



Bridging Epoch: Mapping Two Clinical Trial Ontologies

10th International Protégé Conference
July 17, 2007

Samson W. Tu¹, Douglas Fridsma², Ravi D. Shankar¹,
Martin O'Connor¹, Amar K. Das¹, David B. Parrish³

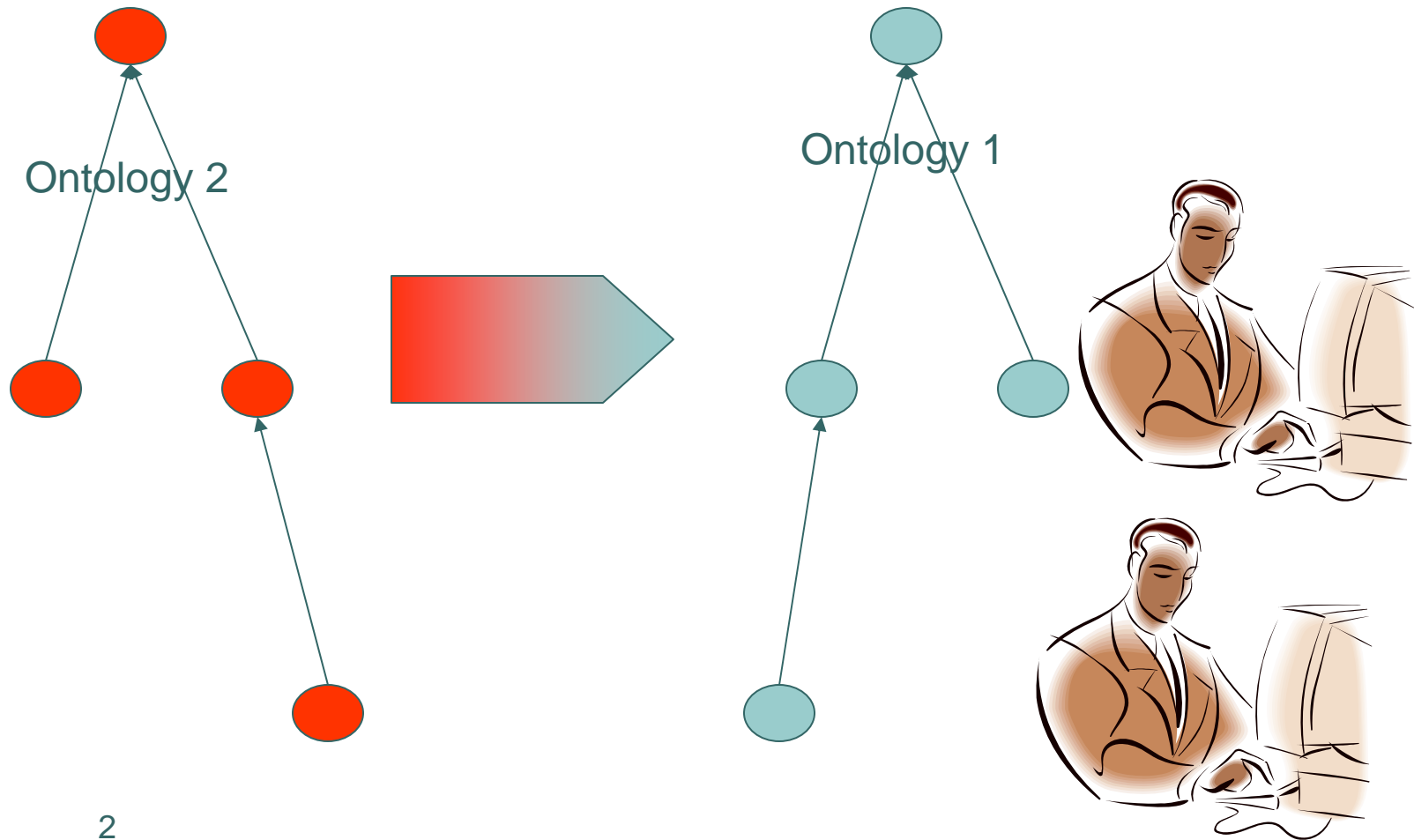
¹Stanford University, USA

²University of Pittsburgh, USA

³The Immune Tolerance Network, Pittsburgh, PA, USA

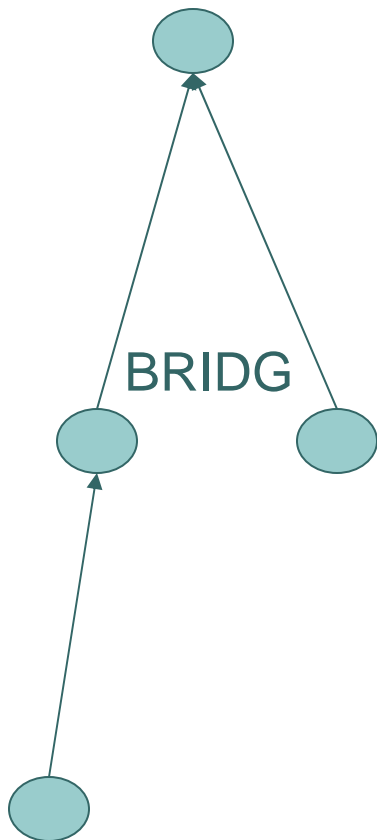


