphics)
- Manage and reuse document components (e.g., text)
- Data conversion from WordPerfect to SGML
- Produce required deliverables
  (unique requirements)
R&D review of document mgmt applications

◆ Architectural considerations:
  • Repository (relational vs object vs structured)
  • Extensibility (DIY?)
  • Supportability (DIY?)

◆ Other considerations:
  • RISK minimisation
  • Licensing and implementation cost
  • Support and maintenance cost
Final short list:

- Short list choice between two generic SGML content management systems
  - low risk 2nd generation US vs developmental 4th generation local
  - dependence on overseas IP vs local support and IP
  - overseas cost components vs 100% $A
What did we get?
Our choice: SIM

◆ RMIT’s Structured Information Manager

  • Academic excellence + interesting client list → world-wide state of the art repository technology
  • A very credible implementation team
  • Acceptable risk mitigation
    – Fixed price (but no ownership of bespoke IP)
    – Two stage implementation (80:20 rule)
What did we get?

- XML-based content repository server
- Web and access security admin server
- Integral Application Construction Environment (ACE)
- Tenix’s document & content management extensions
- No third-party software anywhere!
Delivered class documents 20/10/2000

- Total doc management requirement as at Ship 5 delivery:
  - 10,000 routines → 1,800 = > 80% reduction in docs

- Ship 5 delivery requirement:
  - 40 new routines not 2000! = 98% reduction in delivery

- And we weren’t too bad to start with
How did we do it?

How was the reduction achieved?

- Item applicability to configuration items
- Side-by-side edit navy specific language (RAN, RNZN)
- Effectivity link to specific Engineering Change
Other release 1 benefits

◆ **Low cost deployment**
  - Own PC with Web or intranet access
  - Default web browser (only tool needed to administer SIM or review & annotate documents)
  - SGML/XML compliant editor to draft/edit procedure text
  - No IS support required

◆ **Data validation on Check In/Check Out**

◆ **Source data management**
  - Source registry
  - Source repository
  - 2-way links with SGML comments
More release 1 benefits

- **Workflow management system**
  - All functions performed over intranet web
  - With slight mod to SIM implementation, participants could work anywhere in the world (World Wide Web)

- **Minimal author training (no prior experience)**
  - Workflow role(s)
  - Document structure rules
  - Data/text entry
  - Authors see but have no control over print formats

One day of formal training
One week to full productivity
SIM Release 2 - early in the new year

- Component reuse
  - Auto detection of similar elements
  - Inheritance, detailed tracking of changes at element level
- 50% - 70% more reduction in volume of text to author/manage
- Ensures consistency of text usage
SIM Release 2 - early in the new year (2)

◆ Document and component versioning
  • SIM 1 retains doc versions in repository
  • SIM 2 will control versions down to individual elements
  • Secure/traceable version/release control
  • Auditable change management

◆ Annotated links, changes & elements

◆ Additional toolkit
Project success factors
Strategies that worked (1)

- End-users (doco people) managed project
- Used appropriate outside consultants (including suppliers) to help understanding & check work
- If you don’t know more about technology than suppliers do RFQ not RFT
- Specify what you need, not how to achieve it
- Check supplier financials and market position
- If developer is overseas use reputable local integrator as supplier, not the developer
Strategies that worked (2)

- Negotiate from a commercially realistic draft contract
  - The biggest risk may be to not start the project
- Manage risk above all else
  - Negotiate fixed price
  - Solve potential show-stoppers first
  - Prove major concepts early
  - Know your priorities and apply the 80:20 rule
- Phase work so each phase scores a win