Essential MoReq2

(or towards MoReq2.5)

Refactoring the standard for greater adoption

Jon Garde

Characteristics of MoReq2

- Large
  - Bigger than any other comparable standard
  - MoReq1, DoD 5015.2, TNA 2002, ICA, etc.
- Plus
  - Metadata Model
  - XML Schema
  - Testing and certification
- Complex
- Exhaustive
- Superset
- Unfinished

Suitable For

- General purpose systems (e.g. office, clerical, government, etc.)
- Traditional applications (e.g. word processing, e-mail, etc.)
- Large vendors who offer an entire, integrated EDRM suite

  = between 5 and 15 software vendors' products?

Unsuitable For

- Web 2.0 applications (e.g. Wikis, Blogs, instant messaging, etc.)
- Integrations* with specialised systems (e.g. CRM, ERP, etc.)
- Products aimed at vertical markets (e.g. financial services)
- Single purpose systems (e.g. e-mail archiving)
- Niche/boutique vendors who offer only part of a solution

  = between 50 and 500 software vendors' products?□
- In-house application development

**note: API not mandatory*

Model Requirements *versus* Certified Requirements

- “**MoReq**” comes from = “**Mo**del **Req**uirements for Electronic Records Management”
- The specification contains model requirements because organisations are promised a “model”, or template, that they can modify to suit their own records management needs
- For this reason, MoReq is always distributed in Word format (as well as PDF) so that it is more easily edited and adapted
  - Note: in practice this rarely (if ever) happens

BUT

- MoReq2 has a **standardised** testing and certification process

- Organisations are buying a solution that has **already** been tested and awarded a certification against the unmodified MoReq standard

- How can **organisations** meaningfully append, extend, modify or remove requirements without invalidating the certification process?

- How can **vendors** re-implement / customise their products to suit individual consumers and still claim the end result to be certified against MoReq2?

We must either invent a new interpretation explaining why MoReq2 requirements are “model requirements” or we should drop the moniker: “MoReq”

Over-complexity

In software development
complexity = cost
so a more complex
standard means a more
expensive compliant system

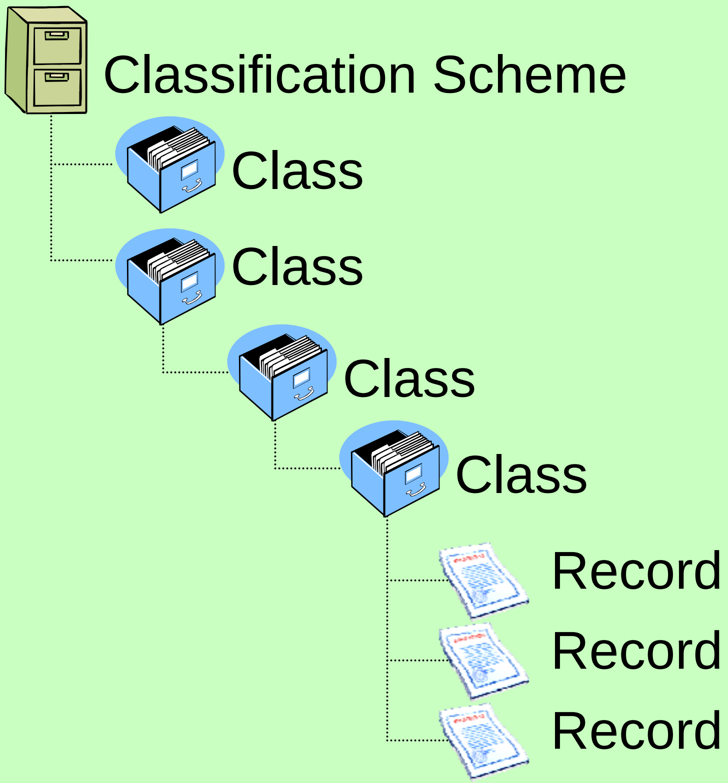
Who pays?

- Cost is passed on to the consumer
- Longer development cycles mean only large vendors build compliant systems
- Fewer compliant products are available to choose from
- Less competition between suppliers
- Stifling of innovation
- Increased evasion / lack of compliance

One example of over-complexity

How many file plan configurations **must** a MoReq2 compliant ERMS support (mandatory)?

