Ontologies

6.871: Lecture 22

What is an Ontology?

- A formal, explicit specification of a shared conceptualization.
- A shared vocabulary that can be used to model a domain, i.e., the objects and/or concepts that exist, their properties and relations
- Imposition of specific set of conceptualizations on a domain of interest
  - Tell me about analog electronics
  - Tell me about digital electronics
- Definitions of terminology
  - and constraints between terms
- Domain mini-theories

Ontology vs KB?

- Can think of an ontology as a kind of KB, but:

- Ontology serves different purpose:
  - Only needs to describe vocabulary, axioms
  - E.g. database schema is an ontology

- KB includes:
  - Specific knowledge needed for problem-solving

Motivations

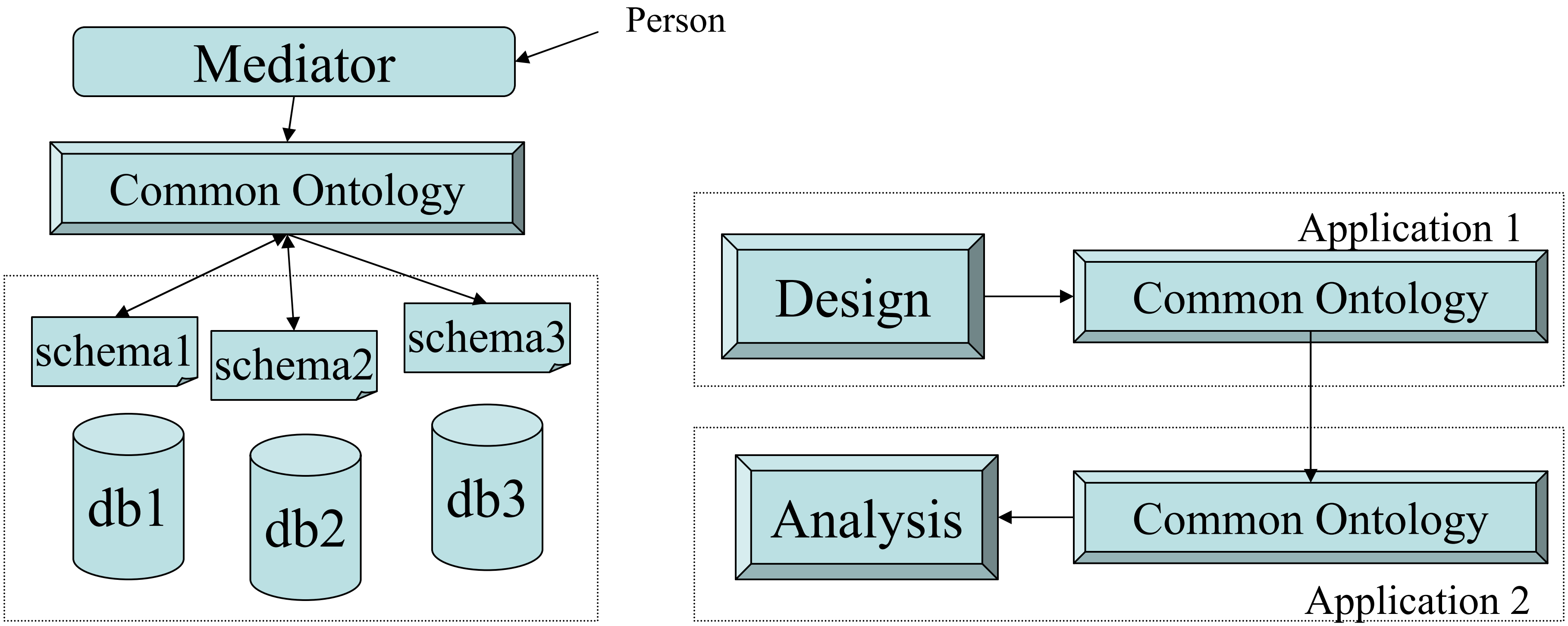
- Engineering motivation:
  - Every knowledge-based system is based on an ontology of its domain
  - Explication of the ontology is a time-consuming component of the development process
  - Why not amortize the effort and share ontologies?
    - E.g. “core ontologies” such as space, time, quantities

- Scientific motivation:
  - Understand fundamental issues about human conceptualizations

Pragmatic Motivations

- Responding to the unexpected
- Distributed Databases
- Distributed Applications
- Communicating Agents
- Semantic Web

Key Question: What does he mean when he says <…>?



Aspects of an Ontology

- Content
- Form
- Purpose
- Development

Aspects of an Ontology

- Content
  - types of objects, relationships
  - e.g. the blocks world conceptualization includes:
    - Object Classes: Blocks, Robot Hands
    - Properties: shapes of blocks, color of blocks
    - Relationships: On, Above, Below, Grasp
    - Processes: stacking plan for a tower
- Form
- Purpose
- Development

Aspects of an Ontology

- Content
- Form
  - Is the taxonomic relationship (instance-of, subclass) primary?
  - Are definitions of, or constraints on, terms provided?
  - Is the definitional language as rich as a full logic?
  - Is it process-centric or object-centric?
- Purpose
- Development

Aspects of an Ontology

- Content
- Form
- Purpose
  - Knowledge sharing
    - E.g. Between people, software systems, agents
  - Knowledge reuse
    - E.g. When models or systems change
  - General (common sense) or domain specific
- Development

Aspects of an Ontology

- Content
- Form
- Purpose
- Development
  - Is it acquired or engineered?
  - If acquired, what about:
    - Quality of knowledge
    - Diversity of content
    - Trust in knowledge
    - Unpredictable use

Building an Ontology

- Planning
- Specification - consider scope and purpose
- Knowledge Acquisition
- Conceptualization - glossary of terms, top-down, bottom-up, middle-out
- Integration - of existing relevant ontologies
- Implementation
- Evaluation - Clarity, Coherence, Extensibility, Minimal Encoding Bias, Minimal Ontological Commitment

Example Ontologies

see http://www.cs.utexas.edu/users/mfkb/related.html

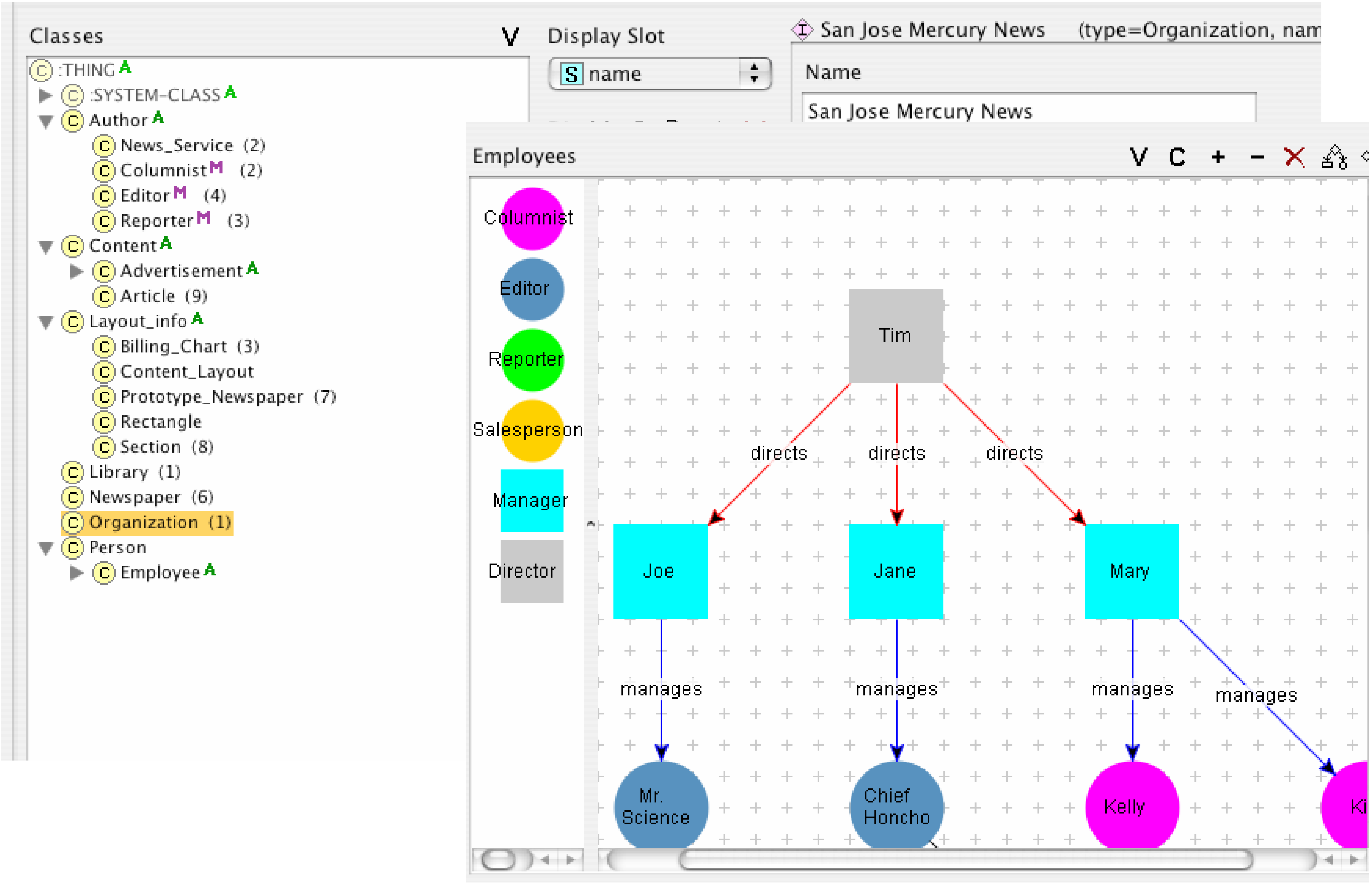
* ARPI Planning and Scheduling ontologies
* Aviation Ontology
* BPMO - The Business Process Management Ontology
* CYC (and the derivative PDKB)
* DOLCE - a Descriptive Ontology for Linguistic and Cognitive Engineering.
* Dublin Core (bibliographic organization)
* The Enterprise Ontology (for business enterprises)
* Ontologies for ethology (animal behavior), e.g. Loggerhead Turtle
* FrameNet (lexical reference)
* Generalized Upper Model (for NLP)
* Mikrokosmos (for NLP)
* ON9 (the CNR-ITBM Ontology Library)
* OWL-S - The OWL (formerly DAML) Services ontology.
* Ontolingua Ontology Library
* OpenMind database and OM CSNet Semantic Network
* PharmGKB - Pharmacogenetics and Pharmacogenomics Knowledge Base
* PSL (process specification)
* QoS (computers and networks)
* SENSUS (for NLP)
* STEP (for product data exchange)
* SUMO (the Suggested Upper Merged Ontology)
* the Twente Ontology Collection
* UMLS (biomedicine)
* Wilkins' ontology (17th century!)
* WordNet (lexical reference)

