Modelling gene expression

J. Saric, E. Ratsch, I. Rojas, R. Kania, U. Wittig

European Media Lab Research gGmbH, Heidelberg

http://www.eml-research.de

A. Gangemi

Laboratory for Applied Ontology, ISTC-CNR, Rome
http://www.loa-cnr.it

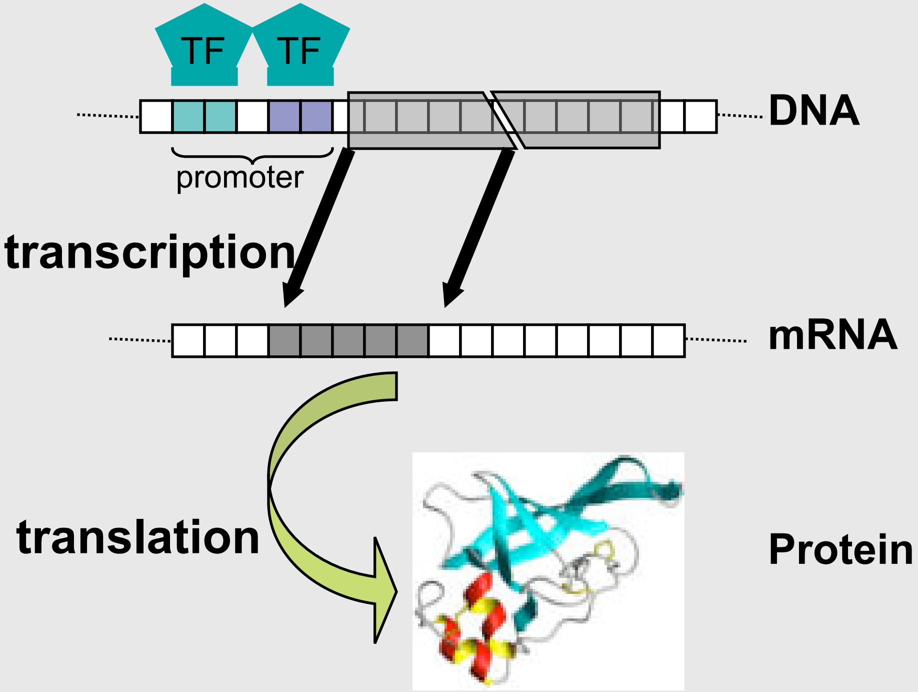
Overview

1. Introduction and Motivation
2. Preliminary information extraction work
4. Formalization and foundation of the ontology
6. Future work and open issues

Motivation

- Huge effort in the bioinformatics community to build large knowledge bases
- Types of entities recorded in KBs are heterogeneous syntactically, linguistically and conceptually
- Gene Ontology
- Static vs. dynamic knowledge assumption
- Conferences (e.g. PSB Biomedical Ontologies)
- Projects (e.g. Semantic Mining FP6 NoE)
- Use of ontologies for
  - information extraction from text
  - categorization and integration of information in/from different sources
  - inference of facts from available (structured) data

A short introduction to gene expression



First steps (an IE experiment)

- Information extraction of gene regulation networks (details in Saric04, ACL proceedings).
- Case study organism: Yeast.

The system had to answer the **questions**:

- Which proteins (transcription factors) regulate the expression of which genes?
- Which type of regulation is mentioned (i.e. up-regulation, down-regulation, underspecified)?
- Which is the organism that this takes place in?