Problem: Ontologies and semantic interoperability





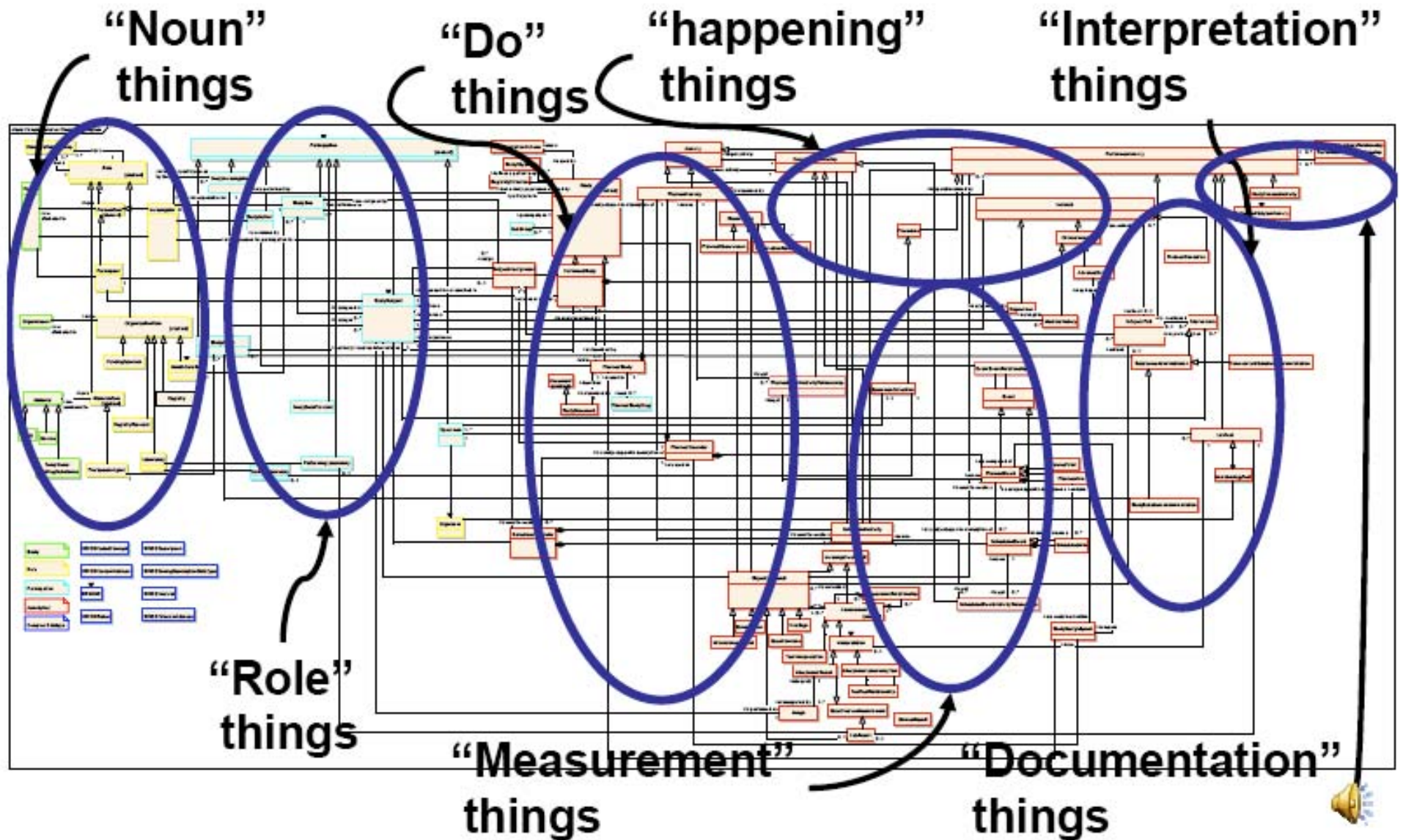
BRIDG: Biomedical Research Integration Group



- Part of US NCI Cancer Biomedical Informatics Grid (caBIG)
- Stakeholders include US FDA, HL7, CDISC
- Create shared domain model for **protocol-driven clinical research**
 - Comprehensive
 - Consensus-based
 - Abstract and context neutral