Example Tools for Ontologies

see http://www.cs.utexas.edu/users/mfkb/related.html,
http://www.xml.com/pub/a/2002/11/06/ontologies.html

* Chimaera
* CODE4
* Generic Knowledge-Base Editor
* Ikarus
* JOE (Java Ontology Editor)
* KAON
* KACTUS
* OilEd
* OntoEdit
* Ontosaurus
* Protege
* Snobase
* Stanford Ontology Editor
* SymOntos
* Word Map

Protégé



http://protege.stanford.edu/

Some Large Ontologies

CYC common sense	10^5 concept types, 10^6 axioms	CYCL	Partially Online: 6000 Top Concepts
SUMO upper ontology	1000 terms, 4200 assertions	KIF Also LOOM, OWL,Protege	Published Online
WordNet lexical memory	152,059 word forms in 115,424 synsets	Semantic Network	Published Online
Sensus text understanding	70,000 terms extension of WordNet	Semantic Network	Published Online
UMLS biomedicine	135 Semantic Types, 54 semantic relations, 975,354 concepts	Semantic Network	Published Online

UMLS

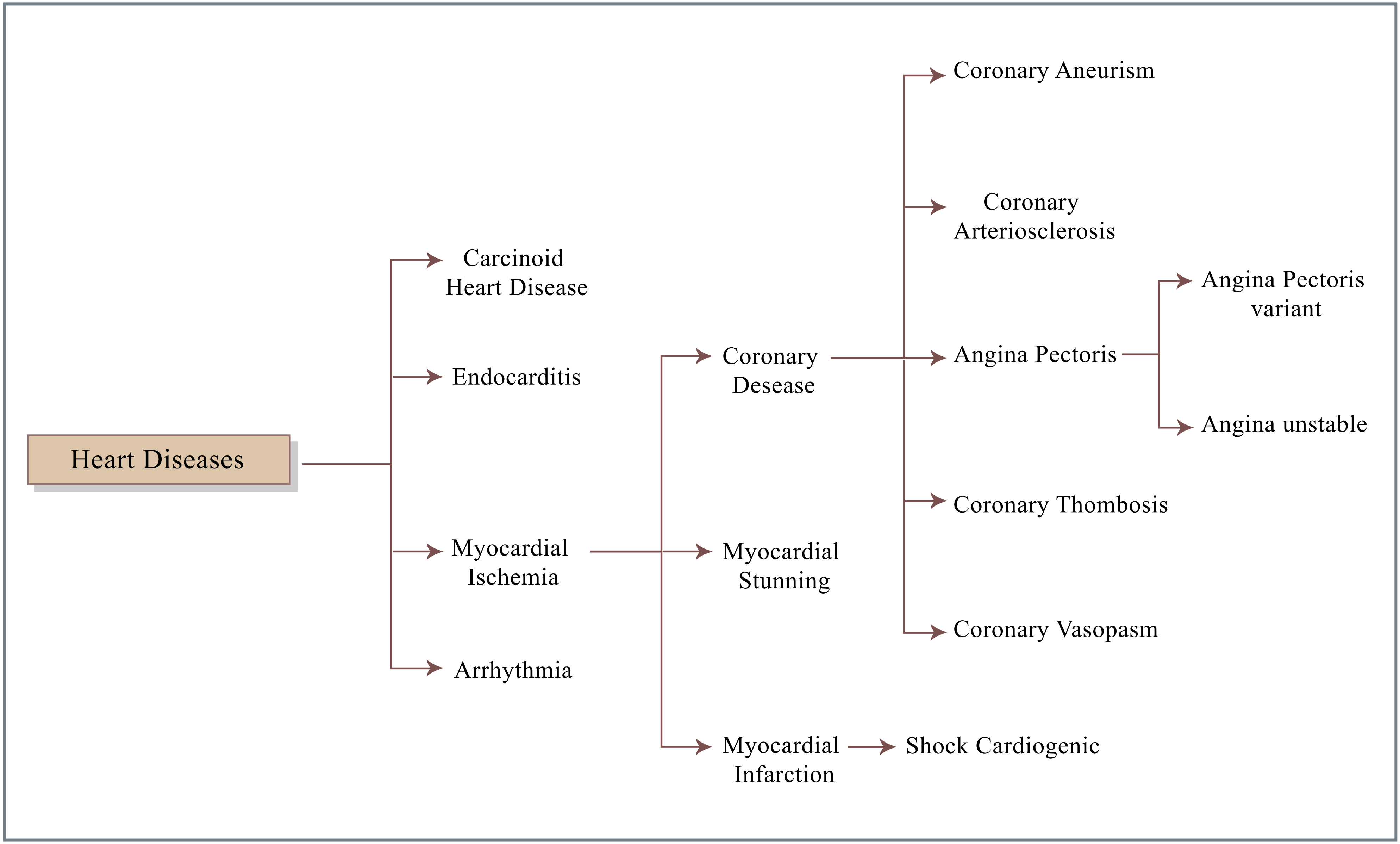
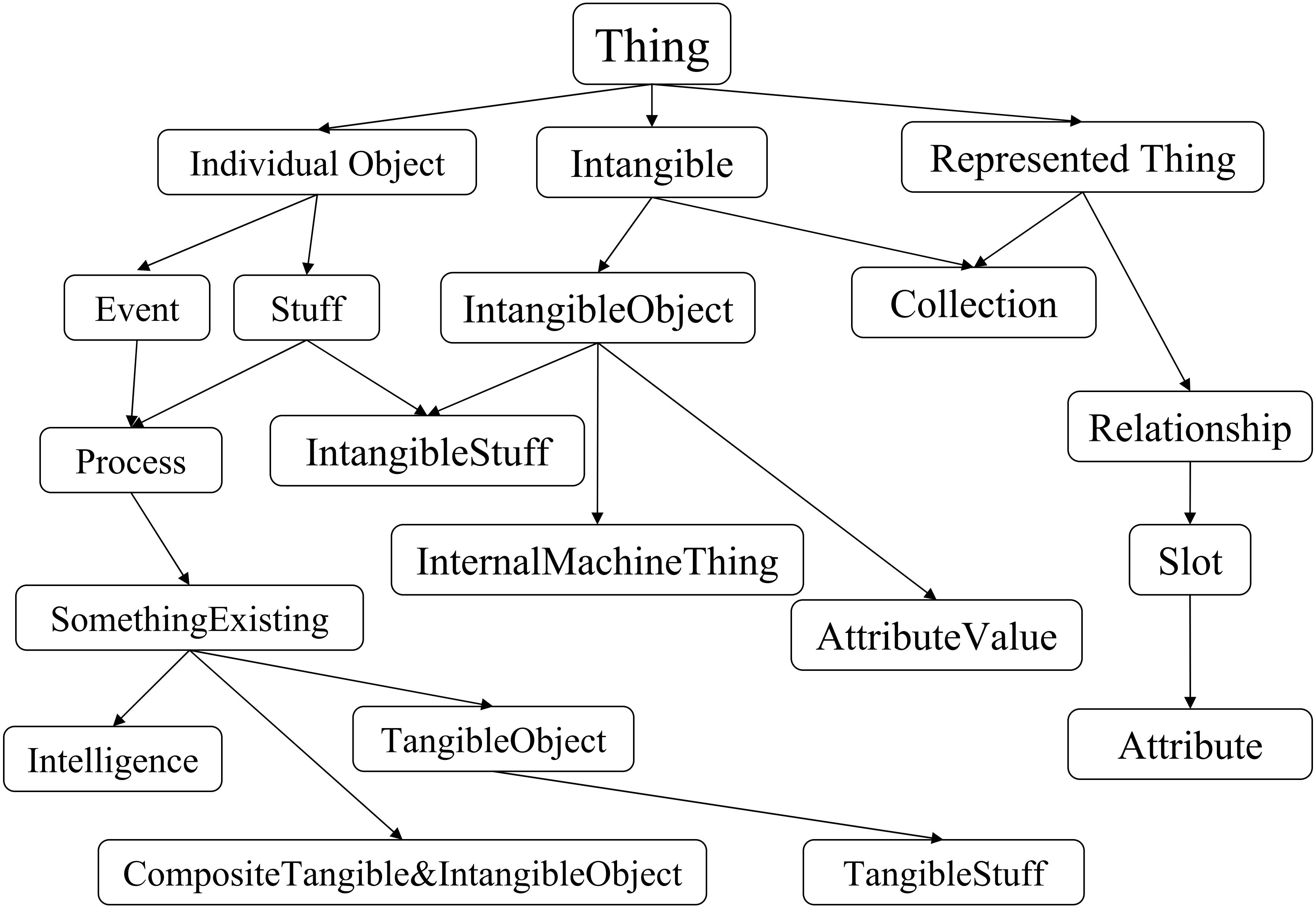


