

The Unified Service Action Model II Proposal (USAMP-II).

Gunther Schadow
Regenstrief Institute for Health Care

USAMP-I recapitulated

1

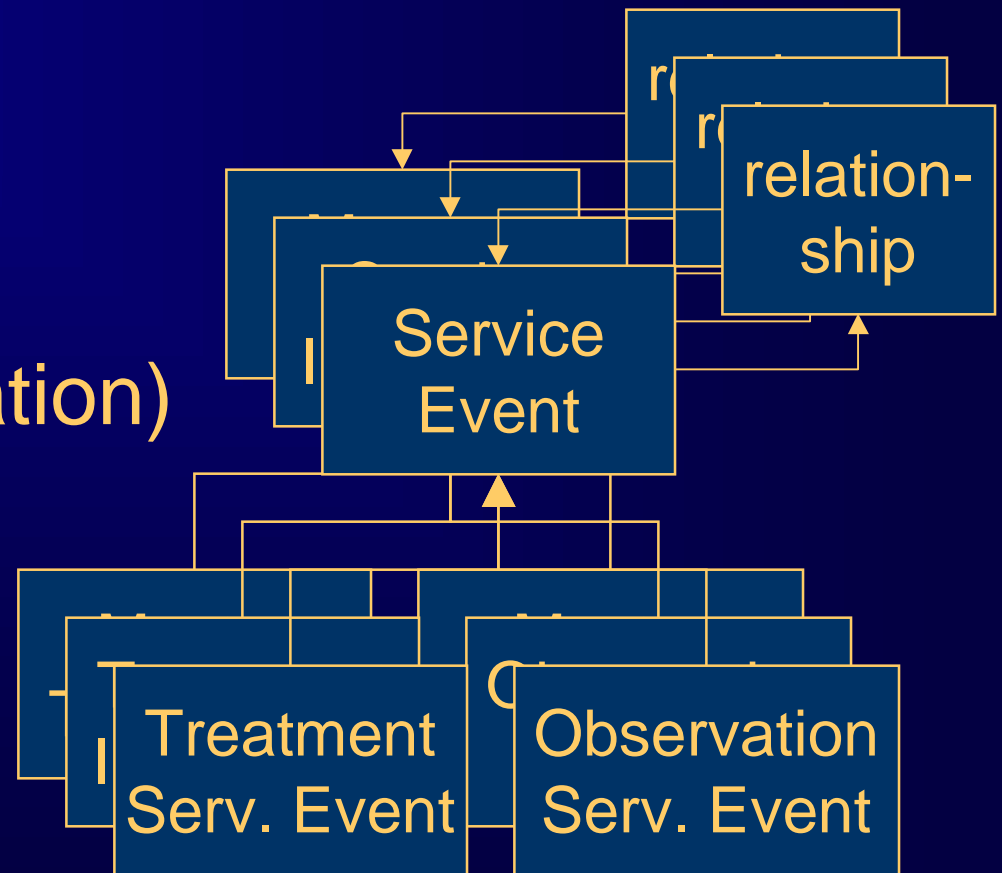
- Motivated from the clinical patient care perspective.
- ***Unified*** the Observation-Result with the Observation-Service as the two sides of the same coin.
- Thus assumed a healthcare ***service action*** oriented perspective for all electronic medical record (EMR) information.
- Offers itself to administration and cost controlling.

USAMP-I recapitulated

2

- USAMP-I defined the action on three levels, in accordance with the RIM.

1. Master (catalog)
2. Intent (or order)
3. Event (documentation)



So, what is left
to be done?

1

The Problem

(Too) many quality issues

- Can any single person, or even the entire committee account for every single model feature?
- Can the model represent all that needs to be represented?
- Can we build efficient messages?
- Is message design efficient?
- How do we expect to deal with future requirements?

Attributes need more analysis

- Many ad-hoc coded elements
 - procedure_medication_cd
 - service_contraindication_cd
 - target_of_service_cd
 - end_condition_cd
 - total_daily_dose_cd
- And worse, many free-text fields:
 - challenge_information_txt
 - service_requirement_desc
 - user_defined_access_check_txt

Deleterious documentation

- Our documentation as HL7's front desk:

*Excuse me please,
what is the "**total
daily dose cd**"?*

*Oh, sure, it's the
**total daily dosage
code!***

(No documentation at all is actually better than this, since it leaves you with a sigh, not with anger.)

More deleterious documentation

verification_required_ind?

“An indication that verification is required.”

Master_treatment_service.

drug_category_cd

“The drug category code of the master treatment service”

Attributes that will never work

- **charge_type_cd**

“A code identifying someone or something other than the patient to be billed for this service.”

- It can only be *either* a coded concept *or* an identifier for a person, not both.
- If it’s an identifier to something else, it’s a foreign key and thus an MDF violation.
- If it’s a coded concept, could there possibly be a complete and interoperable coding system?

Some things are plain wrong

- **Clinical_observation.value_type_cd**

“A qualifier of the observation value.”

- It really is the data type of the attribute “observation_value_txt” (OBX heritage)
- The documentation sounds as if the documenter himself wasn’t quite sure.

Inconsistency

- **Master_service.**
allowable_processing_priority_cd
- Processing priority sounds like a good idea.
- The allowable processing priority would be a set of priorities from which to choose (e.g. STAT, ASAP, REGULAR)
- But there is no corresponding “processing_priority_cd” field anywhere else!

Incoherence

- **Master_service.**
 .confidentiality_cd
- **Service_intent_or_order**
 has no way to specify a confidentiality.
- **Service_event.**
 .confidentiality_ind
 .patient_sensitivity_cd
 .user_defined_access_check_cd
- Is there any *reason* for this, no *excuse*?

Enough of this!

- This is not to blame any one of the few people who care for the RIM.
 - The blame falls on all of us from outside.
 - E.g., has anyone the energy to really address Angelo Rossi-Mori's quality issues?
- The clinical RIM area has 34 classes and 325 attributes ...
- but only about 4 people who actively maintain them!
 - The harmonization process is time consuming and cumbersome, it punishes those who care...

2

A Solution

Design goals (as usual)

- **Simplification**
 - How can we contain the number of attributes?
 - How can we make the model understandable?
- **Rationalization**
 - How can we avoid to be embarrassed by info junk?
 - Where is the big picture? Where the guiding principle?
- **Maintainability**
 - How can we assure continuing quality?
 - How will we accommodate new requirements?

Our current strategy?

- We don't need to contain the number of attributes,
 - because the best way to manage information is by keeping each distinct concept in a separate attribute.
- No need for rationalizing anything,
 - HL7 users know what is right, and don't want to be told how to think,
 - they only have to find their stuff in the model.
- The quality problem is only transient,
 - we only need to enforce style rules more rigorously.
- New requirements deserve new classes and attributes.

An alternative strategy

- Models are abstractions and simplifications of the real world.
 - The more complex our world is, the more abstraction do we need.
- Rules are easier to comprehend and manage than detail.
 - We must find a guiding logic/semantic principle from which to develop top-down.
- We can provide for new requirements now
 - by doing the logically correct thing instead of what is customary.