Class
Record



Classification Scheme

Class

Class

Class

Class

Record

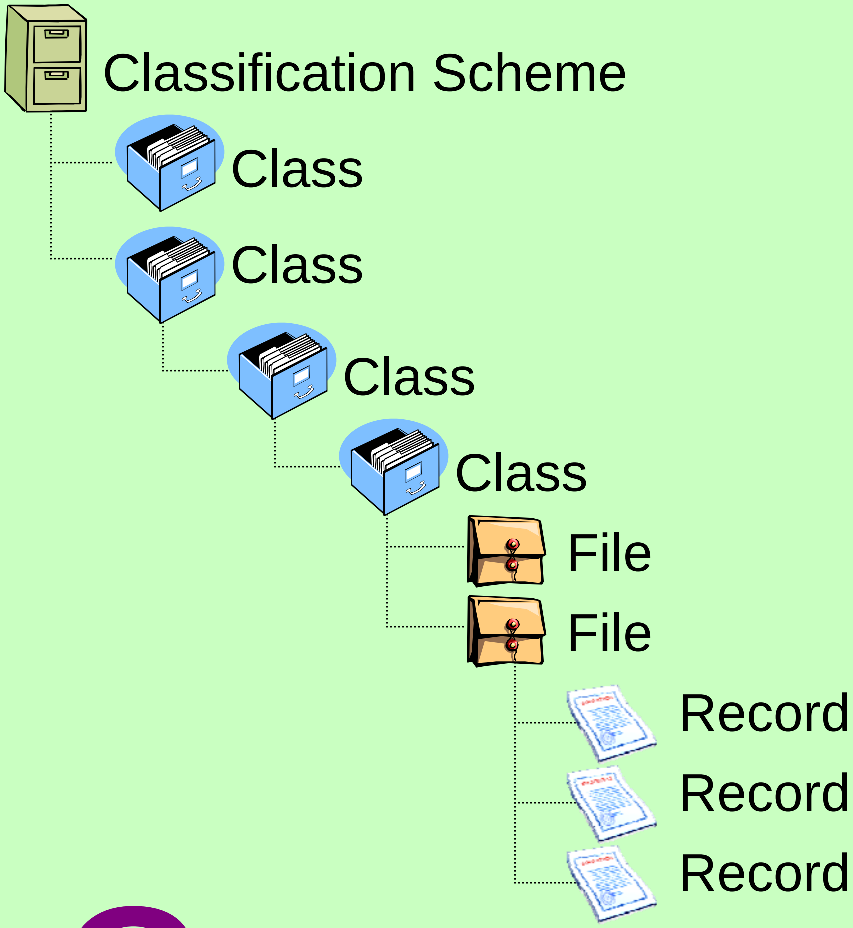
Record

Record

1

- *MoReq2 compliant software must support “unfiled” records in classes*

Class
File
Record



Classification Scheme

Class

Class

Class

Class

File

File

Record

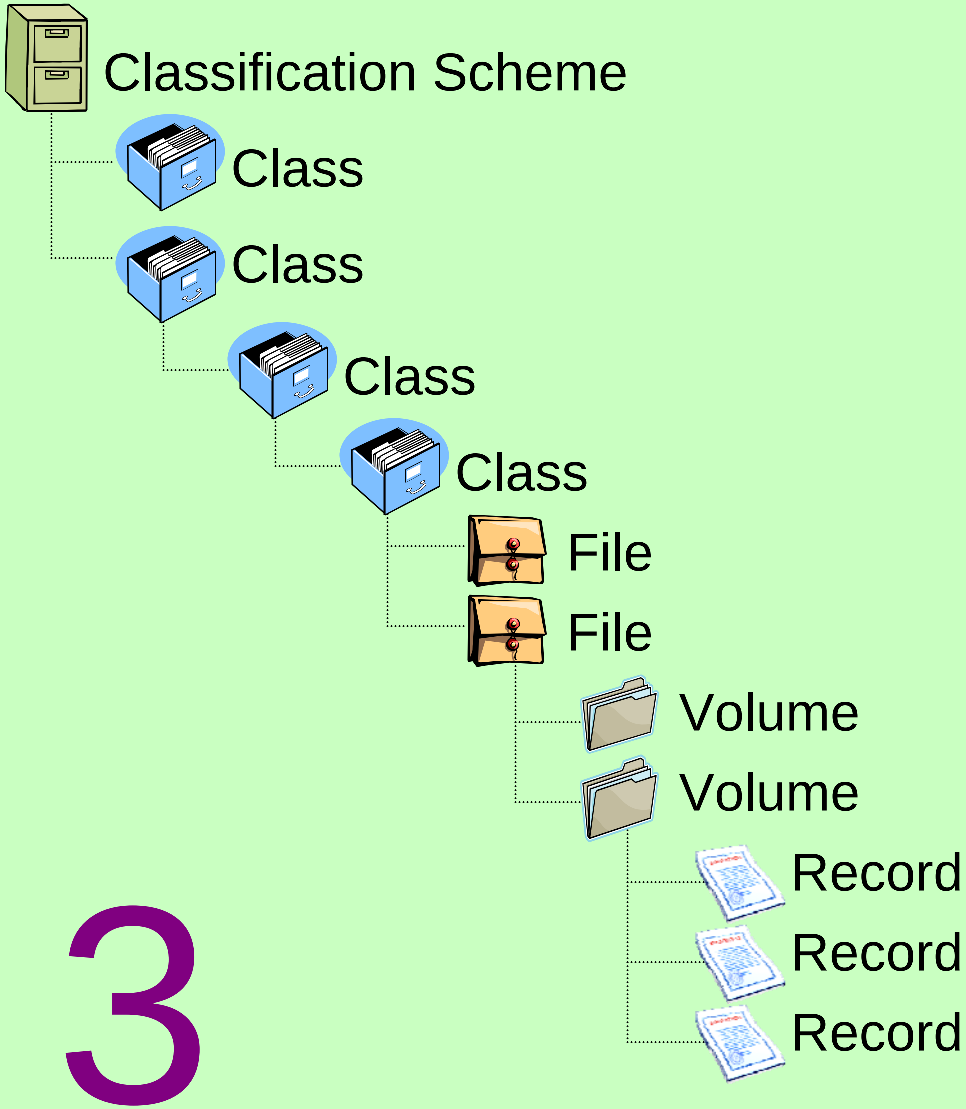
Record

Record

2

- *This is the simplest “traditional” file plan*

Class
File
Volume
Record



Classification Scheme

Class

Class

Class

Class

File

File

Volume

Volume

Record

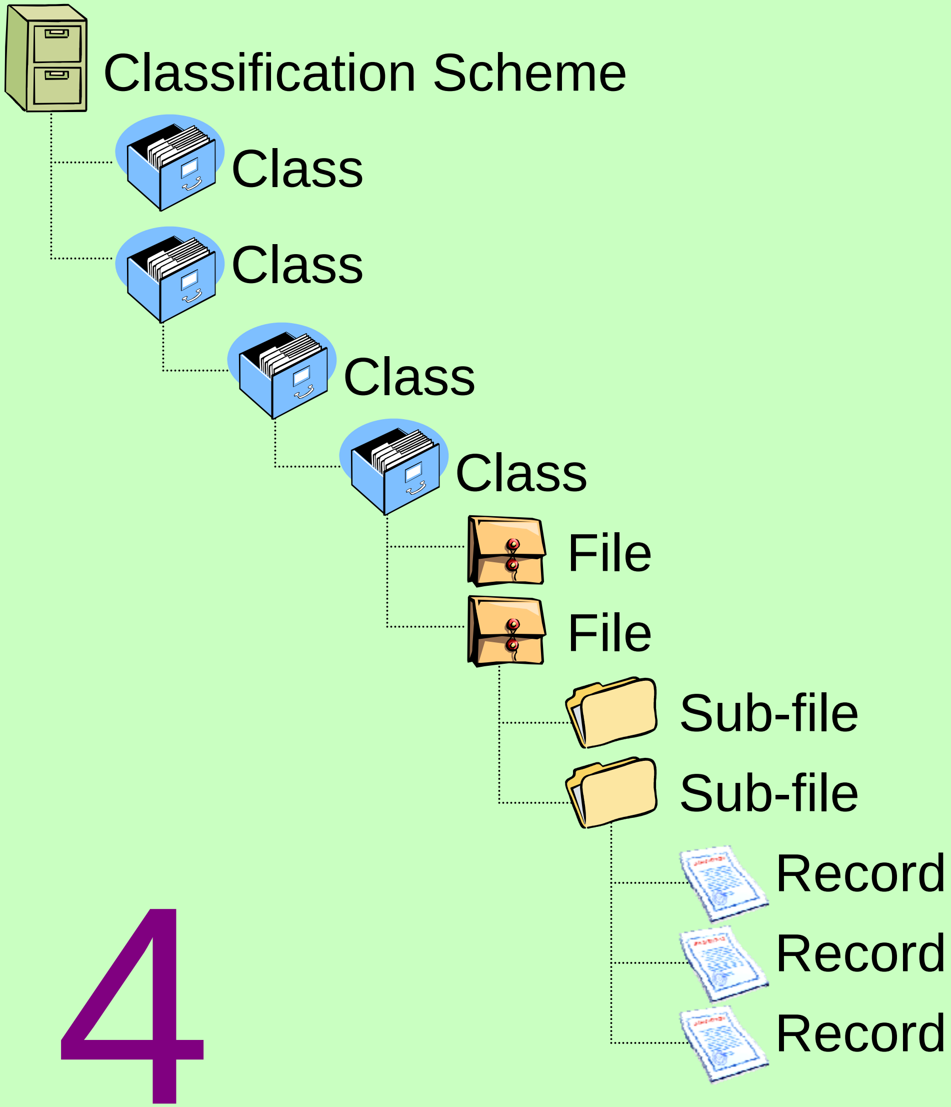
Record

Record

3

- *This is the only configuration supported by MoReq1 and TNA 2002*

Class
File
Sub-file
Record



Classification Scheme

Class

Class

Class

Class

File

File

Sub-file

Sub-file

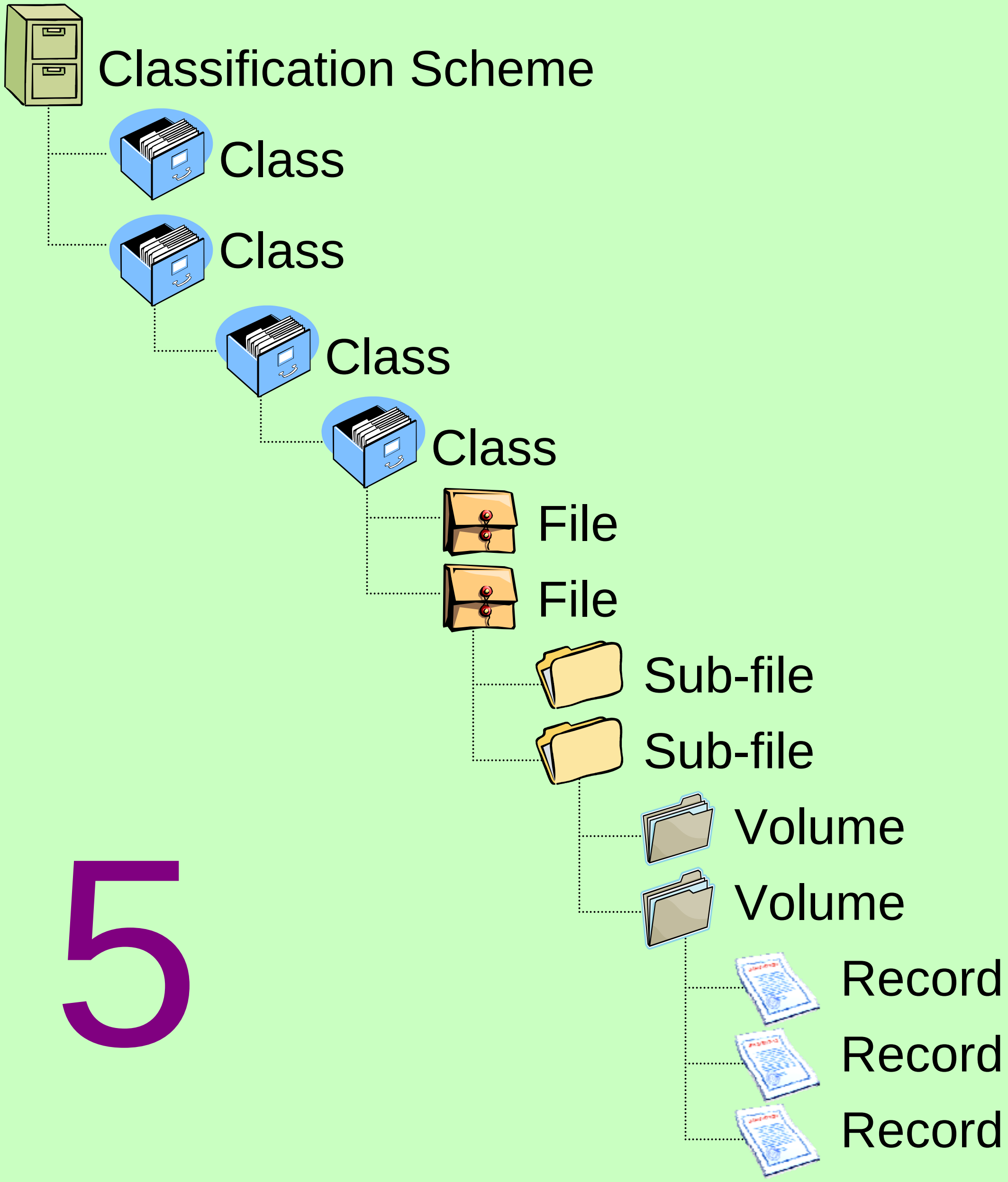
Record

Record

Record

4

- *Sub-files are new to MoReq2*



Classification Scheme

Class

Class

Class

Class

File

File

Sub-file

Sub-file

Volume

Volume

Record

Record

Record

Class

File

Sub-file

Volume

Record

5

