HL7 RIMBAA in Floridaa

Q6 - January 13, 2009

Future Goals/Workitems for RIMBAA WG

- Marketing - Public exposure of successes (of v3 implementability and the RIM itself)
- Sharing of experiences and solutions
- Education - for newbies to RIMBAA
- We focus on Patterns for Application Development. Guidance, no normative outcomes.
- Work with ITS WG on an RS XML ITS and the identification of MS-RS transition issues
- Document/describe (for all possible cell transitions) how those steps could be supported/achieved.
- Identify issues caused by the ‘interoperability mindset’ of the RIM

A brainstorm type discussion about how RIMBAA applications fit within the HL7 SAEAF framework

- RIMBAA is a bit of an oddball implementers group, working on an area that's historically outside of the scope of HL7.. but is it really? It uses the products of the HDF like any of the supported "interoperability paradigms" - it'll be interesting to discuss the implications of including it within the scope of HL7 and its architecture framework.
- Should RIMBAA be part of the architecture framework? Is it already part of the framework?
- Should HL7 change its mission to be about application architectures? Why, or why not?

(MvdZ:) RIMBAA 2009

- P/EHR-S-FM gives Requirements
- SAEAF gives Architectural Framework e.g. for Service Contracts
- SOA/HSSP gives Service Specifications
- RIMBAA gives implementation examples and best practices
- ‘RIM compliant’
- All Model Driven

RIMBAA Introduction

The RIM is “abstract”

The same “instance” of information can be shared via different information interoperability paradigms