Find a principle

- Let the *action* be a guiding principle around which we sort *things* out.
 - Linguists and good writers know: the verb (action) is the soul of a good sentence.
 - The action brings together people, locations, and material resources.
 - The health care action is the reason for our business.
 - Labor is what costs the most money.
 - Facts exist only if we take action to produce those facts.

Find redundancy

- The world is HUGE and *complex*.
- The human brain is small.
- Mankind can only cope with the complex world by exploiting its redundancies.
 - Cataloguing the variety is only a first step to discovering unifying principles.
 - Language reflects the economy of the human mind: we separate verbs from nomina and create new worlds with verbs.
 - We want, we did, but we haven't tried yet, so we could, ..., and then we do!



Event, intent, master

- Any service event is an intentional action and can potentially be ordered.
- An event has a number of parameters
 - when, how, how much, how long, etc.
- An intent or order specifies some or all of the service parameters as desired values.
- The master service defines each of the parameters
 - the allowable values, or
 - the preferred defaults

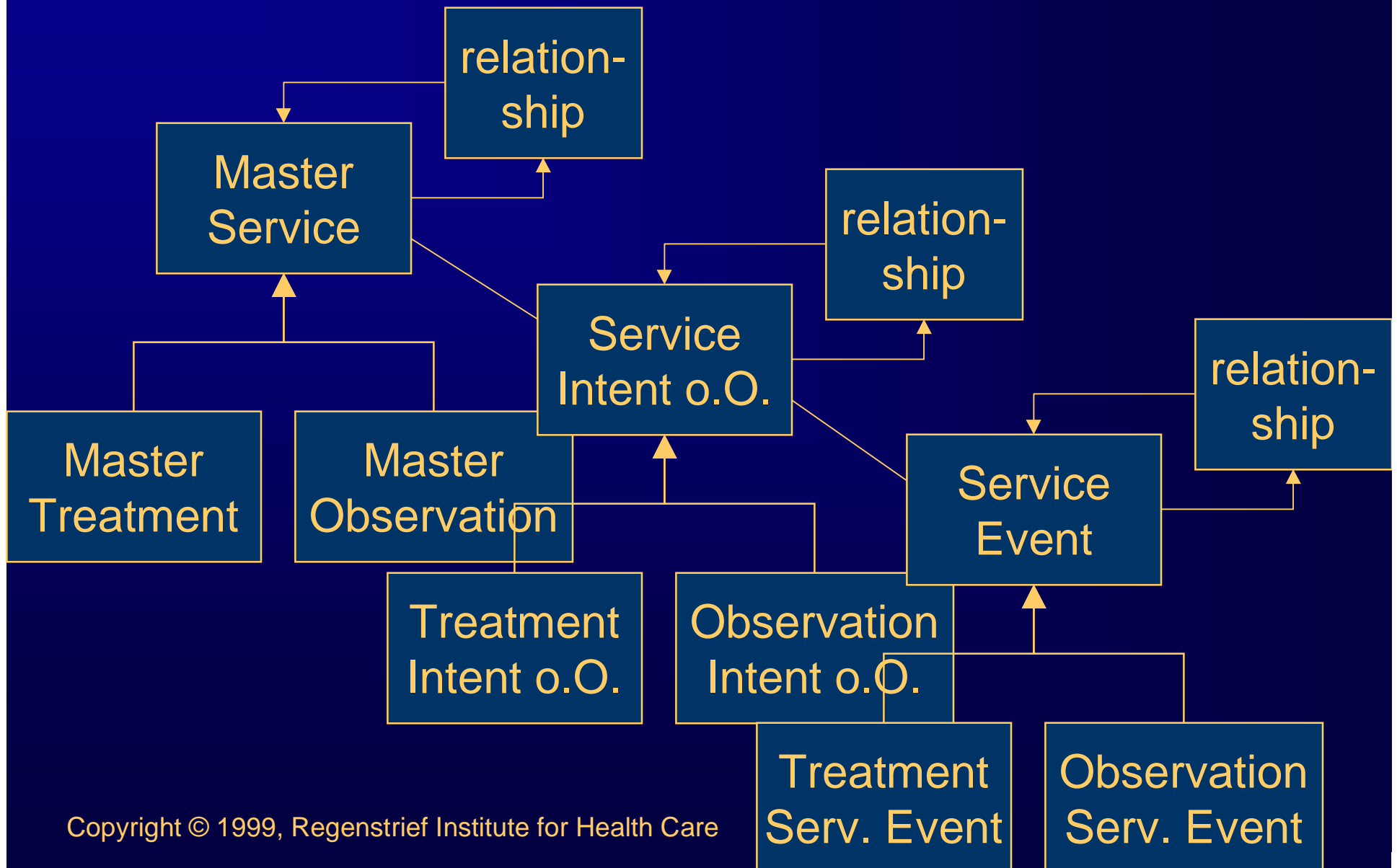
Master, intent, event

- Master specifies
 - what can *potentially* be done,
 - how it's *usually* done,
 - what the *possible* outcomes are.
- Intent or order specifies
 - what one *is supposed to* do,
 - how one *is supposed to* do it.
- Event specifies
 - what has *actually* been done,
 - how it has *actually* been done,
 - what the *actual* outcomes are.