Figure by MIT OCW.

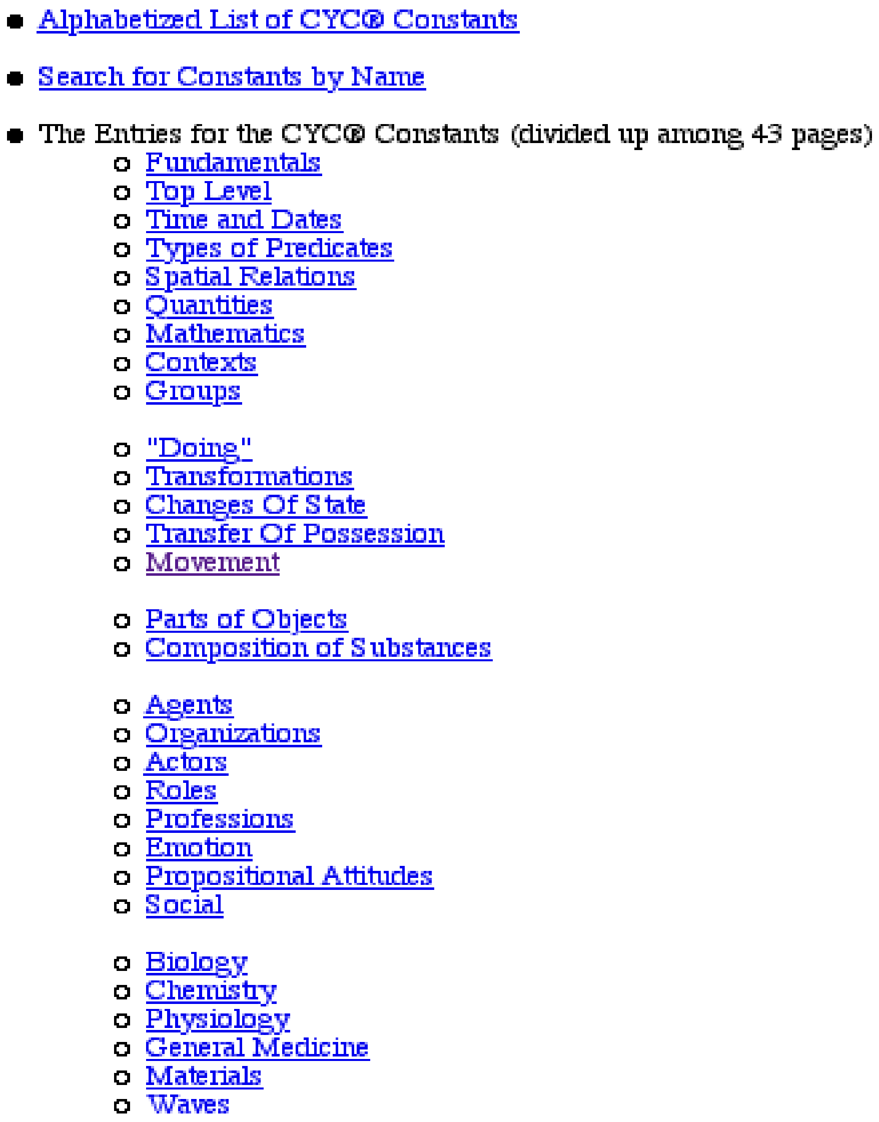
CYC

- Goal: Encode all of human common sense knowledge
- Mechanization: human-entered axioms
- Periodic review, reorganization, compaction, separation into distinct mini-theories, not mutually consistent
- Driven by application domains
- Often seems ad-hoc

CYC Top Level Categories



CYC Ontology



- Alphabetized List of CYC® Constants
- Search for Constants by Name
- The Entries for the CYC® Constants (divided up among 43 pages)
    - Fundamentals
    - Top Level
    - Time and Dates
    - Types of Predicates
    - Spatial Relations
    - Quantities
    - Mathematics
    - Contexts
    - Groups

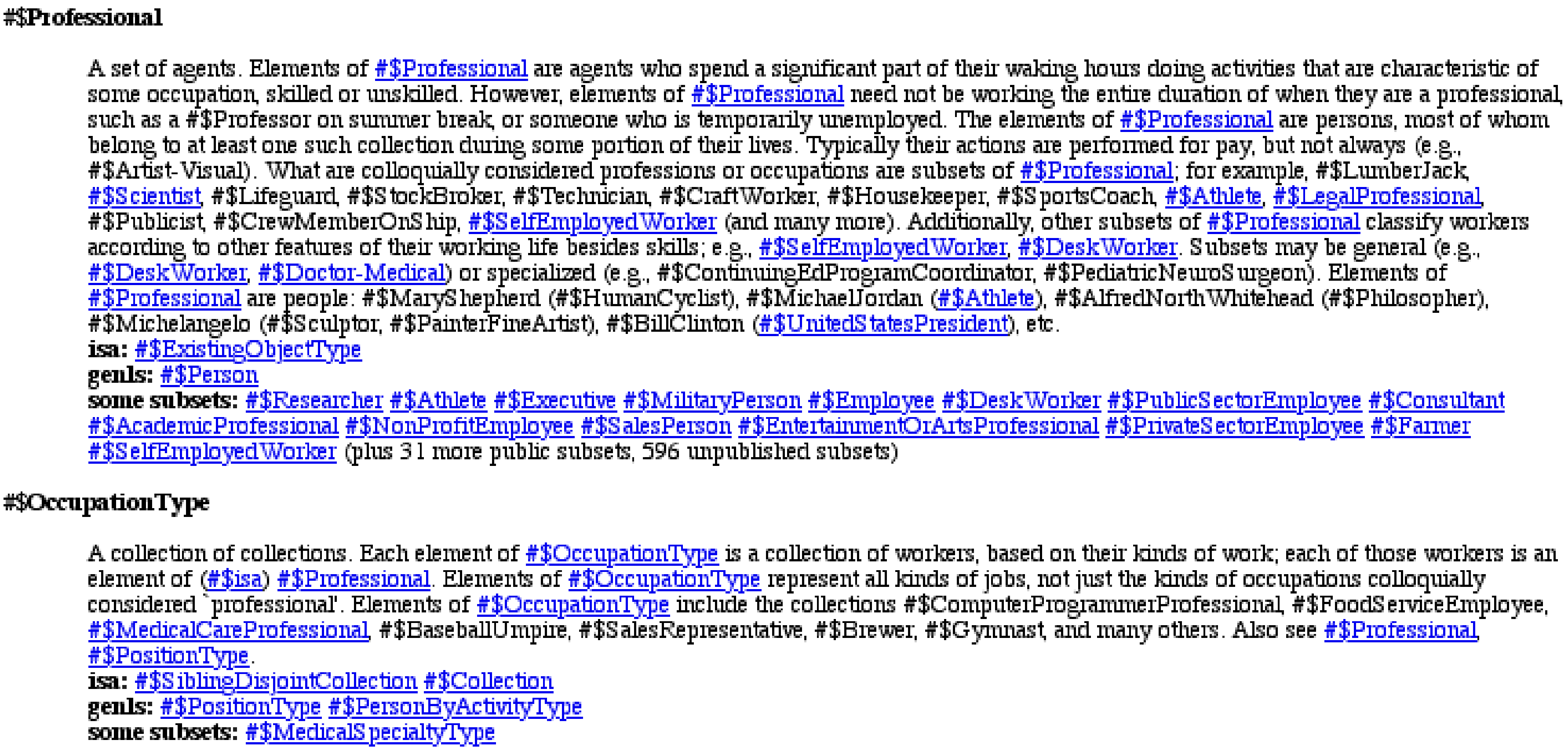
    - "Doing"
    - Transformations
    - Changes Of State
    - Transfer Of Possession
    - Movement