Methods:

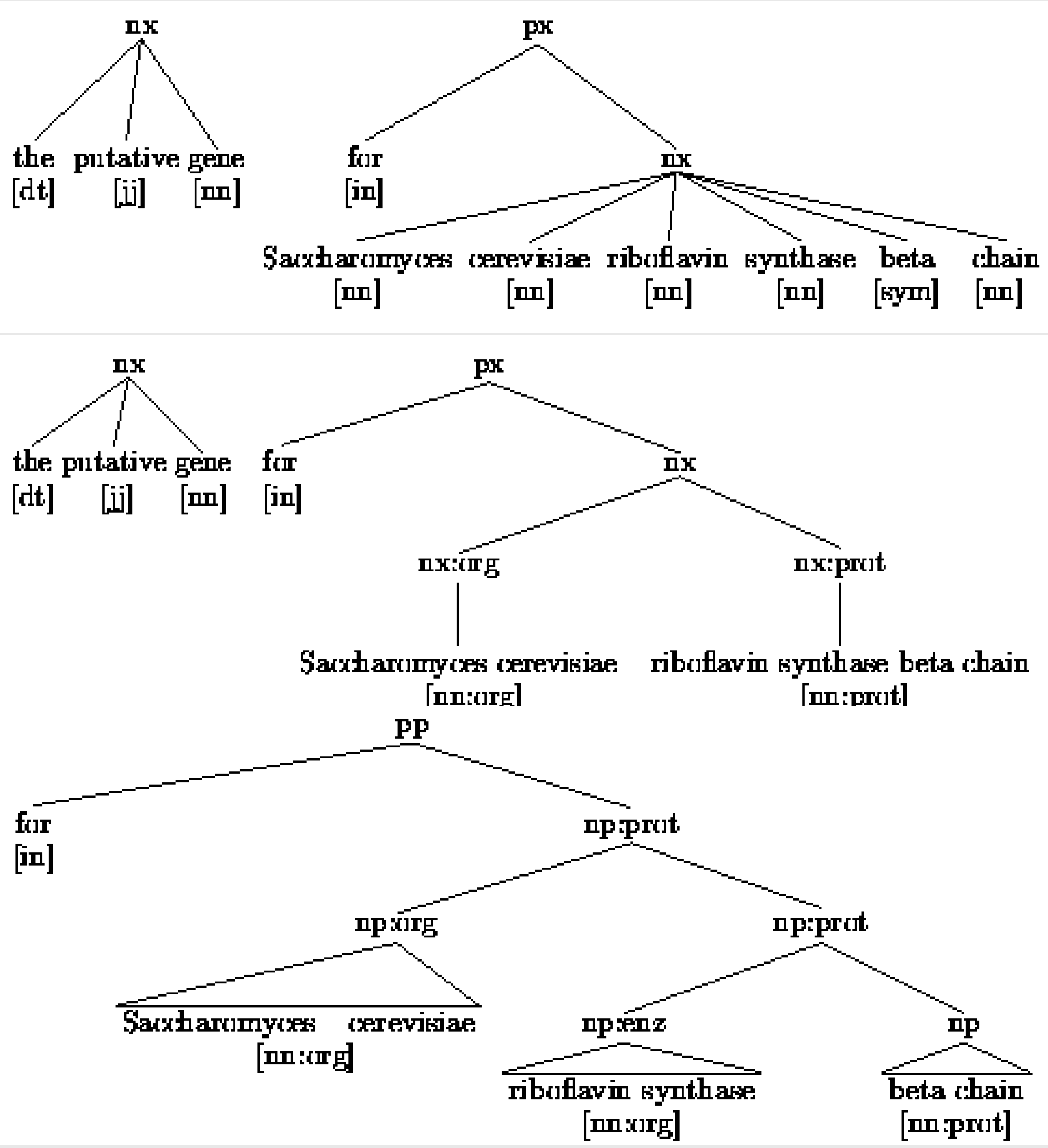
- Shallow NLP techniques
- Hand-crafted rules detecting linguistic patterns

... the putative gene for Saccharomyces cerevisiae riboflavin synthase beta chain ...