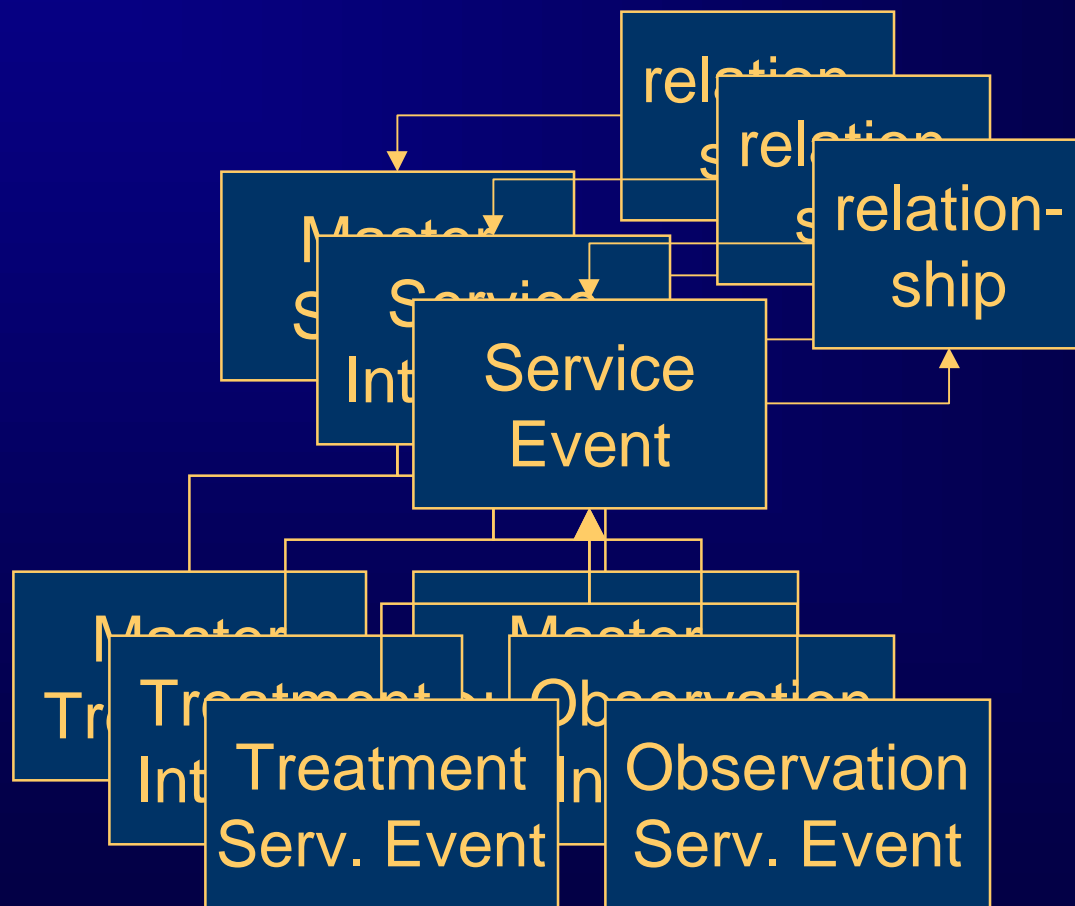
Data types and values

- Master
 - possible site: {hand, forearm, cubit, foot, femoral, ...}
 - normal body mass dose: 5 mg/kg
- Order
 - preferred site: {hand, forearm}
 - ordered dose: 4 mg/kg (x 70 kg = 280 mg)
- Event
 - actual site: forearm
 - administered dose: 250 mg
- try the game with the RIM synopsis ...

We are here

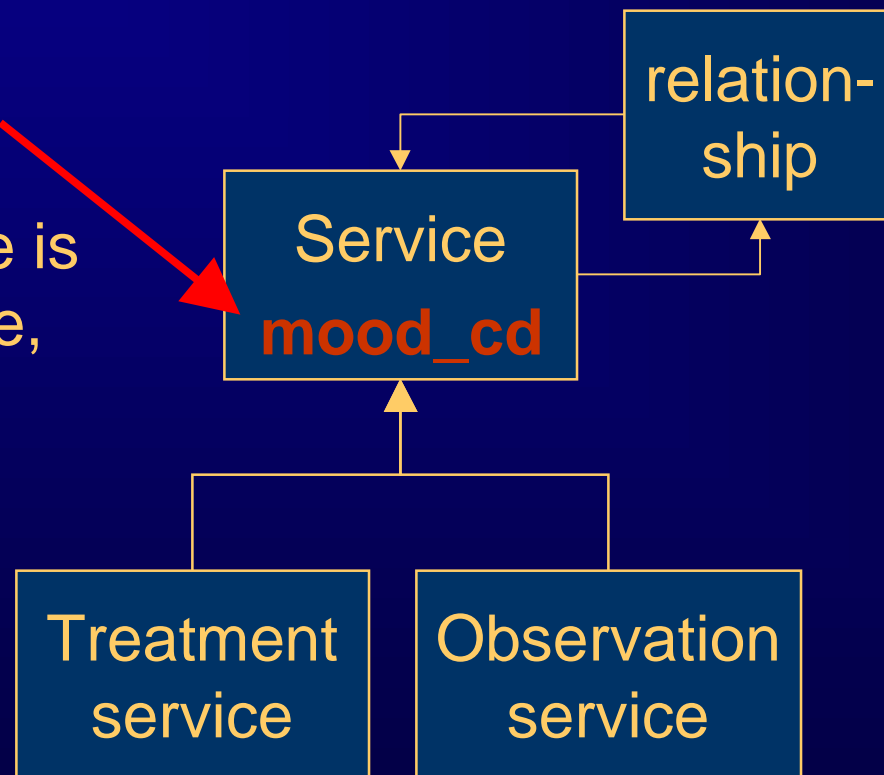


Analyzing

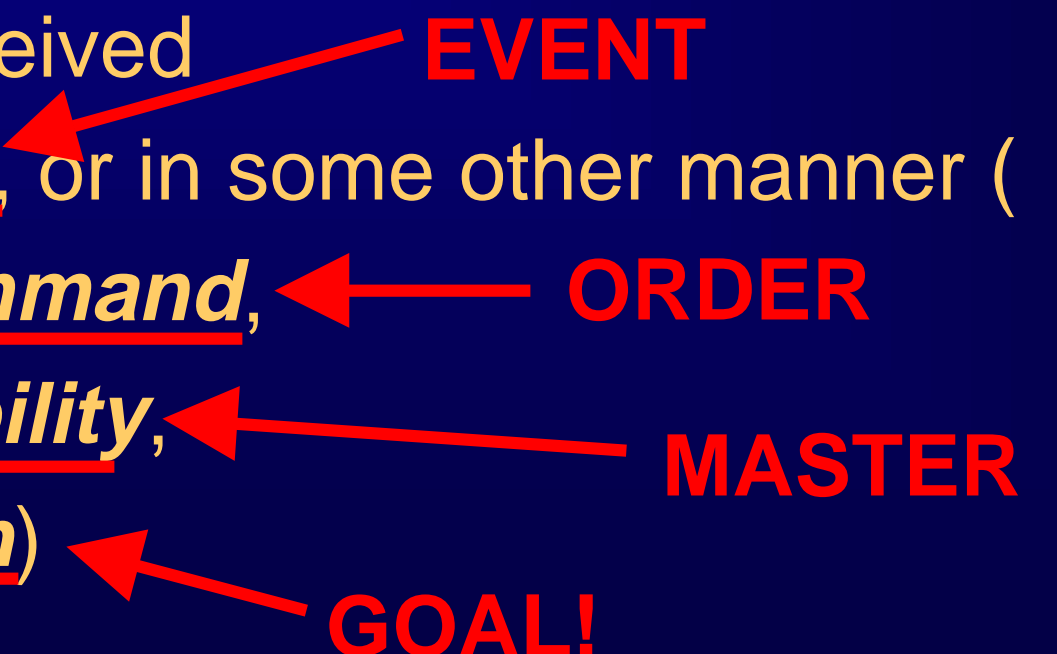


Unifying

The mood code tells whether a service instance is a master service, an order, or an event, or ...



In the mood?