- *MoReq2 allows this many levels*

- To comply with MoReq2 a product must support no less than five different file plan configurations
- None are optional
- By contrast, MoReq1 and TNA 2002 only describe one file plan configuration

Is it appropriate that ALL certified ERM systems MUST include support for ALL of these file plan configurations?

Refactoring MoReq2

Vision

- A MoReq2 based standard that is easier and faster to implement: resulting in a wider choice of inexpensive compliant solutions

- A less complex standard that loses none of the richness of the original

- A refactoring project that takes considerably less effort than writing MoReq3

Process

- One possible way to meet this vision is to make the standard more modular with a smaller mandatory core
- Identify non-essential functionality which results in overly complex requirements and remove it from the core
- Create additional optional modules containing this non-essential functionality that can be selected only by those consumers that need it

Results

- When this approach was applied to the published MoReq2 specification:
- The total number of requirements in the core was reduced by nearly a third
- While the number of optional modules doubled

There was no change to either the requirements themselves nor the total number and type of requirements present in MoReq2

New Modules

Multiple Classification Schemes

(1 requirement)

- The need for a compliant solution to support multiple classification schemes adds enormous complexity (and cost)
- *"Most organisations will mandate that a single classification scheme be used for the primary classification of all the files in the ERMS."* **MoReq2, v1.04, p28**

Sub-Files

(10 requirements)

- Not every organisation needs sub-files
- *"In some environments it is useful to divide files into sub-files... Sub-files are most often used in case processing environment…"*

MoReq2, v1.04, p16

Volumes

(13 requirements)

- Are volumes a hangover from physical records management? Are volumes mandatory for electronic environments?