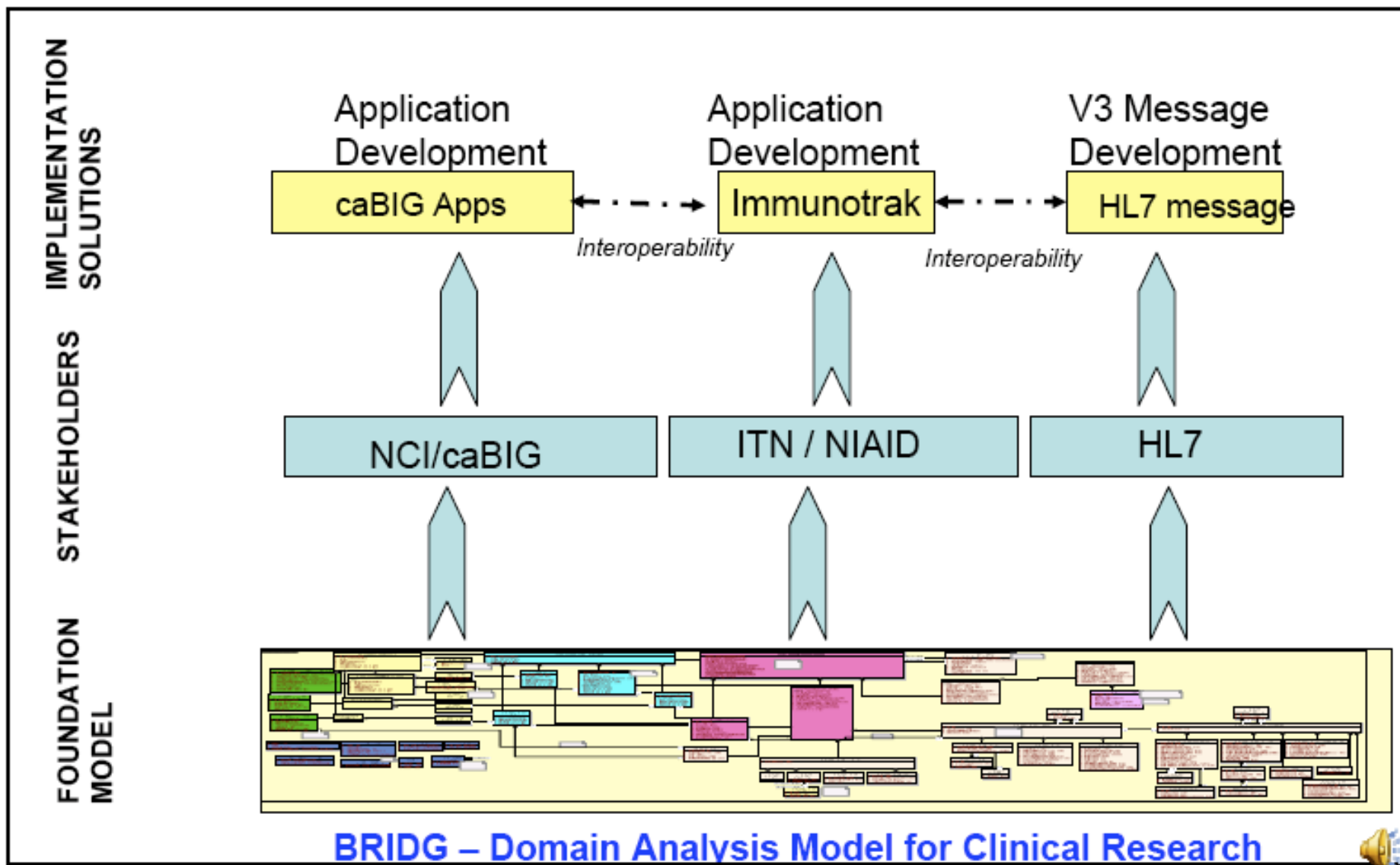


Current Classes in Core Elements

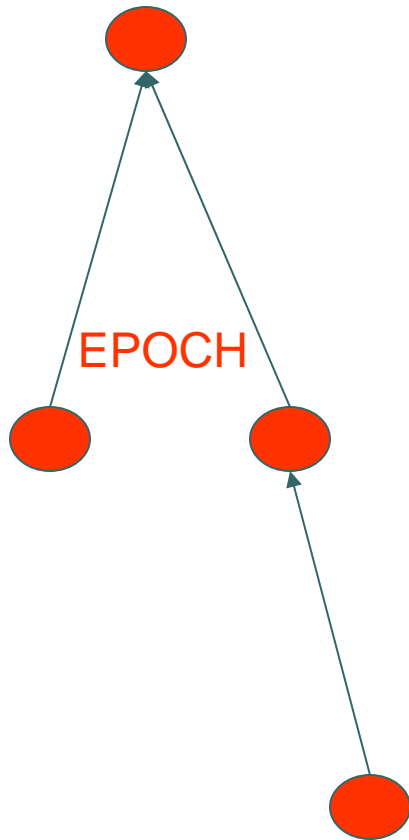




Achieving interoperability from a common semantic foundation



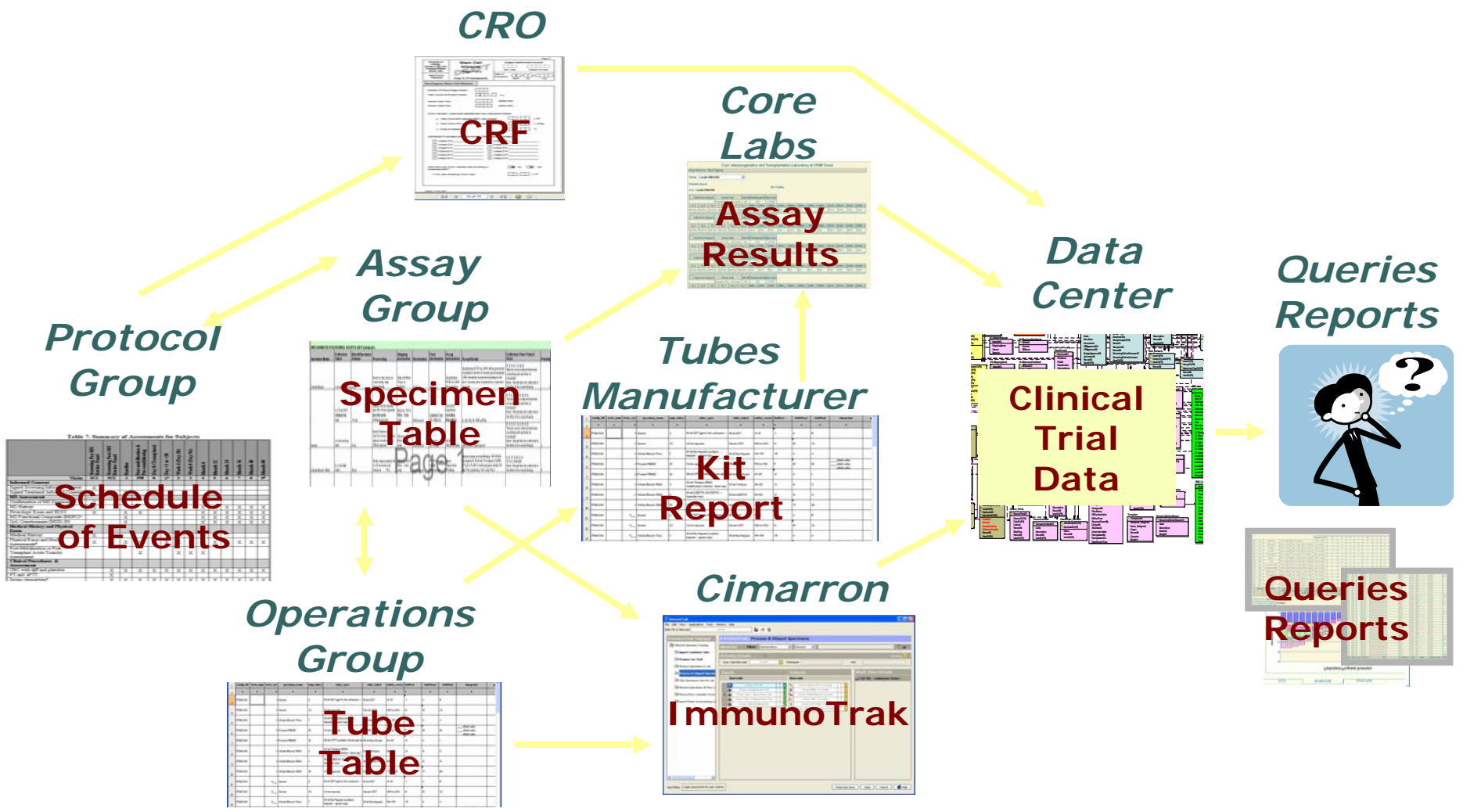
EPOCH: Immune Tolerance Network clinical trial ontologies



- Immune Tolerance Network (ITN)
 - International collaborative research effort that sponsors clinical trials and mechanistic assays on immune tolerance
- EPOCH clinical trial model
 - Developed at Stanford Medical Informatics
 - Designed to provide **semantic foundation for management of clinical trials**

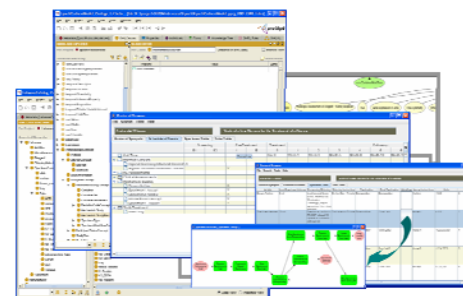


Management of clinical trials involves complex data and multiple groups



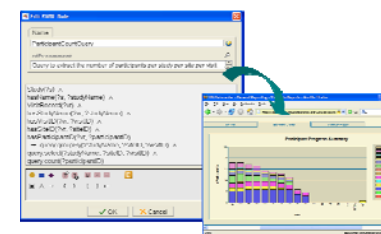
Three goals of EPOCH ontologies

- Design tools to help **acquire and maintain knowledge** about protocol and assay designs
- Use this knowledge to **drive data collection** during a trial
- Implement **querying methods** to support trial management, and ad hoc data analysis



2008-03-04 14:00:00

Name	Expression
KIT-pre-Ref	#Study(%_hasValue(%_)) -> hasKit(%_)
KIT-pre-Ref-ontology	#MLAbInMapping(%_)
KIT-pre-Ref-key	#MLAbInMapping(%_)
KIT-pre-Ref-value	#MLAbInMapping(%_)



EPOCH ontologies created in Protégé OWL