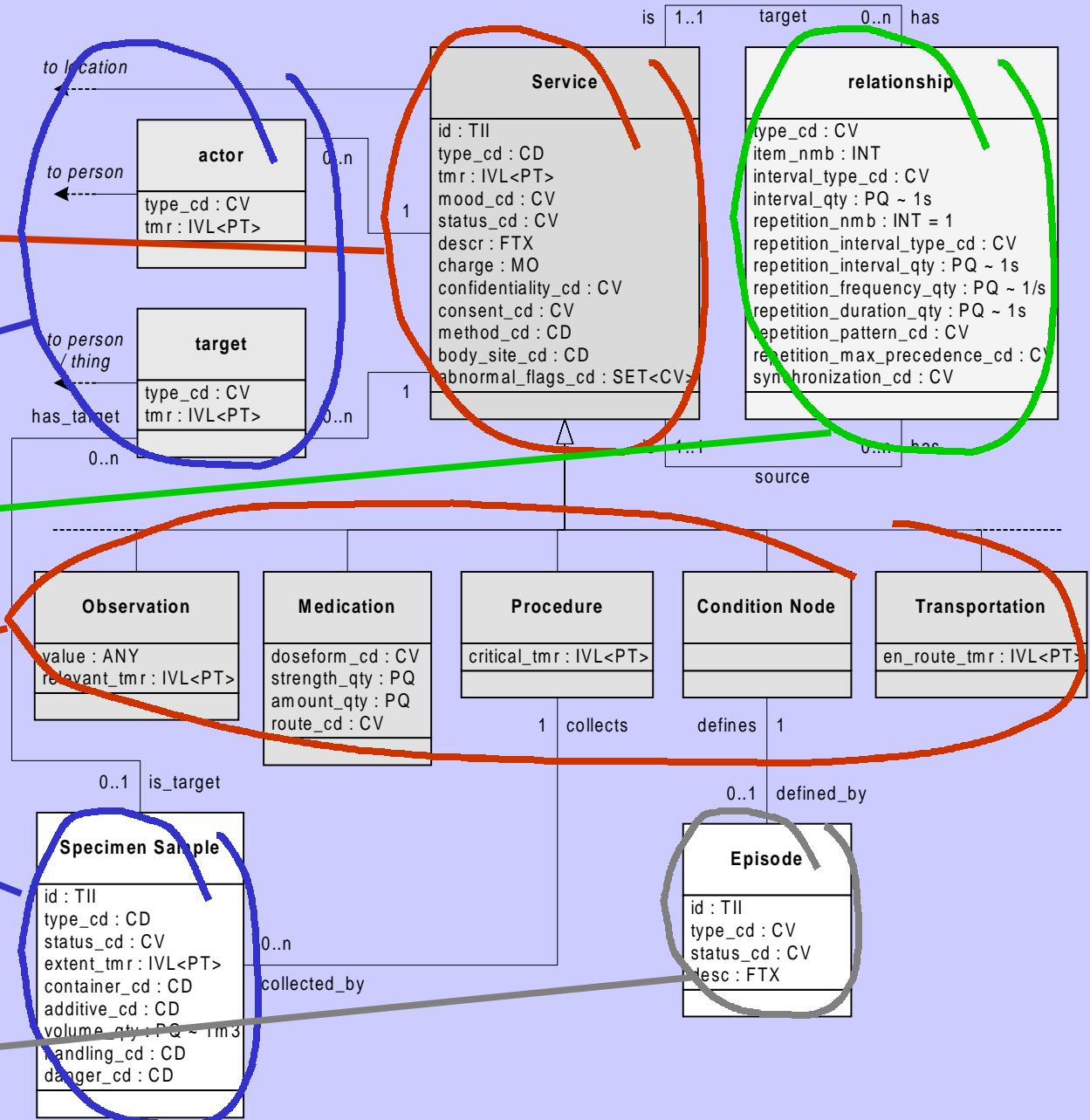
- Mood (2) etymology: alteration of *mode*.
 - “Distinction of *form* [...] of a verb
 - to express whether the *action* [...] it denotes is conceived
 - as fact, or in some other manner (
 - as command,
 - possibility,
 - or wish)
- EVENT**
- ORDER**
- MASTER**
- GOAL!**
- 

Analysis of moods

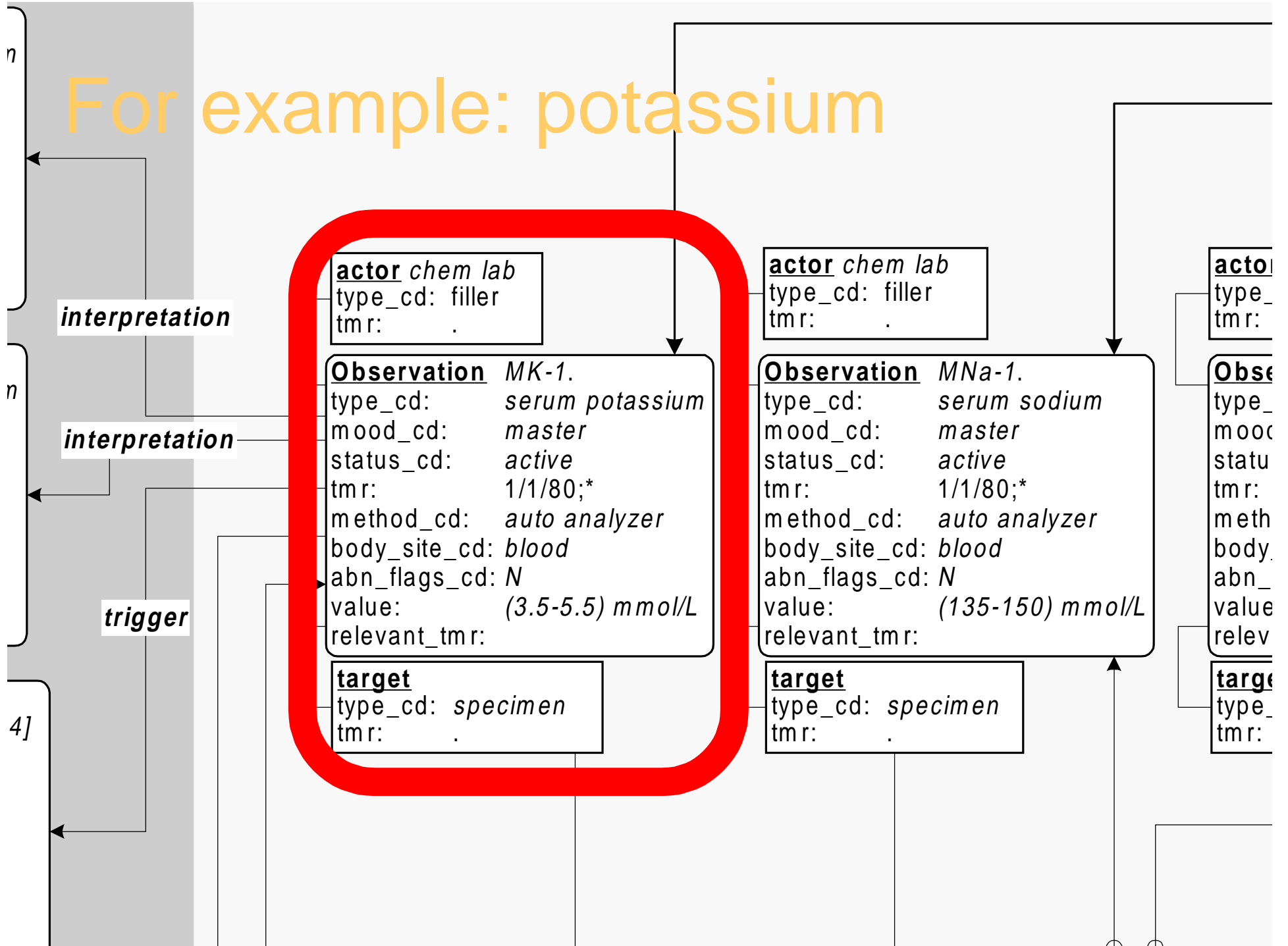
- Infinitive “to do”
 - dictionary form
 - potential actions
 - goals
 - triggers (for PRN orders, guidelines, alerts)
- Indicative / actual
 - present perfect “we have done” (report)
 - past “someone did” (history)
- Imperative / future
 - order: please do!
 - Plan (intent): I will do.