- “*Typically, paper files are limited to 2cm in thickness, by the establishment of volumes.*” **MoReq2, v1.04, p31**

Unfiled Records

(4 requirements)

- MoReq2 advises against using unfiled records: *"This change has been introduced to reflect the requirements of high-volume case management systems. It is, however, not meant to remove the necessity for a hierarchical classification scheme, or for the existence of files. Inappropriate use of this feature will introduce the risk of later difficulties in managing records, and users of MoReq2 are advised to use this functionality only after careful analysis. Most users of MoReq2 are unlikely to require this functionality, and so MoReq2 includes the requirement that this functionality can be disabled."* **MoReq2, v1.04, p24**

Partial Recovery and Back-Up

(2 requirements)

- This requirement sounds good but is totally unfeasible for any technology based on an underlying relational database
- A relational DB cannot be partially backed-up and recovered and still retain Referential Integrity
- Since this includes most, if not all, ERMS implementations on the market today very few systems will ever become MoReq2 compliant

Vital Records

(5 requirements)

- Only some organisations keep vital records
- *“Vital records are the records that are considered absolutely essential to the organisation’s ability to carry out its business functions, in the short term, in the long term or both... This can be either mission-critical in terms of its ability to cope with emergency / disaster conditions or to protect its long-term financial and legal interests.”* **MoReq2, v1.04, p49**
- Many organisations with vital records will keep them in a separate ERMS

Interactive Document Capture

(17 requirements)

- Confuses an ERMS with an EDRMS
- EDRMS systems are only a sub-set of possible ERMS solutions
- ERMS solutions can “sit behind” other systems and have no interfaces visible to general users
  - e.g. MS Sharepoint Records Center API

E-Mail Management

(18 requirements)

- A specialised ERMS may never need to interact with an e-mail client
- An e-mail archiving system will capture records from the e-mail server not the client

Scanning and Imaging

(23 requirements)

- Do all ERMS applications capture scanned images?
- e.g. An electronic billing system?
- These ERM systems will never need to interface to scanning and imaging equipment

Displaying

(4 requirements)

- Does an ERMS need to know how to display the records it holds?
- The ERMS may rely on third-party applications to provide viewing, listening, or watching capabilities

Printing

(19 requirements)

- Does an ERMS need to be able to print the records it holds?

- *"Presentation is producing a representation on-screen ('displaying') or printing; it may also involve, as necessary, playing audio and/or video …"* **MoReq2, v1.04, p98**

Redaction

(9 requirements)

- Redaction is specific to a particular business need
- Do all ERMS contexts have this need?

Total = **125** requirements
in 12 modules

The New Core

Core Modules - 1

- Classification Scheme (54 requirements)
- Classes and Files (16)
- Access (24)
- Audit Trail (16)
- Full Backup and Recovery (5)
- Retention and Disposition Schedules (41)
- Review of Disposition Actions (8)
- Transfer, Export and Destruction (24)
- Record Capture (24)

Core Modules - 2

- Bulk Importing (8 requirements)
- Record Types (5)
- Classification Codes (10)
- System Identifiers (6)
- Search and Retrieval (33)
- General Administration (5)
- Reporting (34)
- Changing and Deleting Records (11)

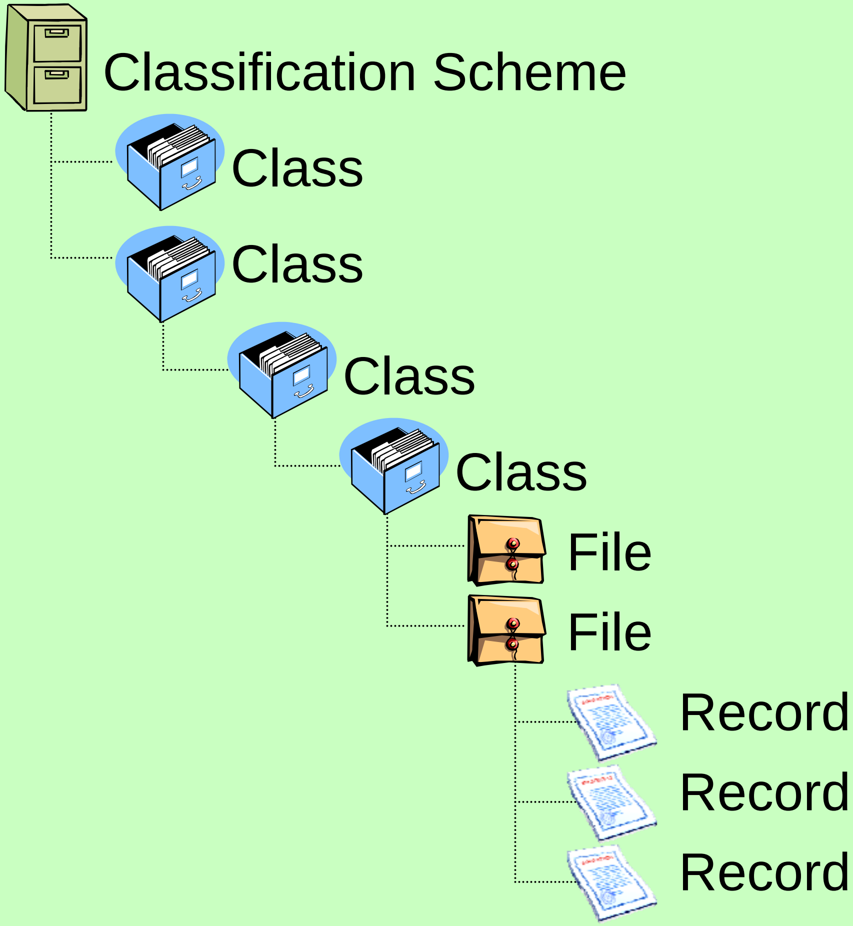
- **Total requirements 324** (previously 442)