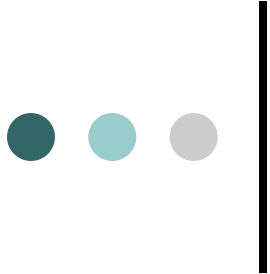
The screenshot shows the Protégé 3.3 beta interface. The main window displays the 'Protocol' class editor. The class name 'Protocol' is prominently displayed in a yellow box. Below it, a description reads: 'captures the protocol schema, the events that are planned and their schedule.

'. The 'Annotations' section shows 'primary' with a language dropdown set to 'Lang'. A separate window in the foreground lists the following properties for the 'Protocol' class:

- hasFacilitiesPlan (single FacilitiesPlan)
- hasLongTitle (single string)
- hasPrimaryScheduleOfEvents (single ScheduleOfEvents)
- hasProtocolId (single string)
- hasSchedulesOfEvents (multiple ScheduleOfEvents)
- hasSpecimenWorkflow (single SpecimenWorkflow)

The background interface includes a menu bar (File, Edit, Project, OWL, Code, Tools, Window, Help), a toolbar, and a class hierarchy tree on the left. The tree shows the following structure:

- org:Facility
- assay:Assay
- lab:LabWare
- lab:PreprocessingCondition
- lab:SpecimenMetaClass_16
- lab:StorageConditions
- ce:ConstraintExpression
- mo:Measurement
- mo:Unit
- PROTOCOL EVENT
- ProtocolEntity
 - FacilitiesPlanEntity
 - Protocol
 - ProtocolEventEntity
 - StudyDesignEntity
 - StudyScheduleEntity