    - Parts of Objects
    - Composition of Substances

    - Agents
    - Organizations
    - Actors
    - Roles
    - Professions
    - Emotion
    - Propositional Attitudes
    - Social

    - Biology
    - Chemistry
    - Physiology
    - General Medicine
    - Materials
    - Waves

CYC Examples



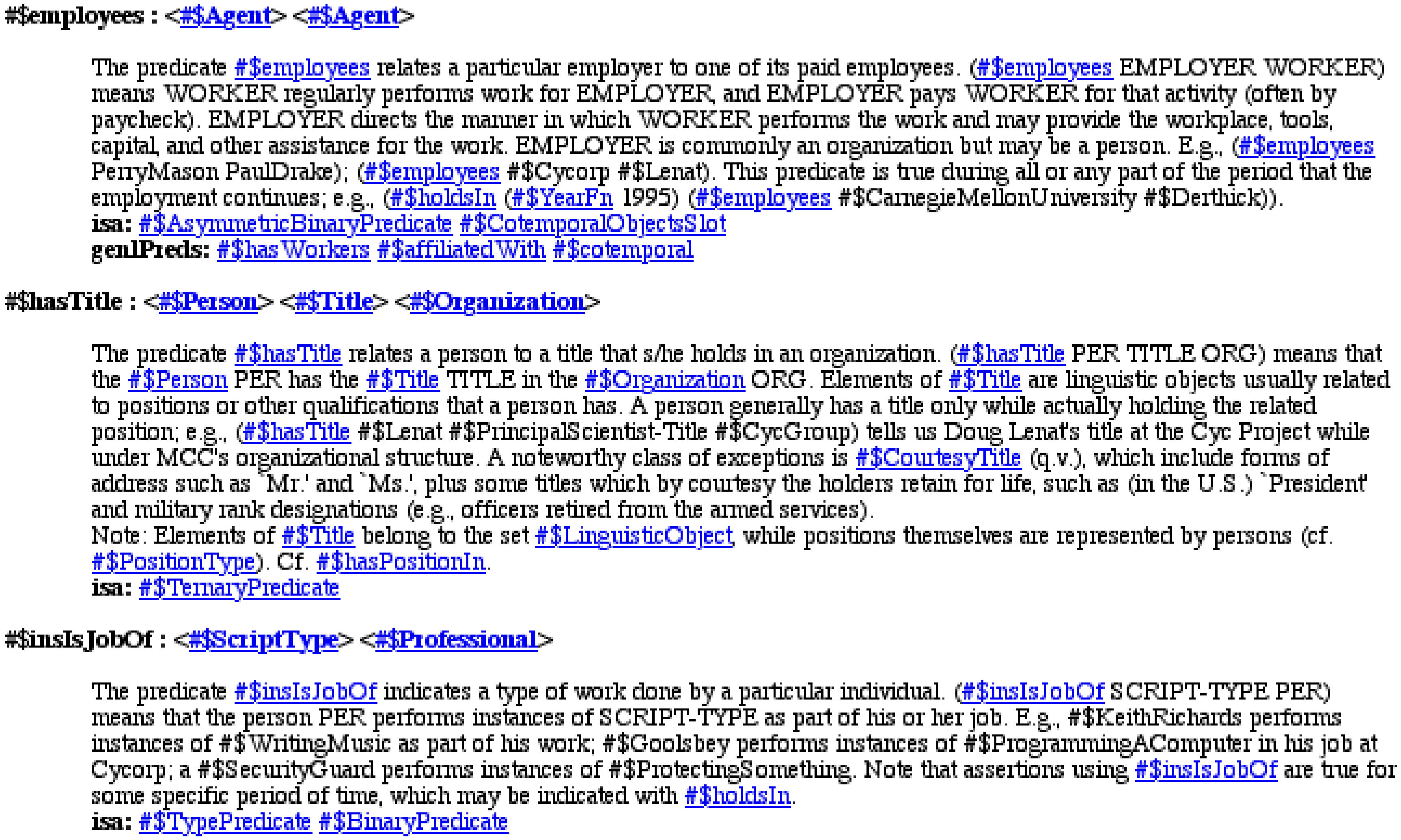
#$Professional

A set of agents. Elements of #$Professional are agents who spend a significant part of their waking hours doing activities that are characteristic of some occupation, skilled or unskilled. However, elements of #$Professional need not be working the entire duration of when they are a professional, such as a #$Professor on summer break, or someone who is temporarily unemployed. The elements of #$Professional are persons, most of whom belong to at least one such collection during some portion of their lives. Typically their actions are performed for pay, but not always (e.g., #$Artist-Visual). What are colloquially considered professions or occupations are subsets of #$Professional; for example, #$LumberJack, #$Scientist, #$Lifeguard, #$StockBroker, #$Technician, #$CraftWorker, #$Housekeeper, #$SportsCoach, #$Athlete, #$LegalProfessional, #$Publicist, #$CrewMemberOnShip, #$SelfEmployedWorker (and many more). Additionally, other subsets of #$Professional classify workers according to other features of their working life besides skills; e.g., #$SelfEmployedWorker, #$DeskWorker. Subsets may be general (e.g., #$DeskWorker, #$Doctor-Medical) or specialized (e.g., #$ContinuingEdProgramCoordinator, #$PediatricNeuroSurgeon). Elements of #$Professional are people: #$MaryShepherd (#$HumanCyclist), #$MichaelJordan (#$Athlete), #$AlfredNorthWhitehead (#$Philosopher), #$Michelangelo (#$Sculptor, #$PainterFineArtist), #$BillClinton (#$UnitedStatesPresident), etc.
isa: #$ExistingObjectType
genls: #$Person
some subsets: #$Researcher #$Athlete #$Executive #$MilitaryPerson #$Employee #$DeskWorker #$PublicSectorEmployee #$Consultant #$AcademicProfessional #$NonProfitEmployee #$SalesPerson #$EntertainmentOrArtsProfessional #$PrivateSectorEmployee #$Farmer #$SelfEmployedWorker (plus 31 more public subsets, 596 unpublished subsets)

#$OccupationType

A collection of collections. Each element of #$OccupationType is a collection of workers, based on their kinds of work; each of those workers is an element of (#$isa) #$Professional. Elements of #$OccupationType represent all kinds of jobs, not just the kinds of occupations colloquially considered `professional'. Elements of #$OccupationType include the collections #$ComputerProgrammerProfessional, #$FoodServiceEmployee, #$MedicalCareProfessional, #$BaseballUmpire, #$SalesRepresentative, #$Brewer, #$Gymnast, and many others. Also see #$Professional, #$PositionType.
isa: #$SiblingDisjointCollection #$Collection
genls: #$PositionType #$PersonByActivityType
some subsets: #$MedicalSpecialtyType

CYC Examples (cont’d)