Let's do it!

- Service action
- who, whom, where, ...
- The omnipotent relationship.
- Specific details of actions
- Specimen (and other *things* hopefully soon)
- Episode (and other groupers?)



For example: potassium



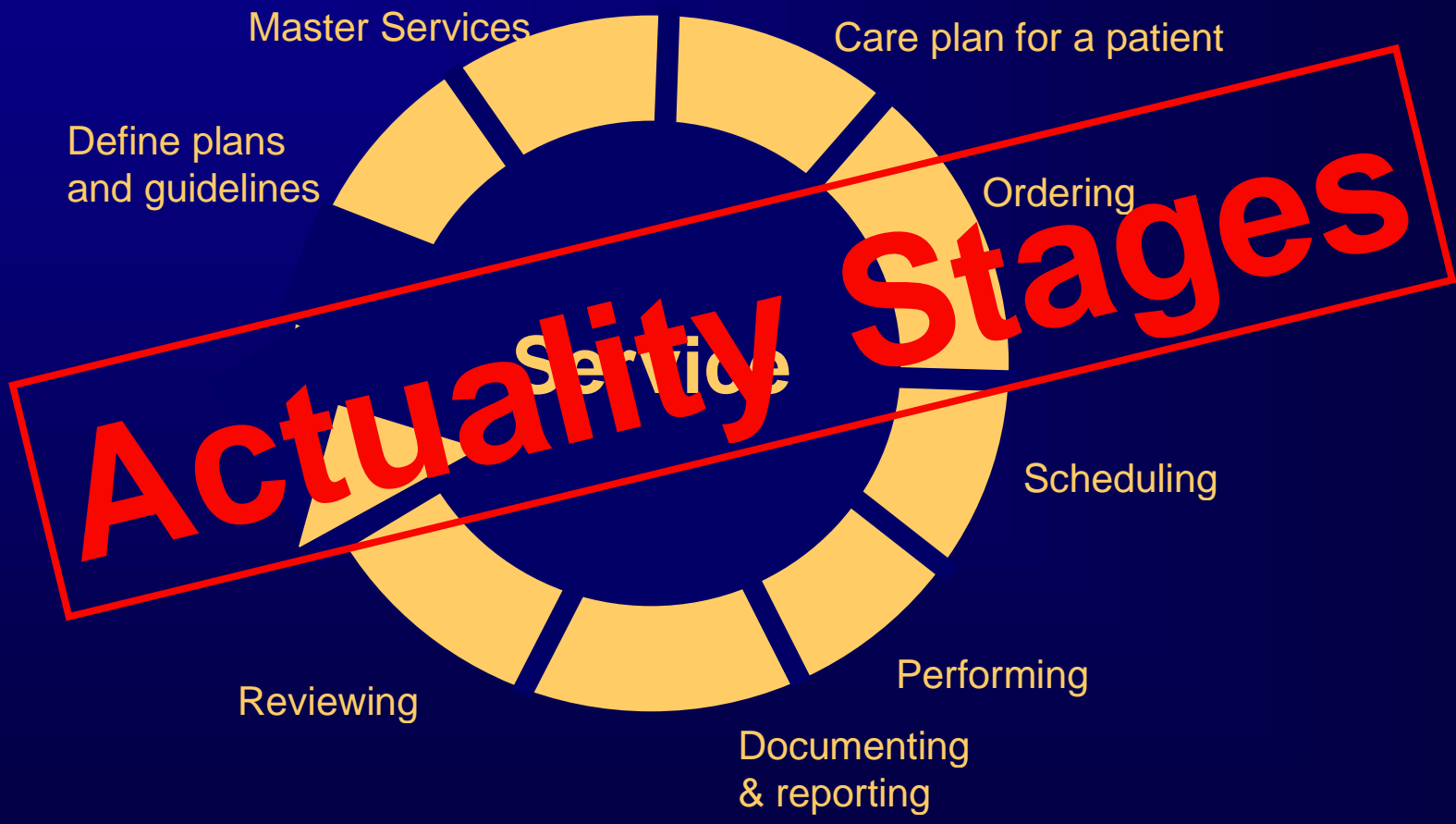
3

Issues and Misunderstandings

States and Transitions?

- State transitions occur in one instance.
- The step from master to intent (or order) is certainly not a transition.
 - A new instance of the service is instantiated.
 - Master is the prototype, that is cloned and modified.
- Arguably the step from order to event is not a transition either.
 - A “filler order” is created from the “placer order”.
 - Filler order is the prototype, that is cloned and modified

What, if not states, is this?