ITN wants to use BRIDG-compliant applications

- ITN protocols encoded as **EPOCH knowledge bases to drive caBIG applications** (e.g., Patient Study Calendar)
- Challenge: Develop methods to
 - Harmonize common subset of BRIDG & EPOCH: **shared semantics**
 - Overcome **representational mismatch**
 - Representation languages
 - Representation choices
 - (Terminological mismatch not consider here)



Approach taken

- Semantic alignment
- Overcoming representation language mismatch
- Overcoming representation choice mismatches



Approach taken

- Semantic alignment
 - Use Excel spreadsheet to systematically review and document possible mappings
 - Define necessary preconditions for mapping
- Overcoming representation language mismatch
- Overcoming representation choice mismatches

Semantic Alignment: Excel spreadsheet

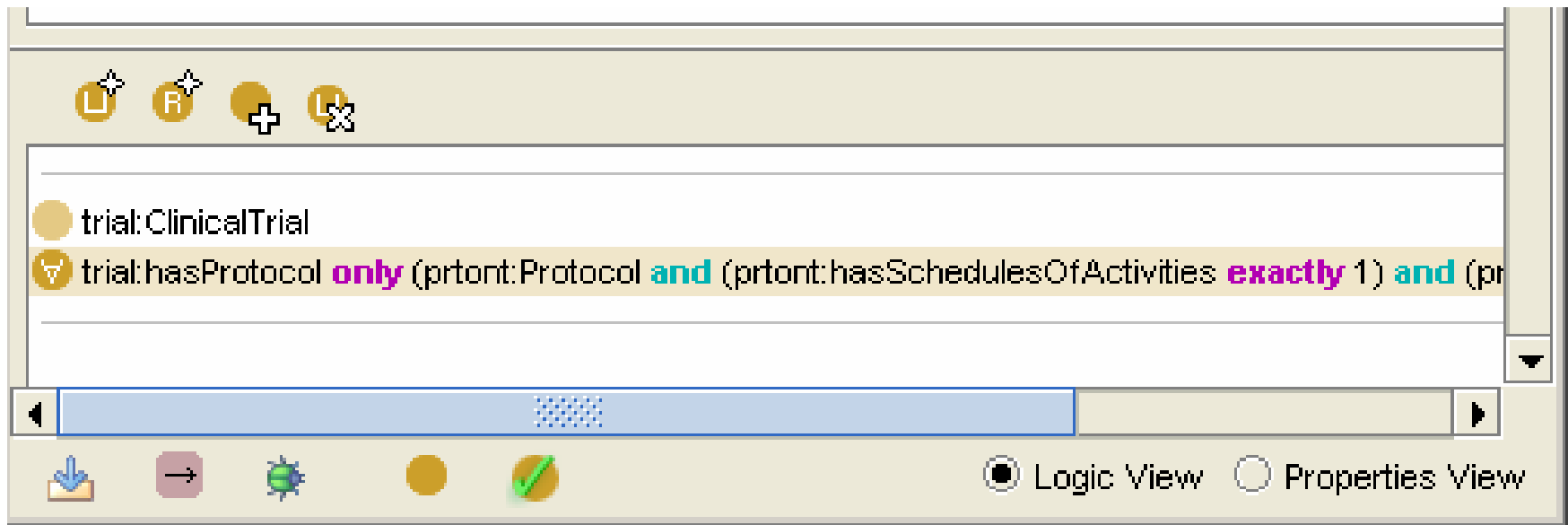
Class	Attribute	Attribute Value or Description	InGreenbaum?	In Scope?	Mappable?	
Period	ClassComment	The entire protocol is segmented temporally into periods. Each period defines a segment of time that groups protocol activities by their function. For example, the Screening Period defines the protocol timing during which the activities related to protocol	y	y	y	BRIDG.Plan
Period	hasName	name of the period	y	y	y	BRIDG.Plan
Period	hasPeriodType	a generic label assigned to the period	y	y	n	<<no type ir
Period	hasSubTimings	a set of sub-periods or encounters that this period is decomposed into	y	y	y	BRIDG.Plan
Period	isPartOfParentTimings	a set of arms that this period is part of	y	y	y	BRIDG.Plan
Period	SubClassOf	#PlannedTiming	y	y	e	epoch-spec
PlannedActivity	ClassComment	Protocol-related activity such as a clinical assessment, a treatment or a mechanistic study, which involves participants and study site personnel.	y	y	y	BRIDG.Plan
PlannedActivity	hasAnnotations	annotations that specify instructional or temporal information on the activity	y	y	y	BRIDG.Plan will cover in information between an BRIDG.Cale
PlannedActivity	hasFunctionalGrouping	a higher-level grouping that the activity belongs to	y	y	y	BRIDG.Cont (allows for c collecting c PlannedAct
PlannedActivity	hasName	name of the activity	y	y	y	BRIDG.Plan in the BRID!