An example of the reduction in complexity

Only **one** (compared to five different) file plan configuration(s) required to meet core requirements

Core (only)

Class
File
Record



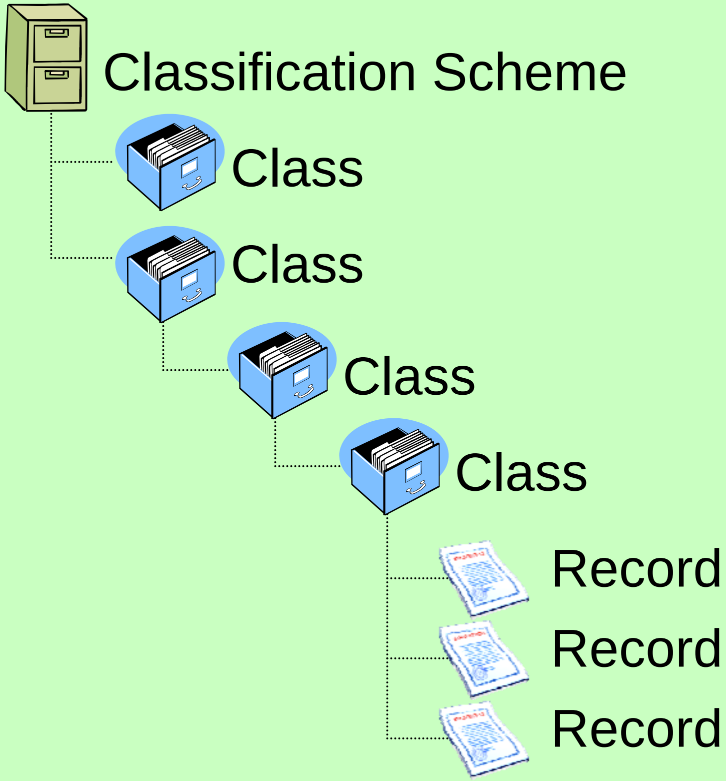
Classification Scheme
Class
Class
Class
Class
File
File
Record
Record
Record

(was 2)

- *This is the simplest “traditional” file plan*

Core + Unfiled Records module

Class
Record



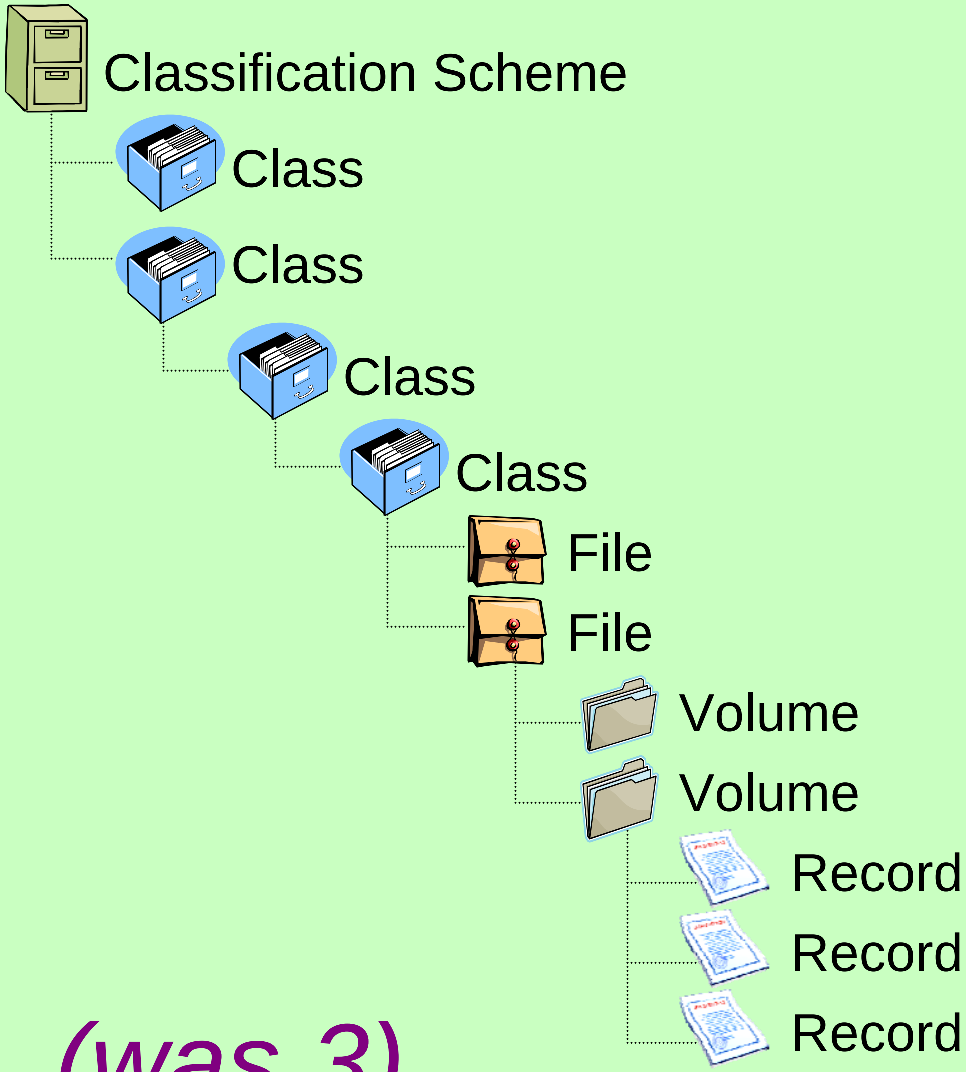
- *MoReq2 compliant software must support unfiled records in classes*