Class versus Instance

1

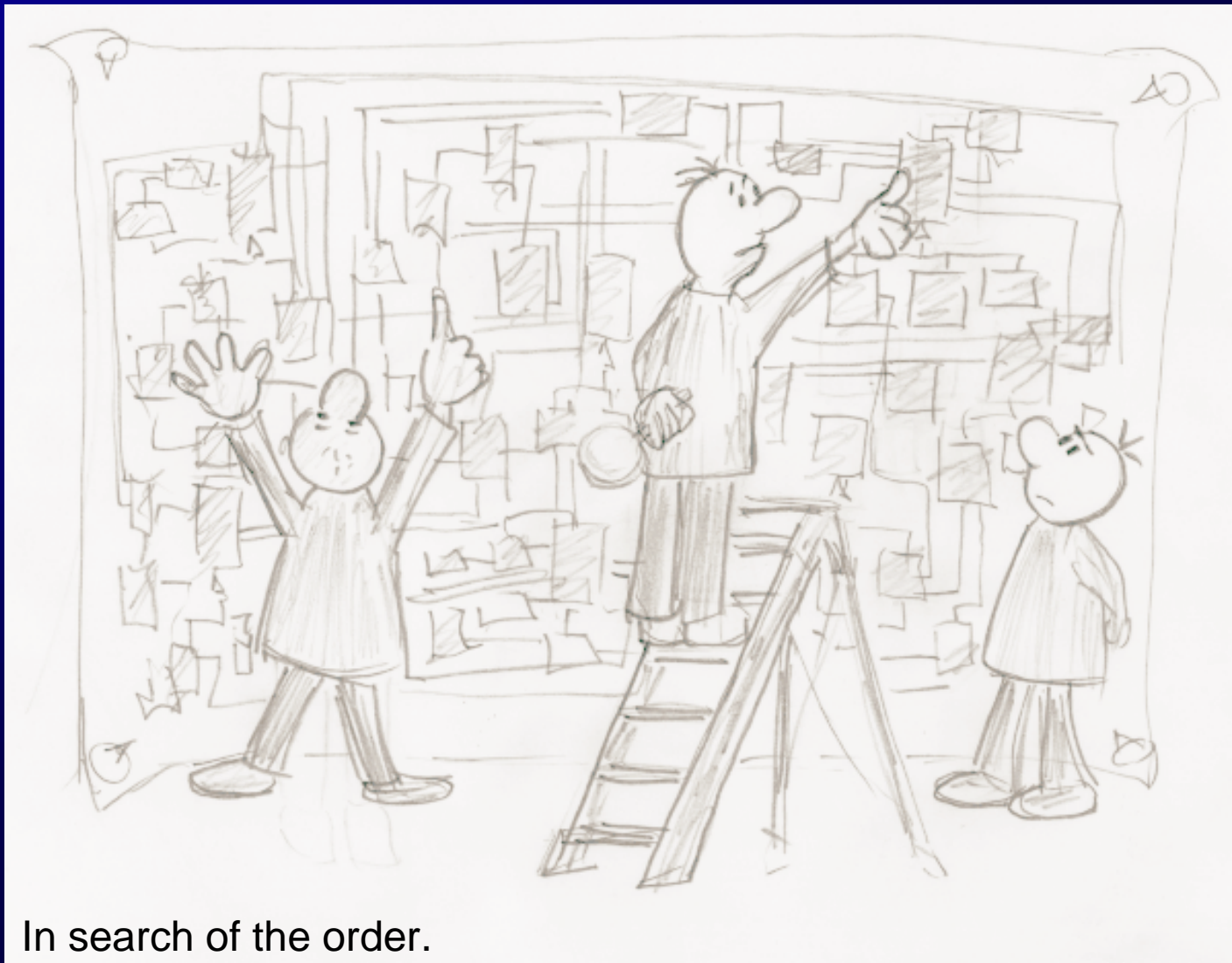
- Given that the master is not the same instance as the order, and the event is yet another instance ...
- this means that we must model those as different classes?
- NO! The only hard constraint in our modeling is that one instance can not span multiple classes.
- Other than that, anything goes.

Class versus Instance

2

- There are vendors who have long followed the one-service model ...
- if one instance can not span multiple classes, does this mean we decide that their implementations are wrong?
- **NO!** The RIM is a “messaging model.”
 - The RIM is not even a true object model, for we don't use object methods.
- We are **FREE** to model information as we see fit.

Clarity of the model?



In search of the order.

Clarity of the model?



About complexity of models

- Any health care information model should reflect the healthcare domain as good as possible.
- The healthcare domain is complex
 - Probably the most complex information domain that has ever been addressed by an information model.
- Thus, no healthcare model can be simple.
- Simplicity at the cost of accuracy and flexibility is in nobody's interest.

Documentation for BIG model



HL7 user seeking guidance.

Documentation for BIG model



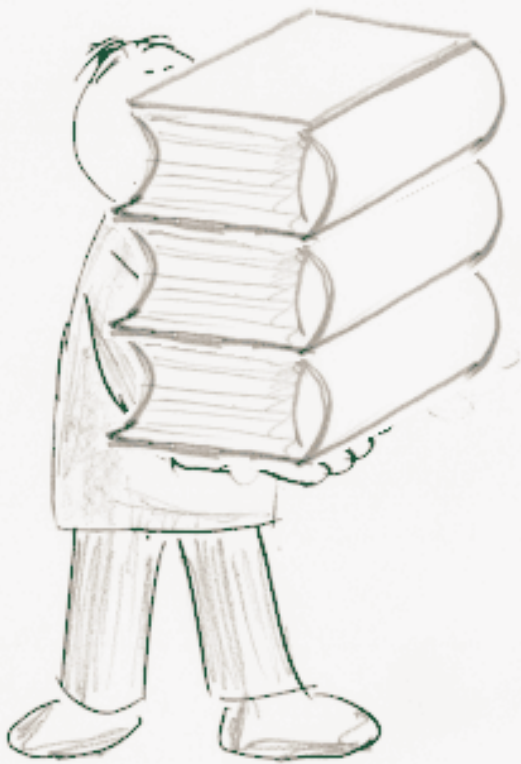
- Message Development Framework

Documentation for BIG model



- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,

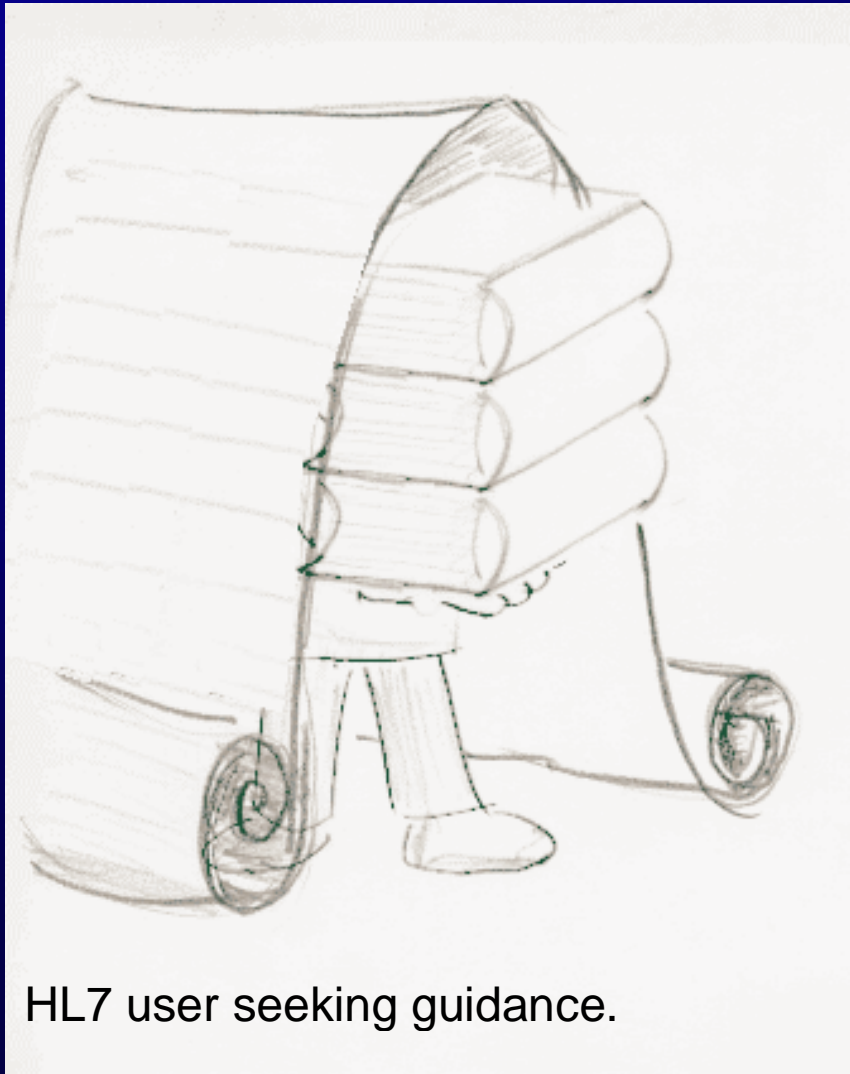
Documentation for BIG model



HL7 user seeking guidance.

- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,
- Grid and index to find items on the diagram,

Documentation for BIG model



HL7 user seeking guidance.

- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,
- Grid and index to find items on the diagram,
- The diagram itself.

Documentation for small model



HL7 user seeking guidance.

Documentation for small model

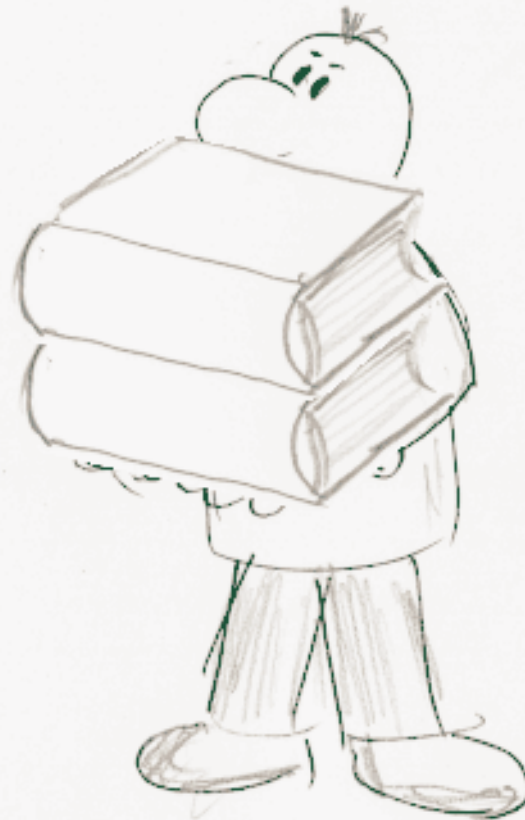
- Message Development Framework,



HL7 user seeking guidance.

Documentation for small model

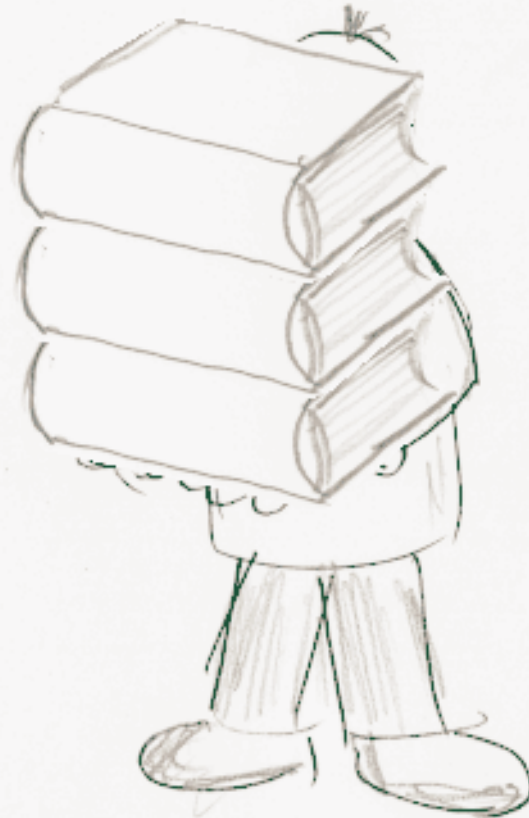
- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,



HL7 user seeking guidance.

Documentation for small model

- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,
- Quick reference of concepts,



HL7 user seeking guidance.

Documentation for small model

- Message Development Framework,
- Definitions of model elements (classes and attributes) and guide to their use,
- Quick reference of concepts,
- The diagram itself.

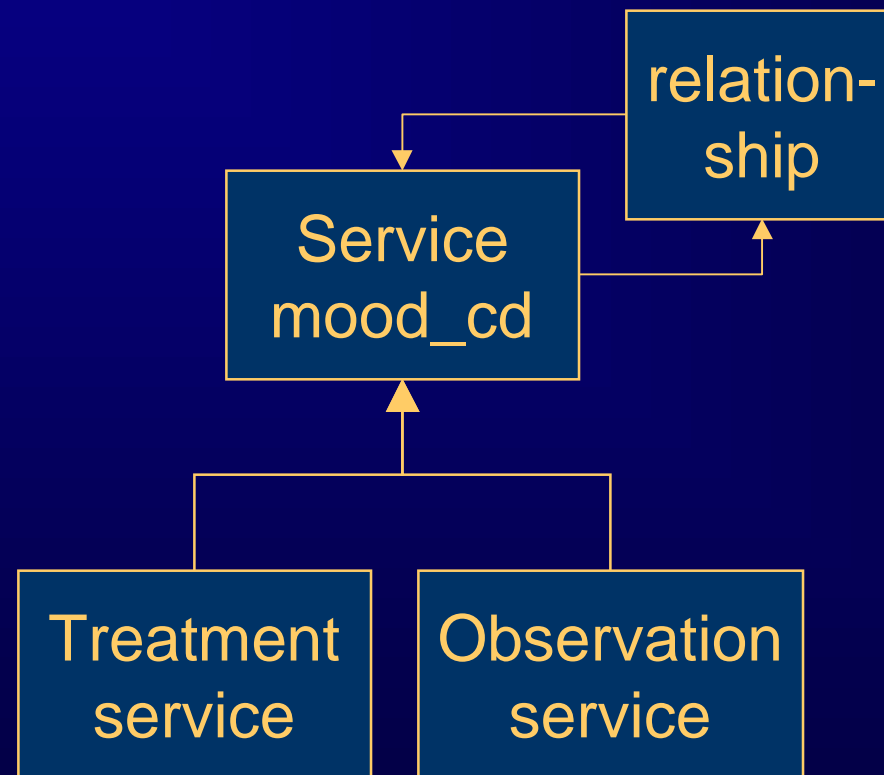


HL7 user seeking guidance.