#$employees : <#$Agent> <#$Agent>

The predicate #$employees relates a particular employer to one of its paid employees. (#$employees EMPLOYER WORKER) means WORKER regularly performs work for EMPLOYER, and EMPLOYER pays WORKER for that activity (often by paycheck). EMPLOYER directs the manner in which WORKER performs the work and may provide the workplace, tools, capital, and other assistance for the work. EMPLOYER is commonly an organization but may be a person. E.g., (#$employees PerryMason PaulDrake); (#$employees #$Cycorp #$Lenat). This predicate is true during all or any part of the period that the employment continues; e.g., (#$holdsIn (#$YearFn 1995) (#$employees #$CarnegieMellonUniversity #$Derthick)).
isa: #$AsymmetricBinaryPredicate #$CotemporalObjectsSlot
genlPreds: #$hasWorkers #$affiliatedWith #$cotemporal

#$hasTitle : <#$Person> <#$Title> <#$Organization>

The predicate #$hasTitle relates a person to a title that s/he holds in an organization. (#$hasTitle PER TITLE ORG) means that the #$Person PER has the #$Title TITLE in the #$Organization ORG. Elements of #$Title are linguistic objects usually related to positions or other qualifications that a person has. A person generally has a title only while actually holding the related position; e.g., (#$hasTitle #$Lenat #$PrincipalScientist-Title #$CycGroup) tells us Doug Lenat's title at the Cyc Project while under MCC's organizational structure. A noteworthy class of exceptions is #$CourtesyTitle (q.v.), which include forms of address such as `Mr.' and `Ms.', plus some titles which by courtesy the holders retain for life, such as (in the U.S.) `President' and military rank designations (e.g., officers retired from the armed services).
Note: Elements of #$Title belong to the set #$LinguisticObject, while positions themselves are represented by persons (cf. #$PositionType). Cf. #$hasPositionIn.
isa: #$TernaryPredicate

#$insIsJobOf : <#$ScriptType> <#$Professional>

The predicate #$insIsJobOf indicates a type of work done by a particular individual. (#$insIsJobOf SCRIPT-TYPE PER) means that the person PER performs instances of SCRIPT-TYPE as part of his or her job. E.g., #$KeithRichards performs instances of #$WritingMusic as part of his work; #$Goolsbey performs instances of #$ProgrammingAComputer in his job at Cycorp; a #$SecurityGuard performs instances of #$ProtectingSomething. Note that assertions using #$insIsJobOf are true for some specific period of time, which may be indicated with #$holdsIn.
isa: #$TypePredicate #$BinaryPredicate

Perspectives

- Philosophy
- Library and Information Science
- Natural Language Processing
- Artificial Intelligence
- Semantic Web

Perspectives

- Philosophy
  - Objectives: Classify and categorize the world
  - E.g.: Aristotle …
- Library and Information Science
- Natural Language Processing
- Artificial Intelligence
- Semantic Web

Perspectives

- Philosophy
- Library and Information Science
  - Objectives: organize bibliographic world, model universal and domain knowledge
  - Usage: provide access points to bibliographic entities
  - E.g., MARC; LCC, UDC, SAB
- Natural Language Processing
- Artificial Intelligence
- Semantic Web

Perspectives

- Philosophy
- Library and Information Science
- Natural Language Processing
  - Objectives: Model lexical and domain knowledge
  - Usage: Machine Translation, Information Extraction, Q/A
  - E.g.: Wordnet, Sensus, Generalised Upper Model
- Artificial Intelligence
- Semantic Web

Perspectives

- Philosophy
- Library and Information Science
- Natural Language Processing
- Artificial Intelligence
  - Objectives: Model common sense and domain knowledge
  - Usage: Knowledge representation and reasoning
  - E.g.: OpenMind, CYC; UMLS, ...
- Semantic Web

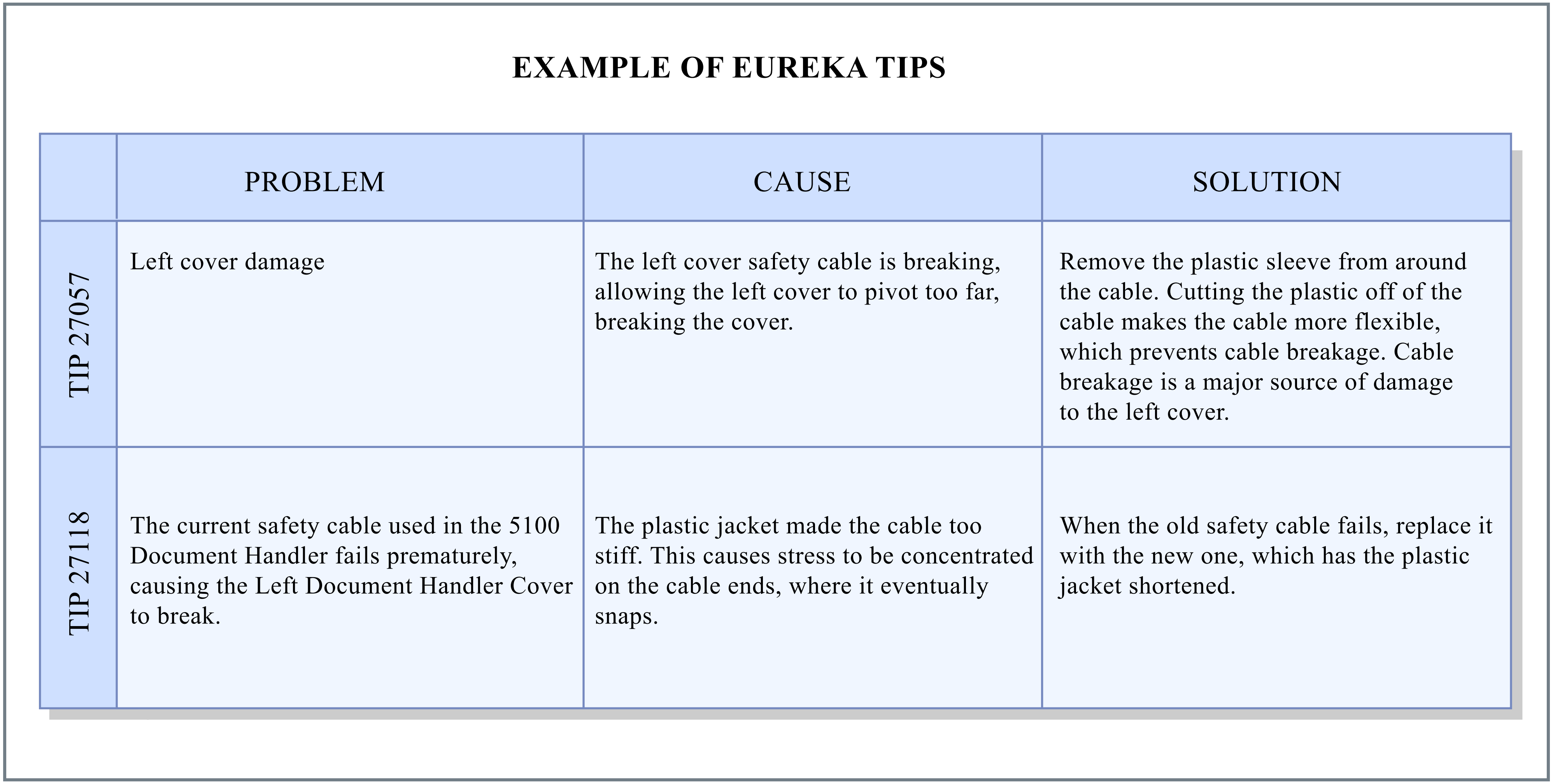
Perspectives

- Philosophy
- Library and Information Science
- Natural Language Processing
- Artificial Intelligence
- Semantic Web
  - Objectives: Provide semantics for web resources
  - Usage: Describe resources and their contents

Application Example

Document comparison (Xerox; Everett, et al. CACM, Feb 02)

- – Goal: Identify similar documents
- – Have: 40,000 technician-authored tips for copier repair



EXAMPLE OF EUREKA TIPS

	PROBLEM	CAUSE	SOLUTION
TIP 27057	Left cover damage	The left cover safety cable is breaking, allowing the left cover to pivot too far, breaking the cover.	Remove the plastic sleeve from around the cable. Cutting the plastic off of the cable makes the cable more flexible, which prevents cable breakage. Cable breakage is a major source of damage to the left cover.
TIP 27118	The current safety cable used in the 5100 Document Handler fails prematurely, causing the Left Document Handler Cover to break.	The plastic jacket made the cable too stiff. This causes stress to be concentrated on the cable ends, where it eventually snaps.	When the old safety cable fails, replace it with the new one, which has the plastic jacket shortened.

Figure by MIT OCW.