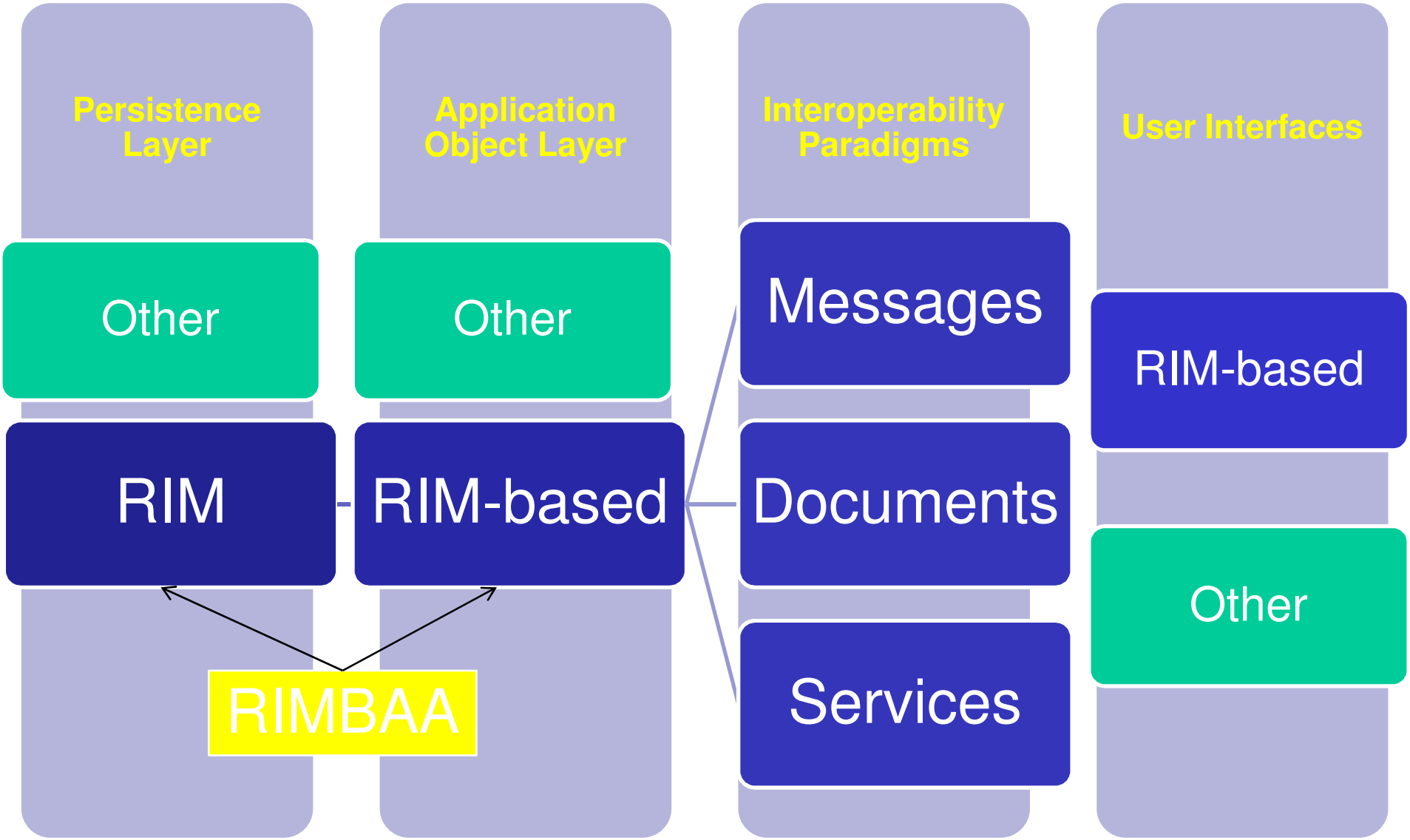
E.g. *lab results* can be shared

- via (electronic) documents (v3 Clinical Document Architecture)
- via V3 lab messages
- via Web Services with v3 payloads

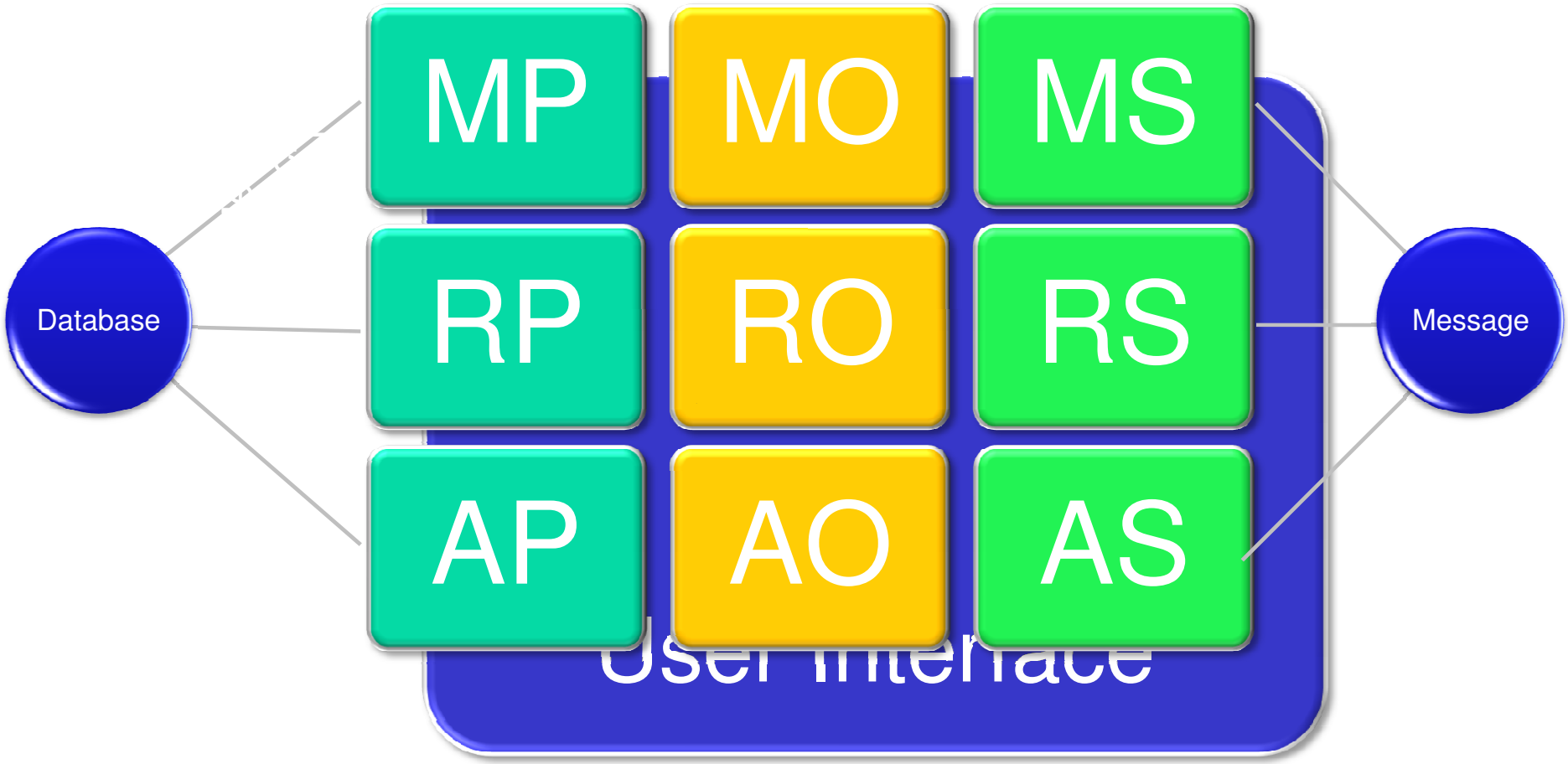
..or they can be imported

- In a RIM Based Application (RIMBAA)