Semantic Alignment: Restrictions on EPOCH

- Mapping from EPOCH to BRDG => Place restrictions on EPOCH
 - Only one schedule of activities
 - Period has no subperiods
 - Limited temporal annotations
 - ...
- Define necessary conditions for mapping
 - Formulate as DL definition of “BRIDGClinicalTrial” subclass of epoch:ClinicalTrial

BRIDGClinicalTrial in EPOCH



- Need trial-specific closure axioms to do automated classification of EPOCH trials that can be mapped to BRIDG

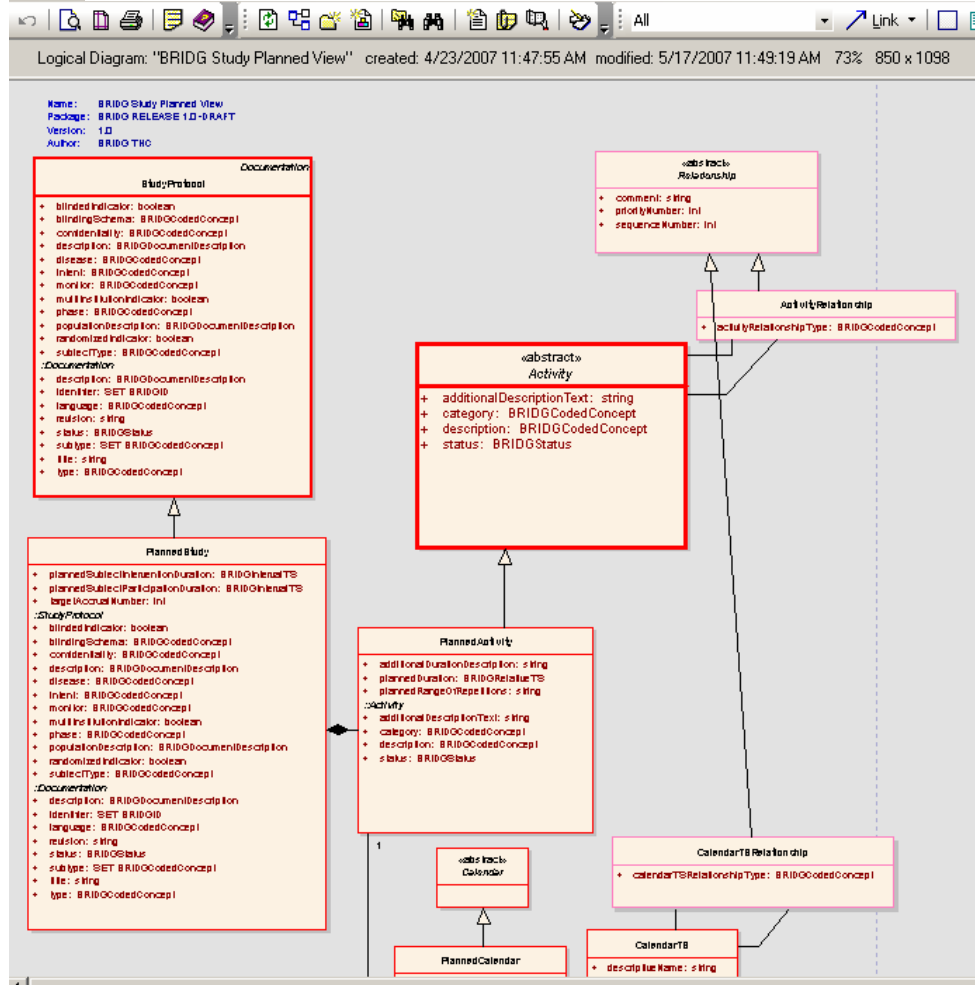


Approach taken

- Semantic alignment
- Overcoming representation language mismatch
- Overcoming representation choice mismatches

Overcoming representation language mismatch: BRIDG-in-OWL

Scope: BRIDG Study Planned View + BRIDG Complex Data Types



BRIDG Protégé 3.3 beta (file:D:_CurrentWork\ITN\Epoch2BRIDG\BRIDG.ppr, OW...)