EXAMPLE OF EUREKA TIPS

	PROBLEM	CAUSE	SOLUTION
TIP 27057	Left cover damage	The left cover safety cable is breaking, allowing the left cover to pivot too far, breaking the cover.	Remove the plastic sleeve from around the cable. Cutting the plastic off of the cable makes the cable more flexible, which prevents cable breakage. Cable breakage is a major source of damage to the left cover.
TIP 27118	The current safety cable used in the 5100 Document Handler fails prematurely, causing the Left Document Handler Cover to break.	The plastic jacket made the cable too stiff. This causes stress to be concentrated on the cable ends, where it eventually snaps.	When the old safety cable fails, replace it with the new one, which has the plastic jacket shortened.

Figure by MIT OCW.

Natural language requires a huge ontology…

(1) The left cover broke in half.

(2) The sheet of paper breaks the light beam.

(3) Before doing step 3, you might want to break for coffee.

Natural language requires a huge ontology…

(1) The left cover broke in half. BreakDamage

(2) The sheet of paper breaks the light beam. BreakInterrupt

(3) Before doing step 3, you might want to break for coffee. BreakRecuperate

Natural language requires a more abstract one.

(1) The left cover broke in half. BreakDamage

(2) The sheet of paper breaks the light beam. BreakInterrupt

(3) Before doing step 3, you might want to break for coffee. BreakRecuperate

Map to concepts instead.

A direct word-to-concept ontology for the concept Incapacitate.

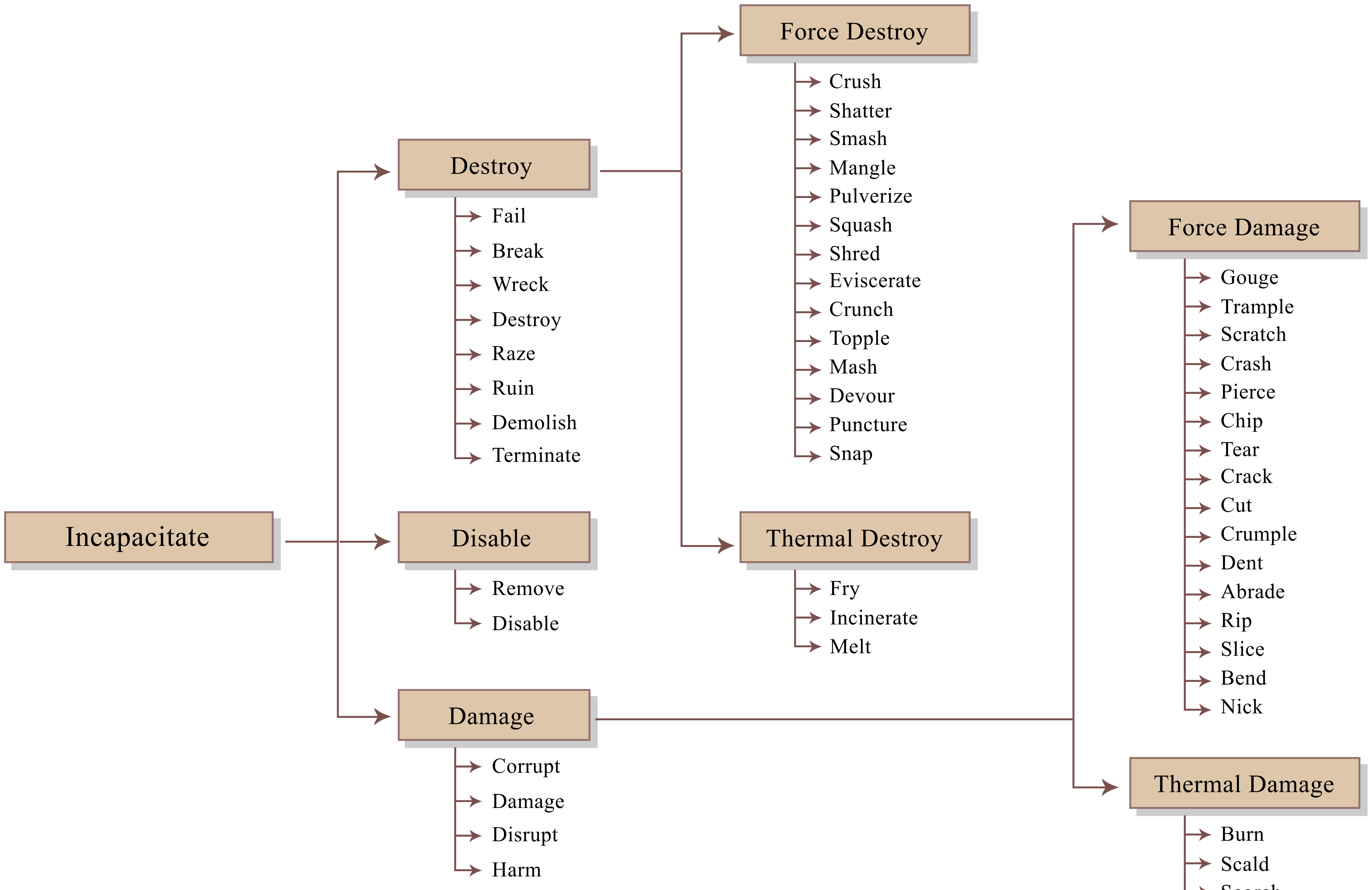
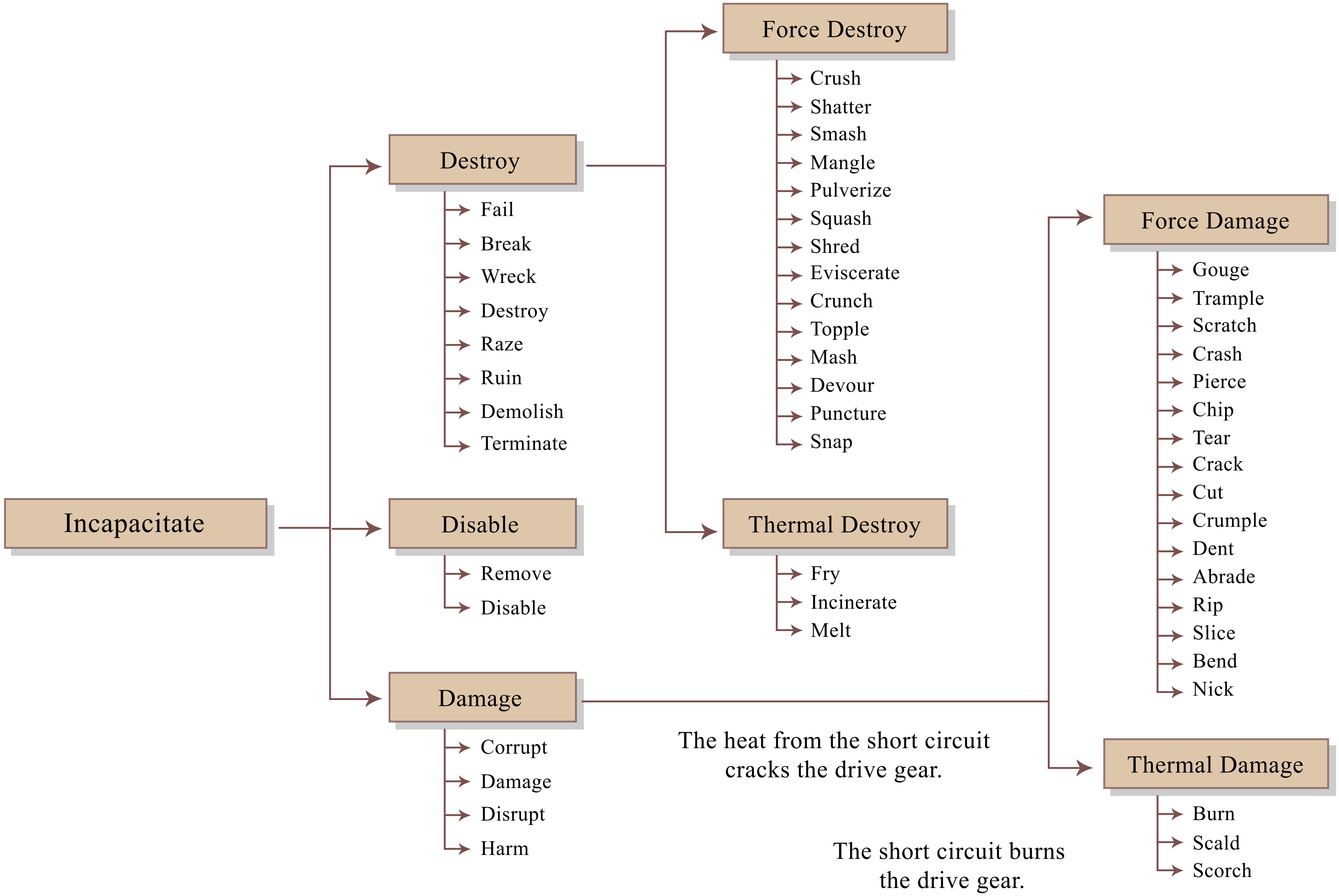


Figure by MIT OCW.