Use of V3 artefacts



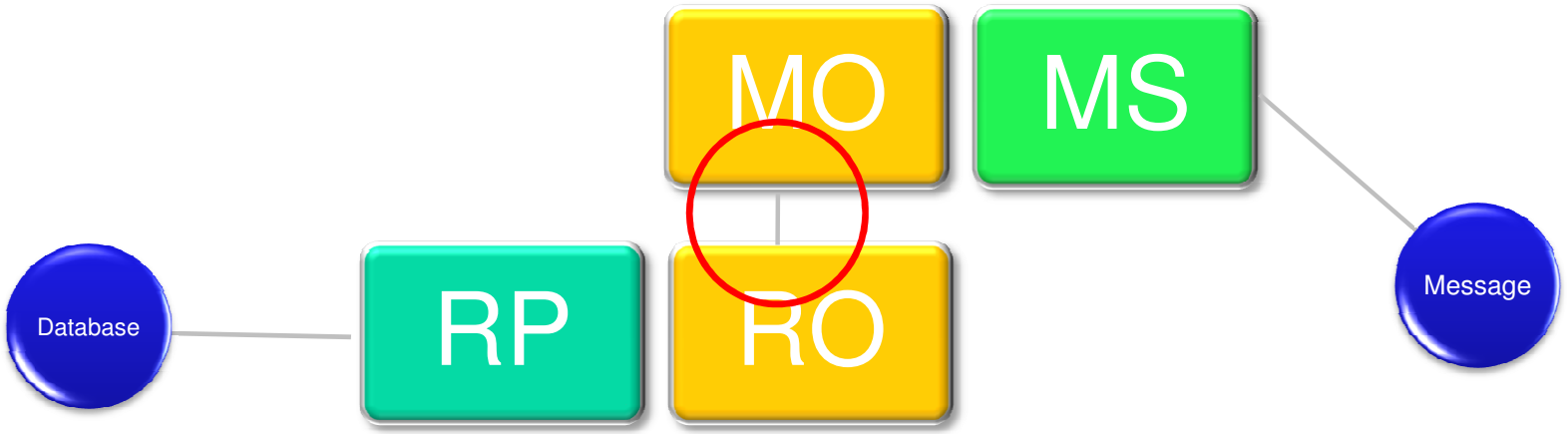
The Technology Matrix



Consistentency of Models

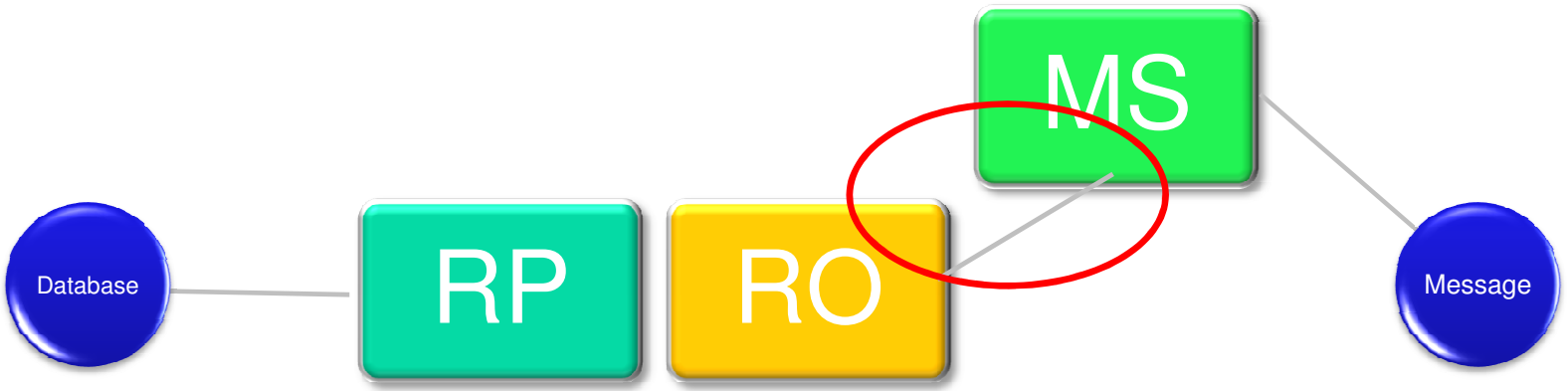
- It is beneficial to use one and the same semantic (reference) model for messaging, in-memory application objects as well as at the persistence layer.
  - This ensures the lowest degree of semantic loss due to mappings between data models.
- Mapping of application internal data models to message models is possible if one only has to support a couple of messages