File Edit Project OWL Code Tools Window Help

Properties Individuals Forms SWRL Rules

Metadata (BRIDG.owl) OWLClasses

SUBCLASS EXPLORER For Project: BRIDG

CLASS EDITOR For Class: (Instance of owl:Class) Inferred View

Property

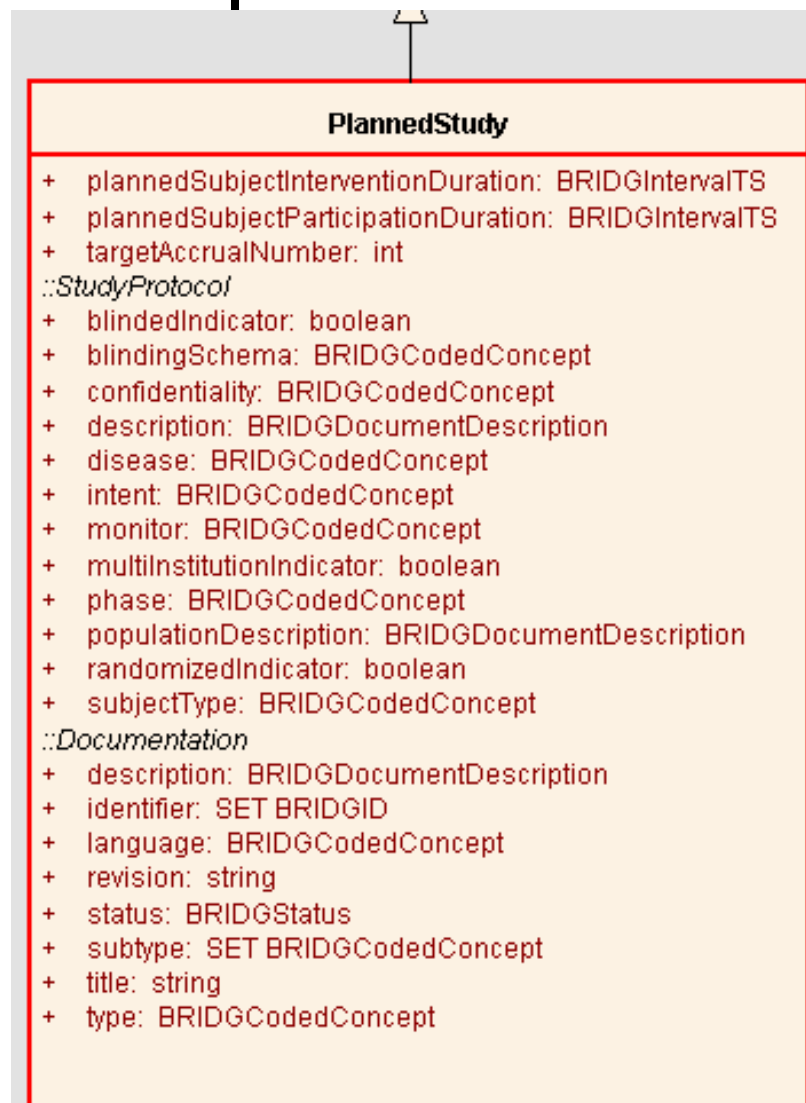
Property List:

- hasArms (multiple Arm)
- hasEpochs (multiple Epoch)
- hasPlannedActivities (multiple PlannedActivity)
- hasPlannedCalendar (single PlannedCalendar)
- hasPlannedCalendarTCRelationships (multiple CalendarTSRelationship)
- documentDescription (single BRIDGDocumentDescription)
- hasStudySites (multiple FunctionalRole)

Asserted Hierarchy:

- owl:Thing
 - rdf:List
 - BRIDGConcept
 - Activity
 - CalendarCell
 - CalendarTC
 - FunctionalRole
 - Organization
 - OrganizationRole
 - PlannedCalendar
 - Relationship
 - StudyProtocol
 - PlannedStudy
 - Data Type
 - BRIDG CodedConcept
 - BRIDGDocumentDescription
 - BRIDGID
 - BRIDGPhysicalQuantity
 - EnumeratedValues
 - IVLOfDuration
 - Range
 - SetOperator

Example: PlannedStudy



PlannedStudy (instance of owl:Class)

CLASS EDITOR

For Class: PlannedStudy (instance of owl:Class) Inferred View

Property	Value
<input checked="" type="checkbox"/> rdfs:comment	

Properties

- hasArms (multiple Arm)
- hasEpochs (multiple Epoch)
- hasPlannedActivities (multiple PlannedActivity)
- hasPlannedCalendar (single PlannedCalendar)
- hasPlannedCalendarTCRelationships (multiple CalendarTCRelationship)
- documentDescription (single BRIDGDocumentDescription)
- hasStudySites (multiple FunctionalRole)
- identifier (multiple BRIDGID)
- title (single string)

Superclasses

- StudyProtocol

Logic View Properties View



Modifications to BRIDG

- Driven by Patient Study Calendar application requirements
- Added several associational relationships
- Modified some subsumption relations



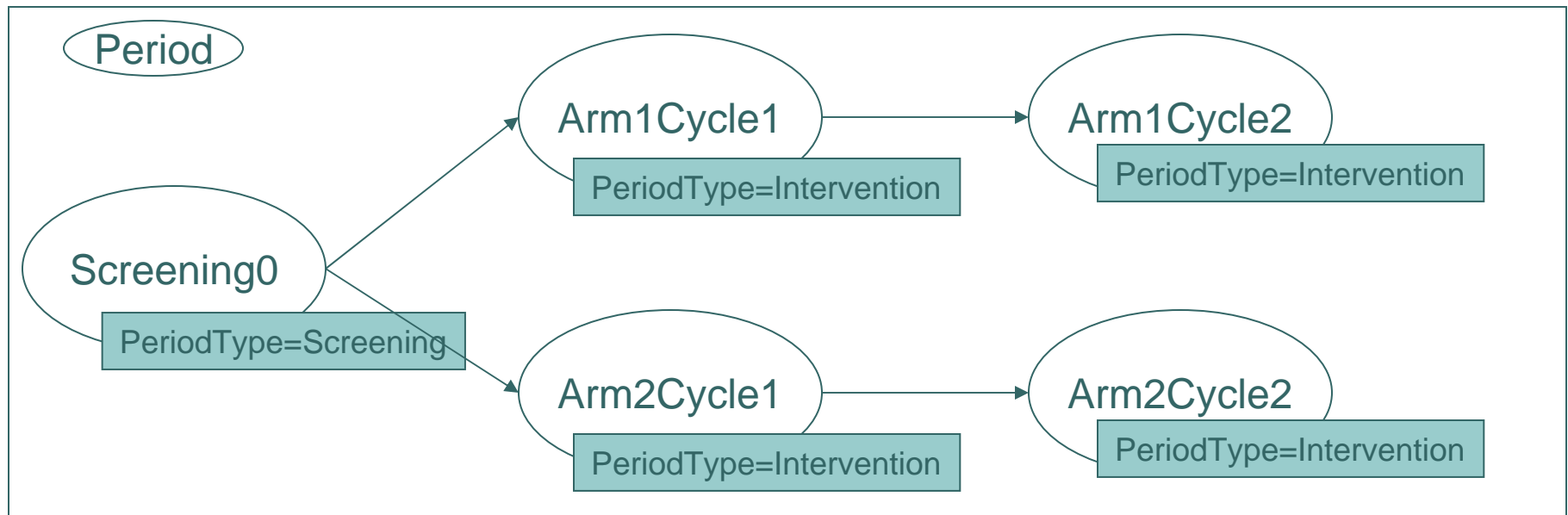
Approach taken

- Semantic alignment
- Overcoming representation language mismatch
- Overcoming representation choice mismatches