the	dt
putative	jj
gene	nn
for	in
Saccharomyces	nn
cerevisiae	nn
riboflavin	nn
synthase	nn
beta	sy
chain	m
	nn

the	dt
putative	jj
gene	nn
for	in
Saccharomyces cerevisiae	nn:orgy
riboflavin synth. beta chain	nn:nnpg

the	dt
putative	jj
gene	nn
for	in
Saccharomyces cerevisiae	nn:org
riboflavin synthase	nn:enz
beta	sym
chain	nn



Characteristics of the system

- Medline Corpus (MeSH terms)
- Tokenisation and multi-word detection
- Part-of-speech tagging
- Semantic labeling
  - **Gene and protein names**
  - Cue words for entity recognition
  - Verbs for relation extraction
- Named entity chunking
  - [nxgene The **GAL4** gene]

- Relation chunking

  [nxexpr The expression of

  [nxgene the cytochrome genes

  [nxpg **CYC1** and **CYC7**]]]

  is controlled by

  [nxpg **HAP1**]

NLP needs knowledge

Term boundary recognition needs semantics

What are the borders of the following term?

And, how can we re-construct the nested (compositional) structure?

Eg.

5. *Nuclear factor NF-kappa-B p50 subunit ....*

⇒ Need for a terminological dictionary of proteins and protein families with associated protein functions.

8. *Endotoxin increased NF-kappaB p50/p65 heterodimer binding.*

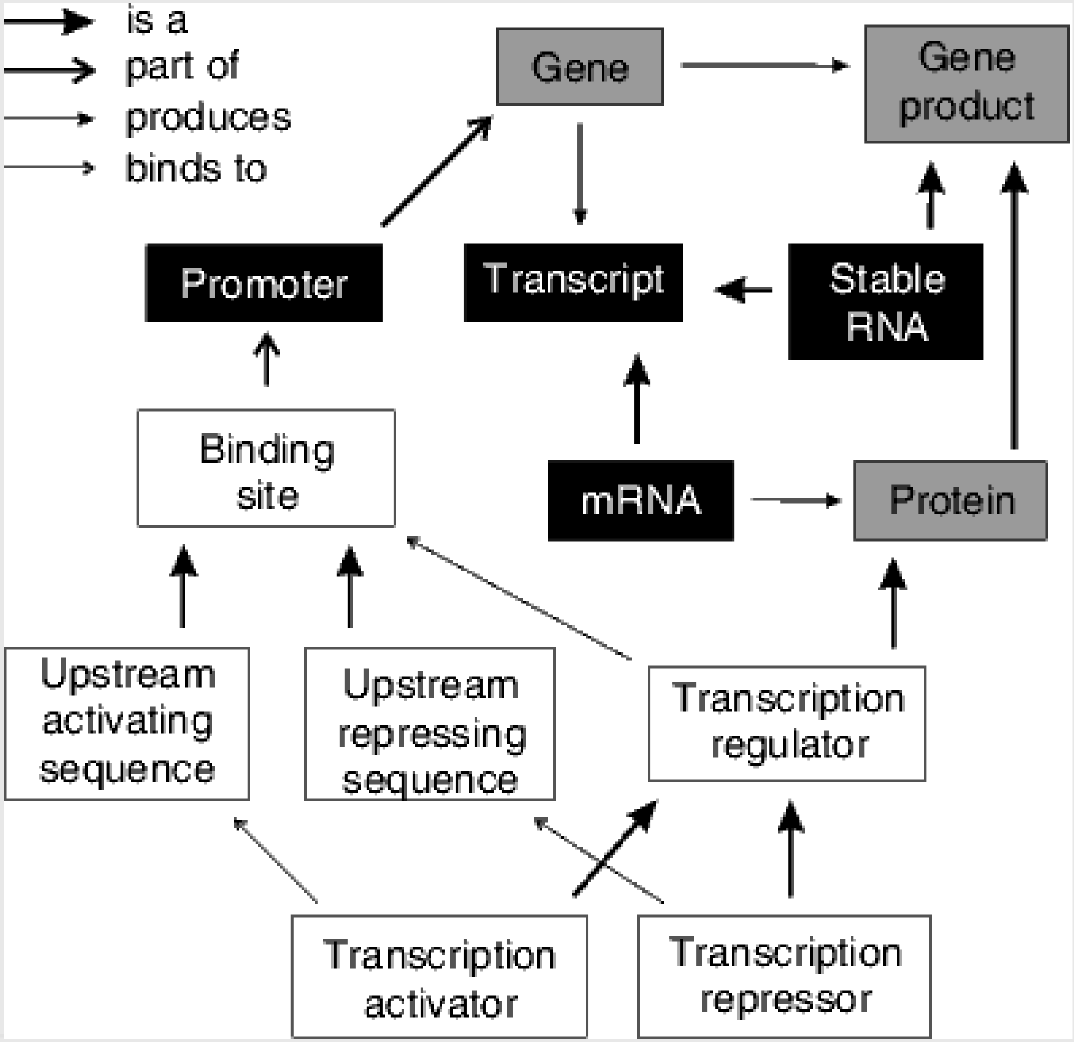
⇒ ***heterodimer*** presuposes existence of A and B with A ≠ B:

a. A = NF-kappaB and B = p50/p65

b. A = p50 and B = p65

The a-reading is false, we need to know that p50 and p65 are proteins being part of the complex NF-kappaB.

The built-in *informal* schema



Results overview

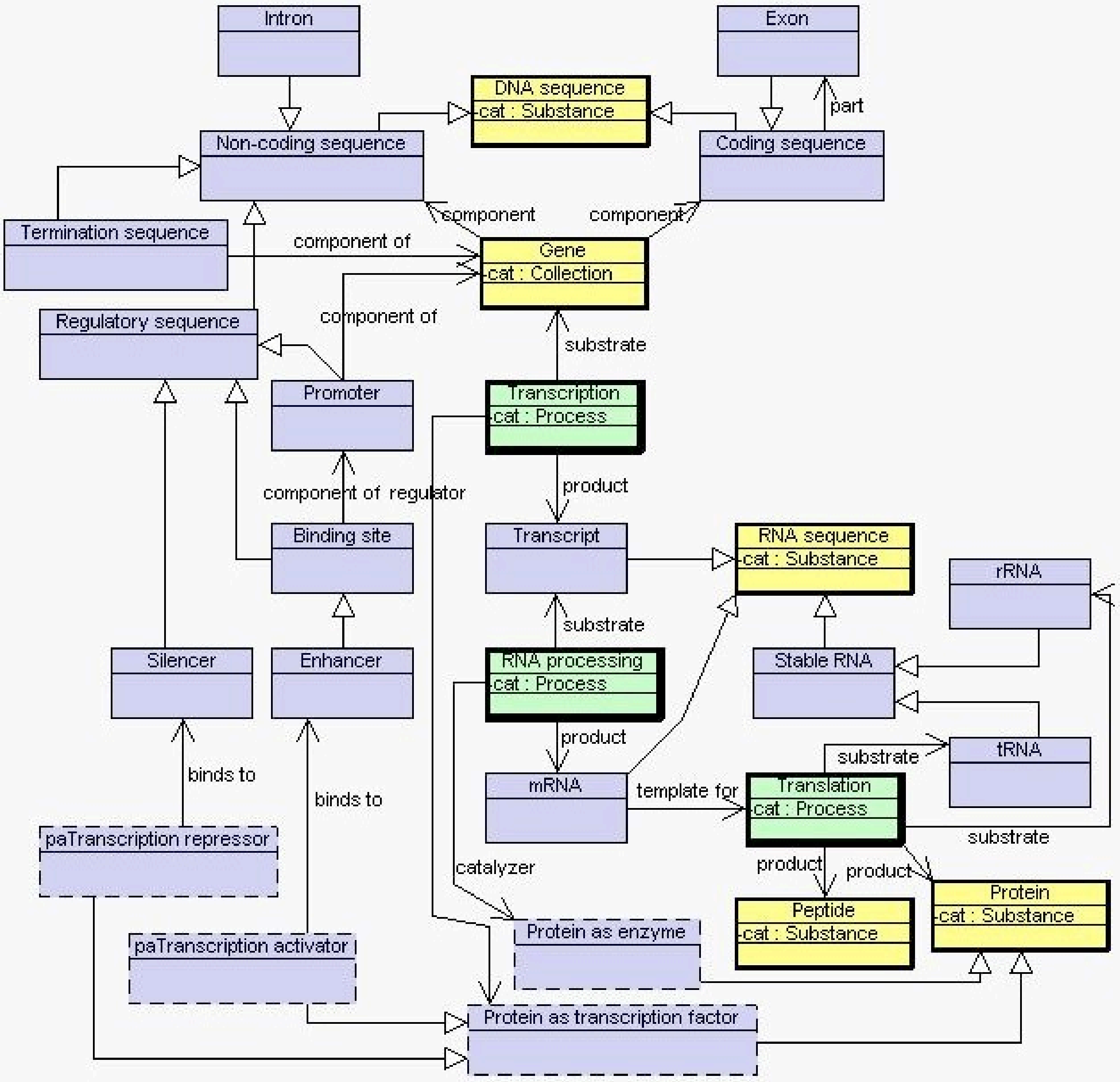
- The **precision** of our method is very good
  - 83-90% on relation extraction
  - 97% on named entity recognition
- Evaluating the **recall** is difficult, estimate:
  - ~30% (looking through 250 of 44,354 sentences that contain at least two gene/protein names)