(was 1)

Core + Volumes module

(prerequisite for physical records module)

Class
File
Volume
Record



Classification Scheme

Class

Class

Class

Class

File

File

Volume

Volume

Record

Record

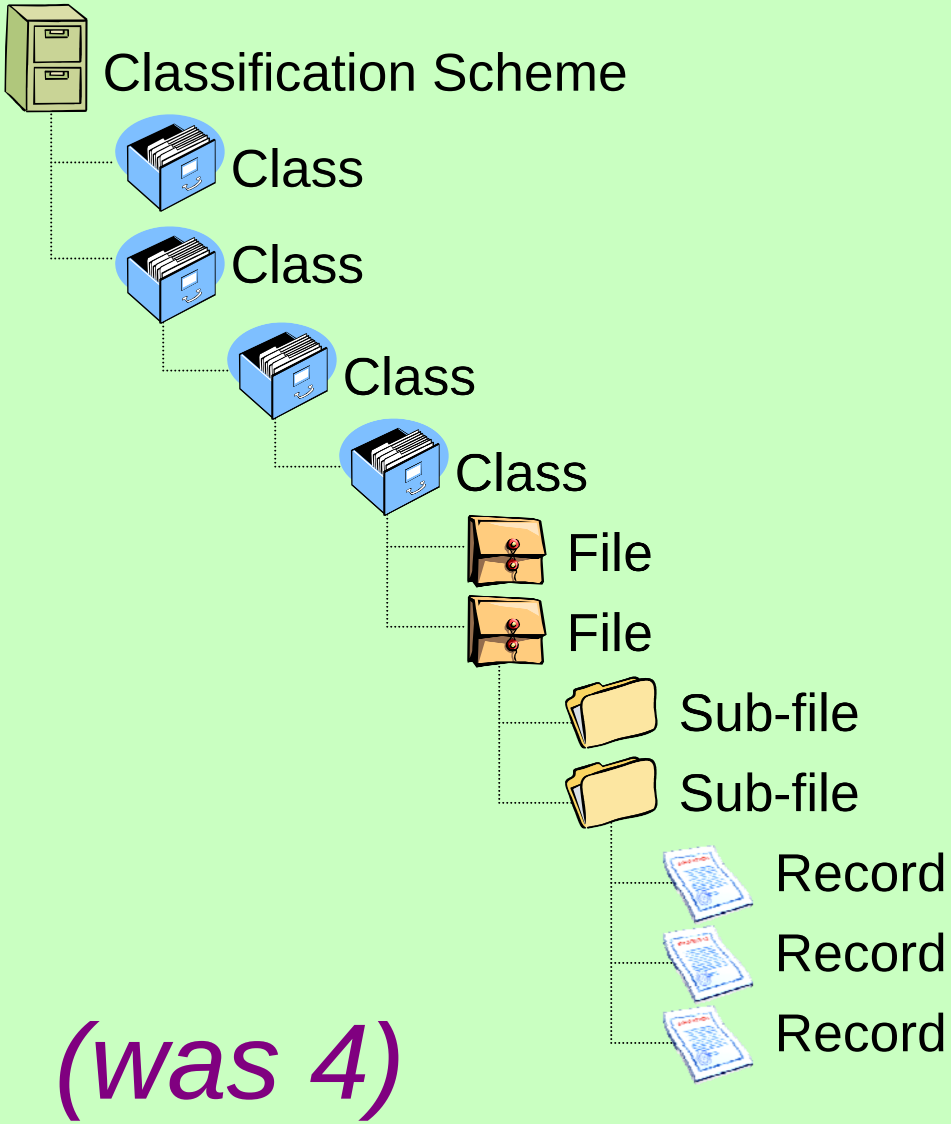
Record

- *This is the only configuration supported by MoReq1 and TNA 2002*

(was 3)

Core + Sub-Files module

(prerequisite for case work module)



Class
File
Sub-file
Record

- *Sub-files are new to MoReq2*

(was 4)