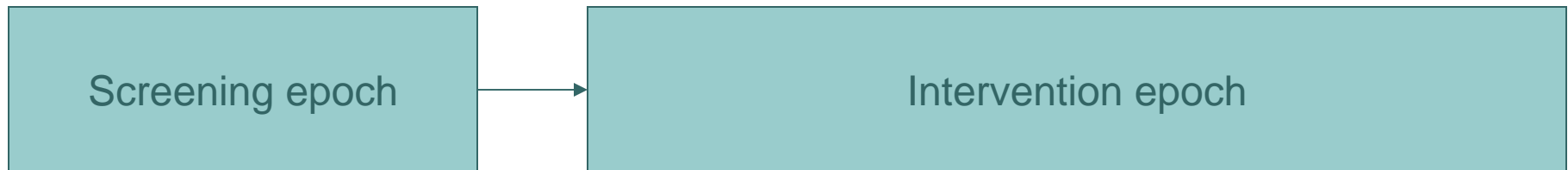


Overcoming representation choice mismatch: *Epoch* example

EPOCH



BRIDG





SWRL rule to map epochs

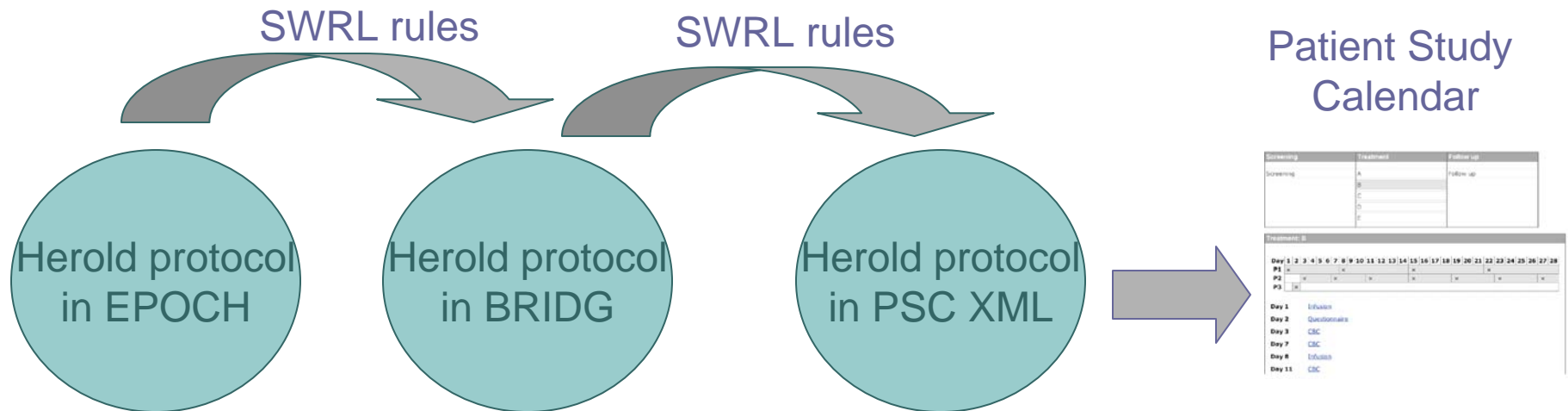
SWRL Rule

```
prontont:Period(?p) ∧  
prontont:hasPeriodType(?p, ?ptype) ∧  
prontont:label(?ptype, ?epochLabel) ∧  
swrlx:createOWLThing(?epoch, ?ptype) ∧  
swrlx:createOWLThing(?code, ?epoch) ∧  
bridg:PlannedStudy(?pstudy)  
  → bridg:Epoch(?epoch) ∧  
bridg:BRIDGCodedConcept(?code) ∧  
bridg:epochName(?epoch, ?code) ∧  
bridg:displayName(?code, ?epochLabel) ∧  
bridg:descriptiveName(?epoch, ?epochLabel) ∧  
bridg:hasEpochs(?pstudy, ?epoch)
```

- EPOCH:periodTypes of periods correspond to BRIDG:epochs
- EPOCH :periodType.label corresponds to BRIDG:epoch.code.displayName



Successfully used an EPOCH clinical trial to configure BRIDG Patient Study Calendar application



- Automated mappings except for one relationship
 - Because of OWL/SWRL's open-world assumption, *First epoch* cannot be derived as *an epoch that has no predecessor*



Conclusions

- Semantic interoperability requires
 - Harmonization of subsets of ontologies/models
 - Overcoming mismatches in representation languages and representation choices
- OWL restrictions and SWRL rules help to overcome semantic and syntactic mismatches
- Possible future work
 - Continued harmonization of BRIDG/EPOCH
 - Scalability and (semi-)automation of method



Thank you!

- Questions?
- Comments?
- Suggestions?