A direct word-to-concept ontology for the concept Incapacitate.

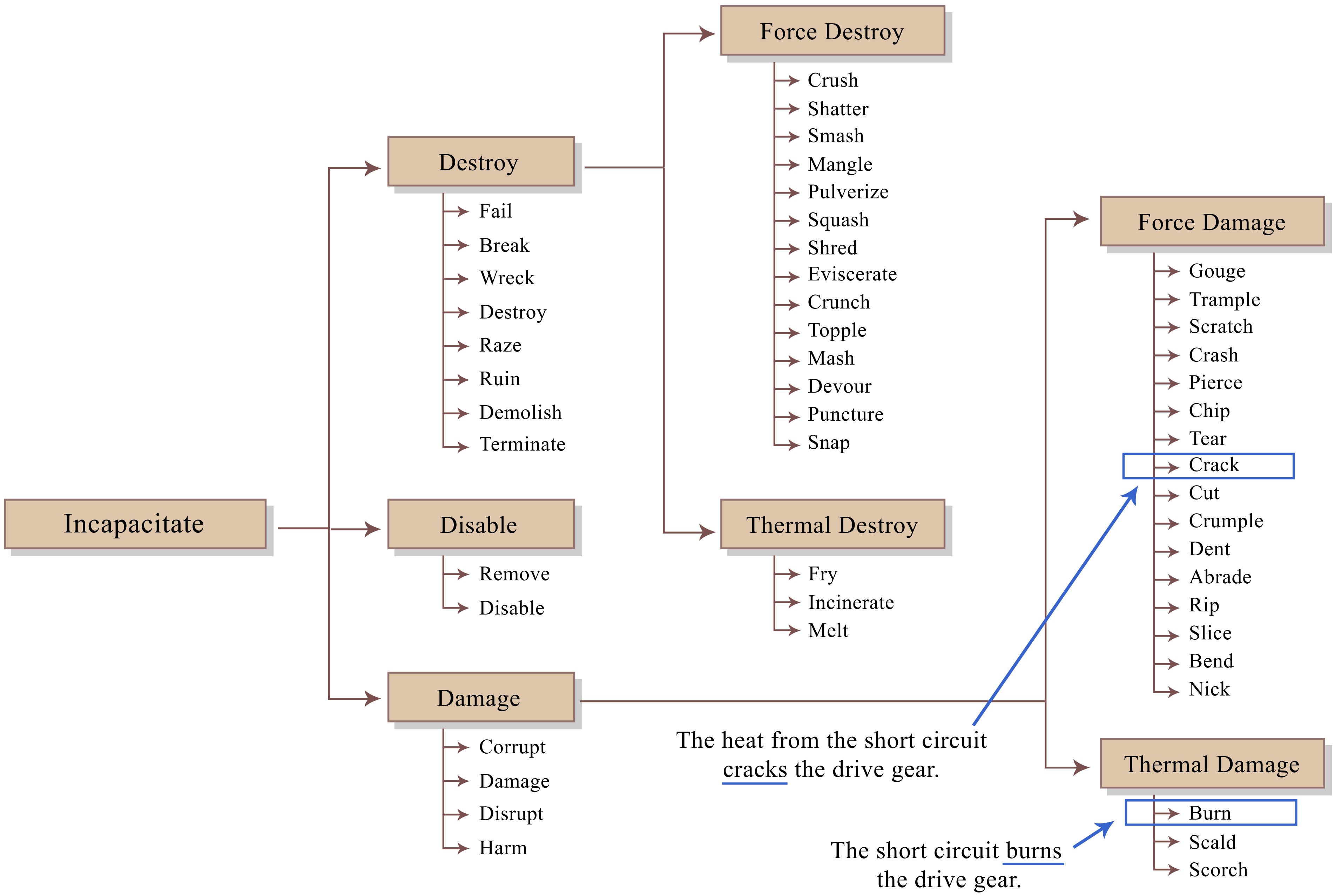


The heat from the short circuit cracks the drive gear.

The short circuit burns the drive gear.

Figure by MIT OCW.

A direct word-to-concept ontology for the concept Incapacitate.



Incapacitate

- Destroy
  - Fail
  - Break
  - Wreck
  - Destroy
  - Raze
  - Ruin
  - Demolish
  - Terminate
  - Force Destroy
    - Crush
    - Shatter
    - Smash
    - Mangle
    - Pulverize
    - Squash
    - Shred
    - Eviscerate
    - Crunch
    - Topple
    - Mash
    - Devour
    - Puncture
    - Snap
  - Thermal Destroy
    - Fry
    - Incinerate
    - Melt
- Disable
  - Remove
  - Disable
- Damage
  - Corrupt
  - Damage
  - Disrupt
  - Harm
  - Force Damage
    - Gouge
    - Trample
    - Scratch
    - Crash
    - Pierce
    - Chip
    - Tear
    - Crack
    - Cut
    - Crumple
    - Dent
    - Abrade
    - Rip
    - Slice
    - Bend
    - Nick
  - Thermal Damage
    - Burn
    - Scald
    - Scorch

The heat from the short circuit cracks the drive gear.

The short circuit burns the drive gear.

Figure by MIT OCW.

Document Comparison Summary

- Different ways to represent the same knowledge
- Use determines representation
- Represent only knowledge that is needed
- c.f.: Common sense reasoning, don’t know use or what knowledge is needed.

Success Using Ontologies?

- Domain-specific successes
  - E.g. biomedicine
- More general use shows promise
  - New languages
  - New tools
  - New applications
  - Very active research community

Taxonomic Hierarchies

- Not always obvious what is class, instance, role, etc.

- E.g., what is the relationship between:
  - time duration (instances e.g. 1 hr) and time interval (e.g. 1pm to 2pm today)?
  - water and ocean?
  - mammal and human and a particular human?
  - human and species?
  - book that is a bound volume, book that is abstract concept?

Other Examples

- Enterprise Ontology (Edinburgh; Uschold, et al.)
- Document comparison (Xerox; Everett, et al.)

Other Examples

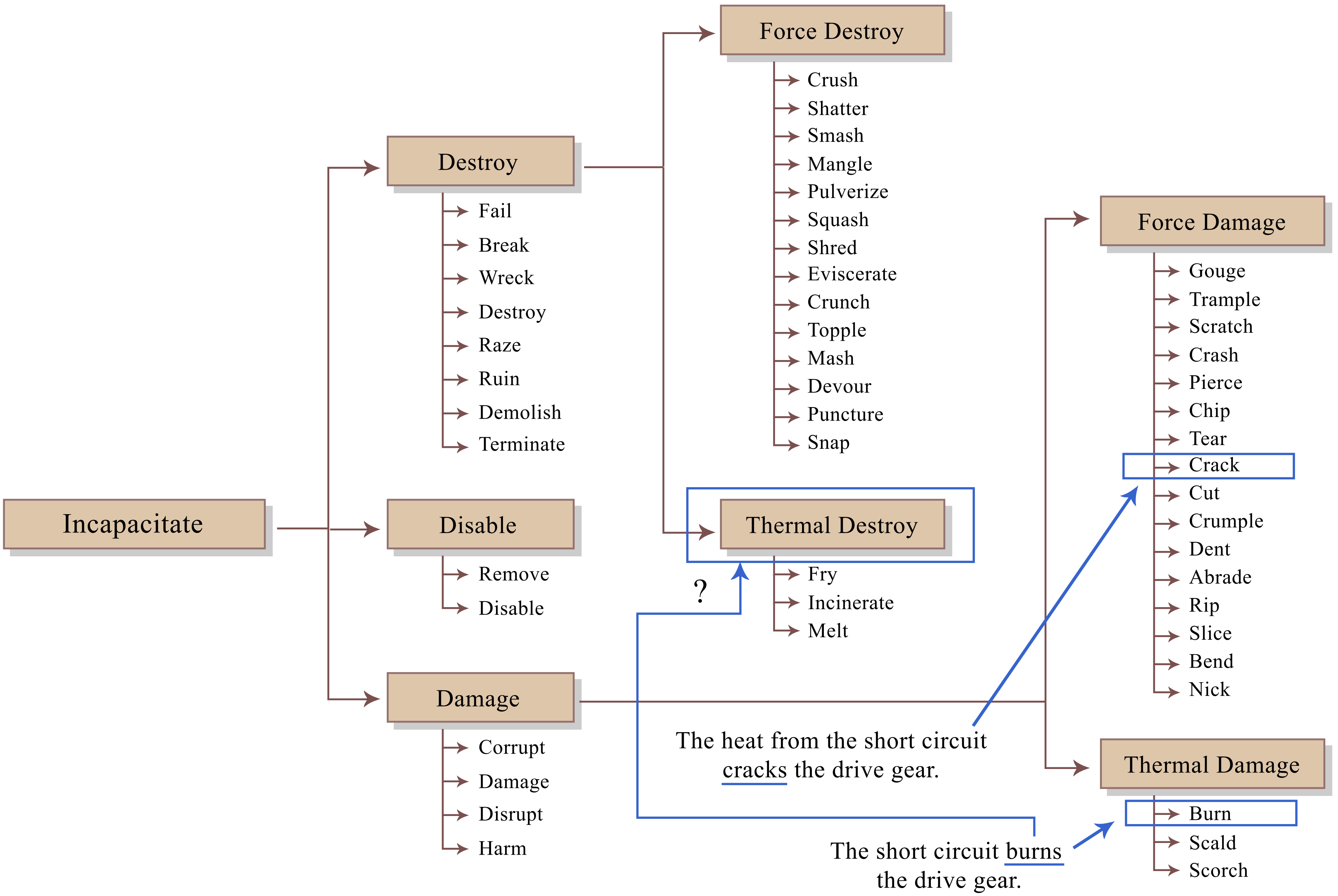
- Enterprise Ontology (Edinburgh; Uschold, et al.)
  - Goal: improve planning via shared enterprise model
  - Meta-ontology: entities, relationships, states of affairs
  - Examples