Core + Volumes + Sub-Files

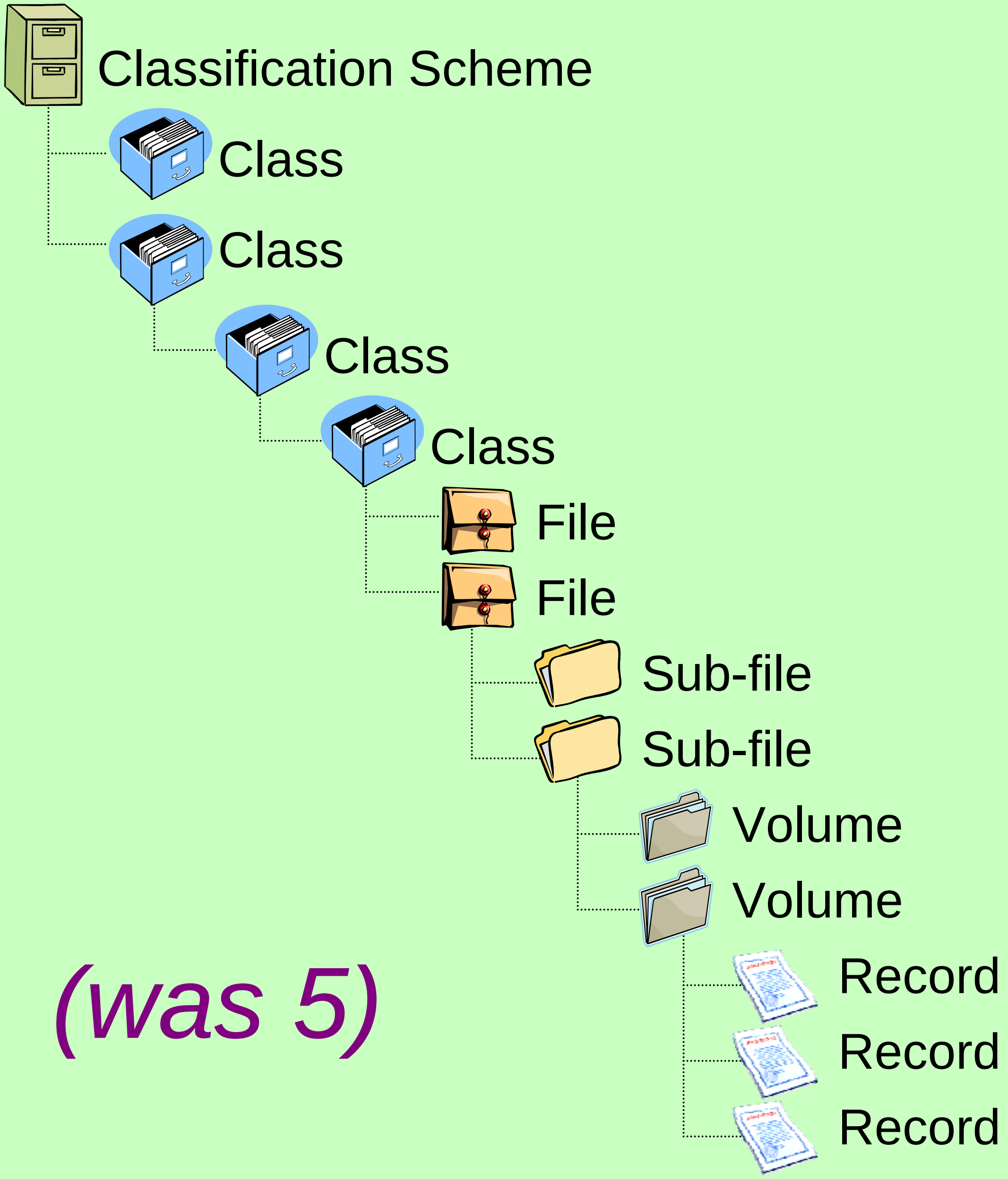
Class

File

Sub-file

Volume

Record



Classification Scheme
- Class
- Class
  - Class
    - Class
      - File
      - File
        - Sub-file
        - Sub-file
          - Volume
          - Volume
            - Record
            - Record
            - Record

(was 5)

- *MoReq2 allows this many levels*

How many businesses need all the complexity of files with sub-files and volumes to hold their records?

Model Requirements

(revisited)

Side Benefit

- By making MoReq2 more modular organisations can start to pick and choose the functionality they need

Not “Model Requirements”

But “Modular Requirements”

Modular Requirements

- Selecting the requirements you need is easy

Reinventing Model Requirements

- What type of system do you need?

	Financial	Government	Health
Case Work			☑
E-mail		☑	
Unfiled	☑		
Physical		☑	
Sub-files			☑
Vital	☑		

Conclusions

- MoReq2 can be refactored relatively easily
- Refactoring reduces both the overall number of core requirements and the complexity of the core
- Refactoring lowers the entry level for basic certification and provides more graduations
- Refactoring allows organisations and industry sectors to tailor-make MoReq2 compliant solutions by selecting modules that are relevant to their business needs

Refactoring allows
future technologies and
approaches to be
included within the
scope of MoReq through
changes to the modules
rather than the core

Questions?