Use-case 1: Approach



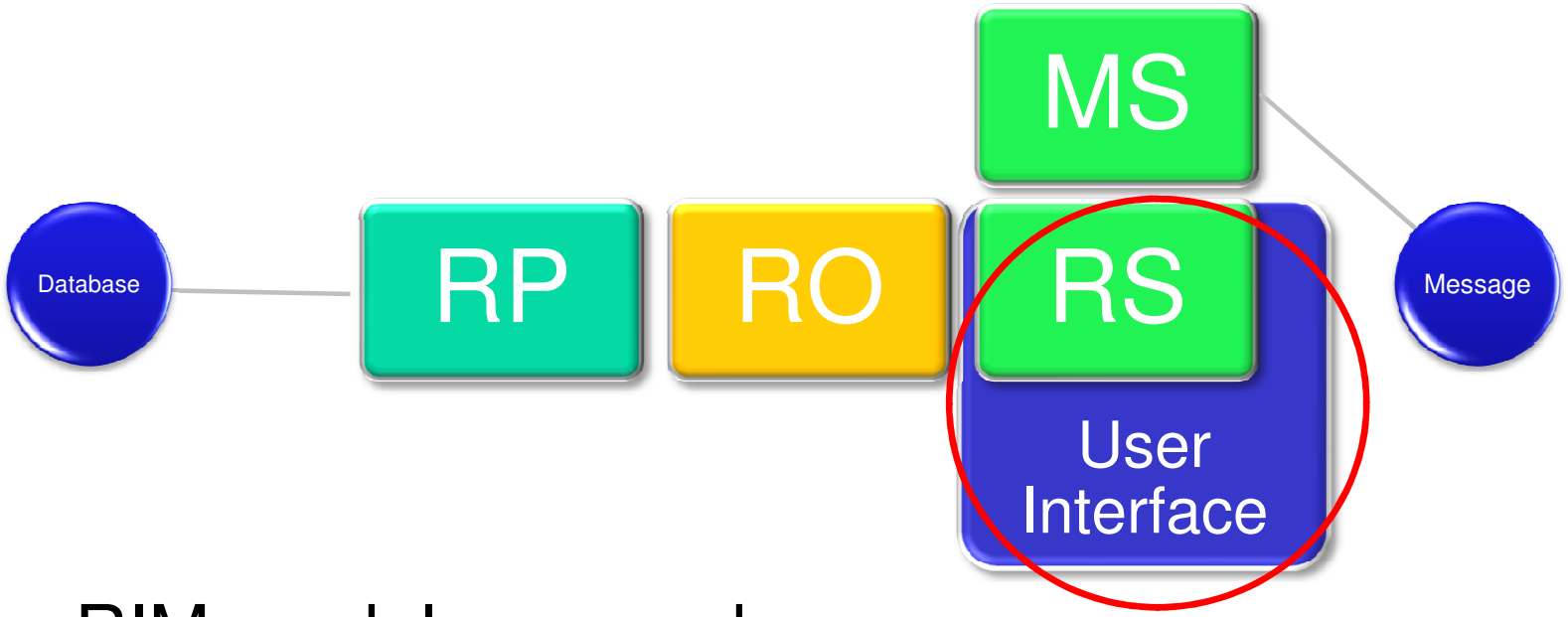
1. Serialized messages mapped to in-memory message objects (JAXB, code generation)
2. Application (business-) objects mapped to relational database (Hibernate ORM)
3. Custom code: mapping of message objects to application objects

Use-case 2: Approach



1. RIM based objects at core of application
2. Direct (MIF-based) mapping to/from serialized messages
3. Direct mapping to relational database

Use case 3: Approach



1. Use RIM-models everywhere

   ..but no R-MIMs

2. Map serialized RIM-derived object trees (in XML) to HL7 v3 interactions

   .. Uses XSLT

Additional RIMBAA materials

Read/view in this order:

http://www.youtube.com/watch?v=tglo-om-r2o

http://www.ringholm.de/docs/03100_en.htm

http://www.youtube.com/watch?v=0jtVaOMGRZY

Known Implementations:

http://wiki.hl7.org/index.php?title=Category:RIMBAA

(username = wiki, password = wikiwiki)

RIMBAA working group

RIMBAA e-mail list at www.hl7.org