ACTIVITY etc.	*ORGANISATION*	*STRATEGY*	*MARKETING*	TIME
Activity	Person	Purpose	Sale	Time Line
Activity Specification	Machine	Hold Purpose	Potential Sale	Time Interval
Execute	Corporation	Intended Purpose	For Sale	Time Point
Executed Activity Specification	Partnership	Purpose-Holder	Sale Offer	
T-Begin	Partner	Strategic Purpose	Vendor	
T-End	Legal Entity	Objective	Actual Customer	
Pre-Condition	Organisational Unit	Vision	Potential Customer	
Effect	Manage	Mission	Customer	
Doer	Delegate	Goal	Reseller	
Sub-Activity	Management Link	Help Achieve	Product	
Authority	Legal Ownership	Strategy	Asking Price	
Activity Owner	Non-Legal Ownership	Strategic Planning	Sale Price	
Event	Ownership	Strategic Action	Market	
Plan	Owner	Decision	Segmentation Variable	
Sub-Plan	Asset	Assumption	Market Segment	
Planning	Stakeholder	Critical Assumption	Market Research	
Process Specification	Employment Contract	Non-Critical Assumption	Brand	
Capability	Share	Influence Factor	Image	
Skill	Shareholder	Critical Influence Factor	Feature	
Resource		Non-Critical Influence Factor	Need	
Resource Allocation		Critical Success Factor	Market Need	
Resource Substitute		Risk	Promotion	

Other Examples

- Enterprise Ontology (Edinburgh; Uschold, et al. AIAI, 1998)
- Document comparison (Xerox; Everett, et al. CACM, Feb 02)
  – Goal: Identify similar documents
  – Have: 40,000 technician-authored tips for copier repair
  – Current system: analyzes 15 pairs of similar tips
  – Examples

A direct word-to-concept ontology for the concept Incapacitate.



Incapacitate

Destroy
- Fail
- Break
- Wreck
- Destroy
- Raze
- Ruin
- Demolish
- Terminate

Disable
- Remove
- Disable

Damage
- Corrupt
- Damage
- Disrupt
- Harm

Force Destroy
- Crush
- Shatter
- Smash
- Mangle
- Pulverize
- Squash
- Shred
- Eviscerate
- Crunch
- Topple
- Mash
- Devour
- Puncture
- Snap

Thermal Destroy
- Fry
- Incinerate
- Melt

Force Damage
- Gouge
- Trample
- Scratch
- Crash
- Pierce
- Chip
- Tear
- Crack
- Cut
- Crumple
- Dent
- Abrade
- Rip
- Slice
- Bend
- Nick

Thermal Damage
- Burn
- Scald
- Scorch

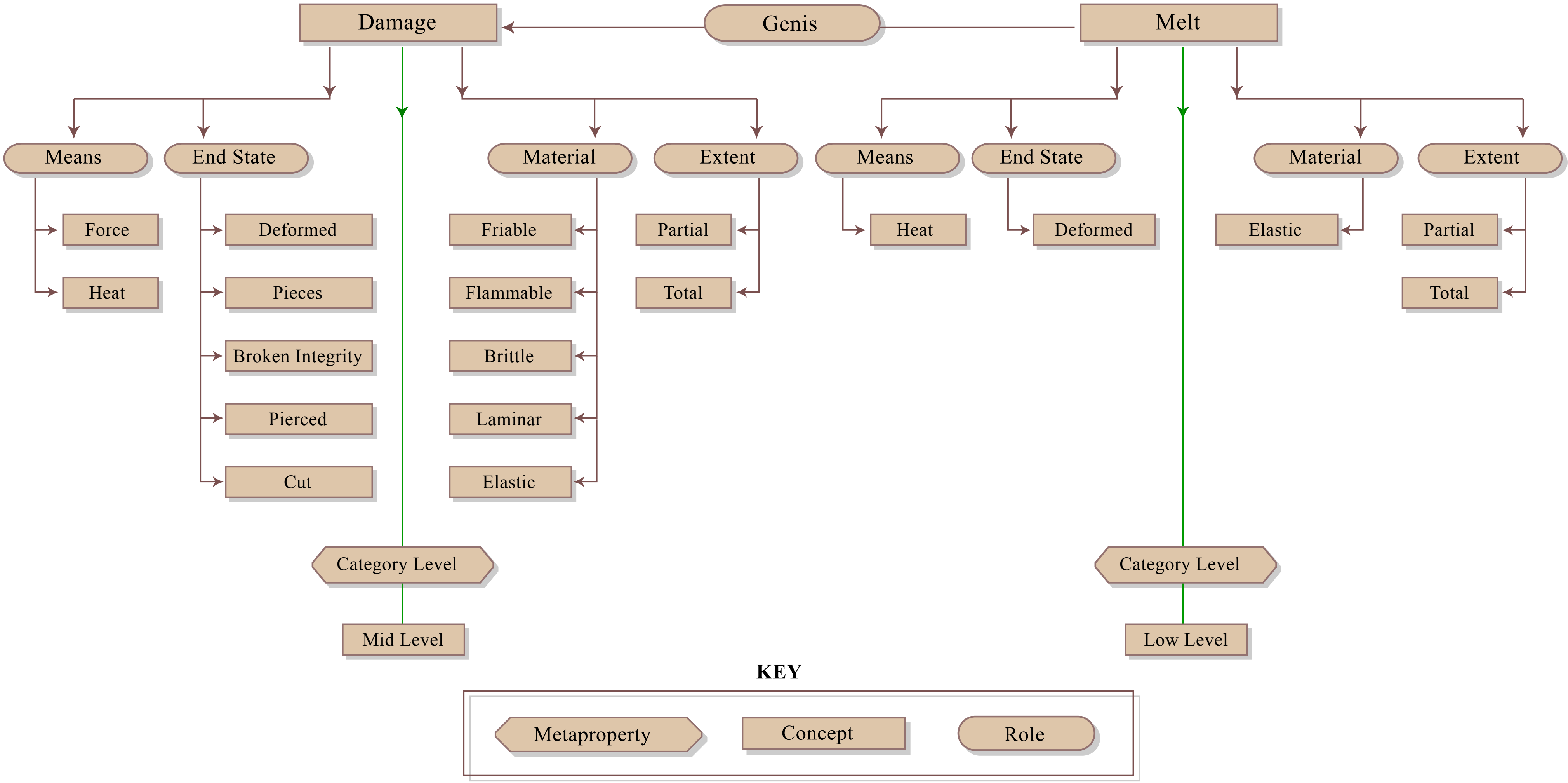
?

The heat from the short circuit cracks the drive gear.

The short circuit burns the drive gear.

Figure by MIT OCW.

A description of logic-based ontology for Damage.



Instead of traversing subsumption relations, logic representation supports arbitrary binary relations between concepts. Matching starts with MidLevel concepts, e.g. Damage.

Figure by MIT OCW.