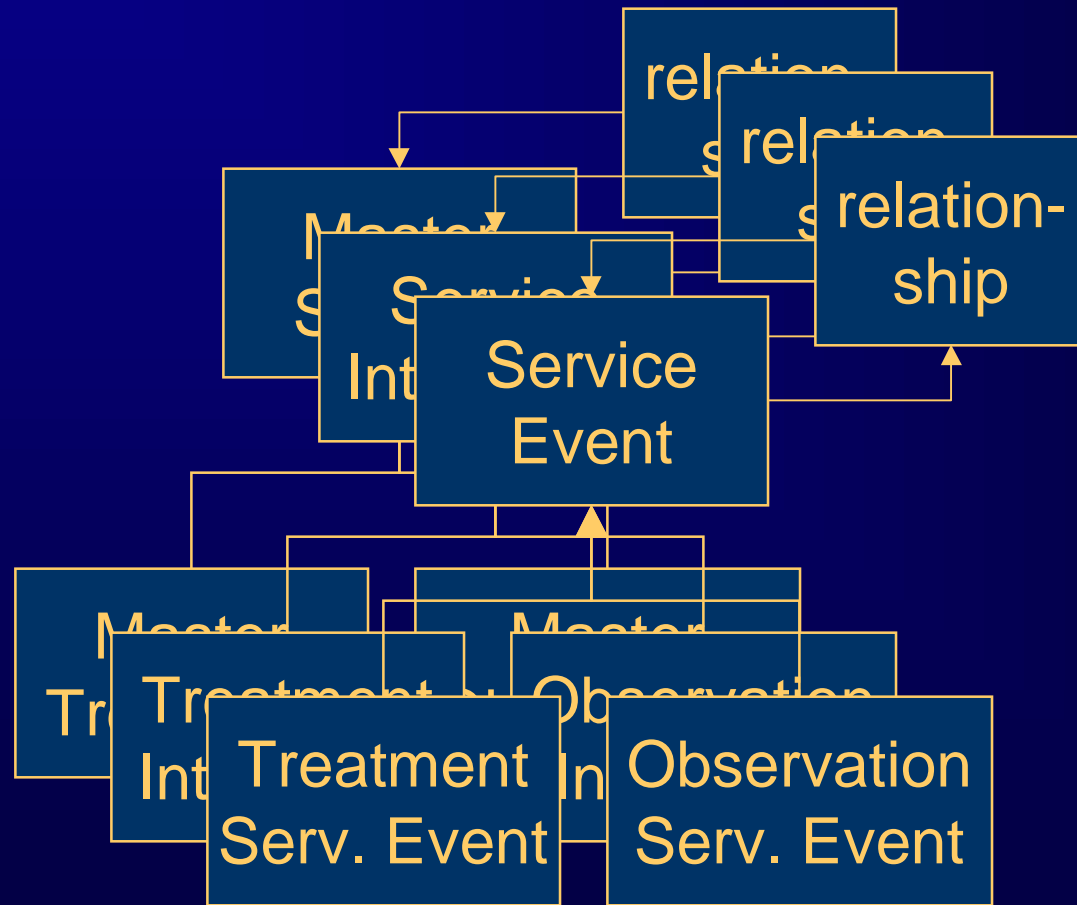
Compromises

- Linda reluctantly suggest to make a common root class for all three hierarchies.
 - Will not influence the detail at the leaf-level
- Orthogonal inheritance of service mood.
 - Is not “correct” UML modeling
- Make special classes for master, order and event,
 - with instance connections to the service *descriptor*.
- Magic with mapping, and multi-layer modeling.

The “lumpers” model ...

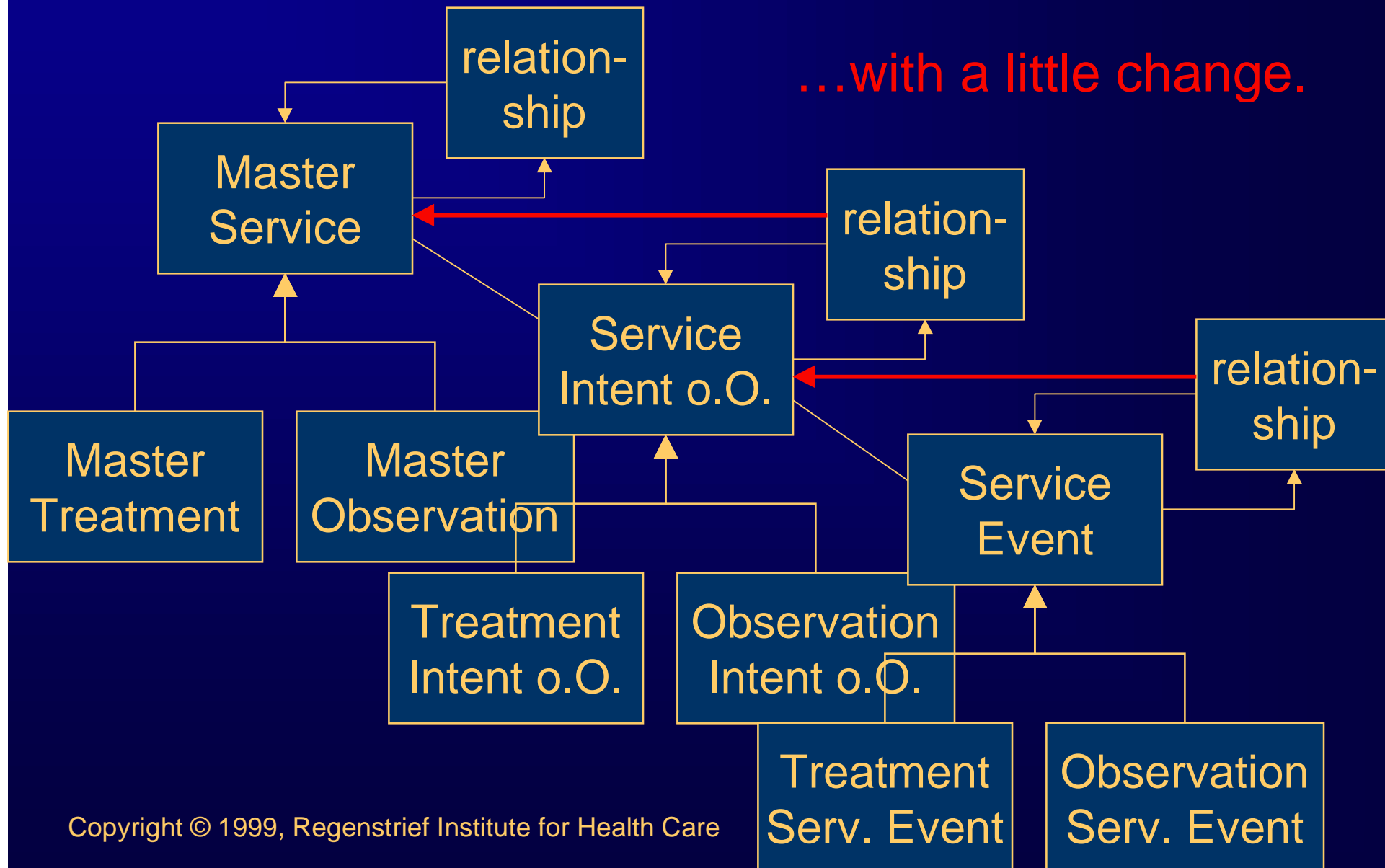


... mapped to ...



... the "splitters" model.

...with a little change.



It isn't so different:

- whether you say

OB1 | ... *data elements* ... |

OBR | ... *data elements* ... |

OBX | ... *data elements* ... |

- or if you say:

OB | 1 | ... *data elements* ... |

OB | R | ... *data elements* ... |

OB | X | ... *data elements* ... |

with this being the
mood code



Conclusion

- Model “size” reduced to 1/6 of the original.
 - Model fits on one letter-size page!
 - Model is easier to maintain and keep consistent.
 - Model is more dense, but that may actually facilitate true understanding of the model.
- This is a very flexible and powerful model.
 - It can assimilate a complex and changing world.
 - A convenient home for templates and decision support.
- The impact on our practice is not so big
 - We don't have to relearn everything.

So, please consider!