⇒ **The quality of our results are not so bad, but …**

... some drawbacks

1. Recognising terminology within a text:
   - What is a technical term?
   - What are the boundaries of the term?
2. Categorisation of recognised terms:
   - What is/are the correct semantic category/ies for a recognised term?
   - The categorisation of the terms cannot be easily done in a compositional way (nestedness & scalability)?
   - Although the template (and pattern) construction reflects an underlying ontology on gene expression, it is hard-wired (implicit).
3. Scalability: although we used rules for related questions (i.e. protein interaction), the scalability of the system is limited.

In order to overcome these drawbacks:
create a more detailed and complete ontology that acts as a backbone for the NLP system -- and also for database design, population, and integration --


Intron
Exon
DNA sequence
-cat : Substance
part
Non-coding sequence
Coding sequence
Termination sequence
component
component
component of
Gene
-cat : Collection
Regulatory sequence
component of
substrate
Promoter
Transcription
-cat : Process
product
component of
regulator
Binding site
Transcript
RNA sequence
-cat : Substance
rRNA
substrate
Silencer
Enhancer
RNA processing
-cat : Process
Stable RNA
product
substrate
tRNA
binds to
mRNA
template for
Translation
-cat : Process
binds to
substrate
paTranscription repressor
catalyzer
product
product
Peptide
-cat : Substance
Protein
-cat : Substance
Protein as enzyme
paTranscription activator
Protein as transcription factor

Basic types and rationale

- DOLCE axiomatic theory (*Descriptive Ontology for Linguistic and Cognitive Engineering*): http://www.loa-cnr.it
- ≈10 basic types, ≈20 basic relations, ≈200 axioms
- Wide-range application: Law, Fishery, Finance, Anatomy, ...
- *Very preliminary* application in biology
- Foundational types use from DOLCE: *Substance*, *Process*, *Collection*
- Foundational (formal) relations used from DOLCE+: *(Proper)Part*, *Component*, *Member*, *Participation*, *Connection*, *Succession*
- Substance types are considered: dna and rna sequence, gene, peptide, protein, nucleotide, aminoacid, etc.
- 3 process types are considered: transcription, RNA processing, translation

Some axioms. Sequences, parts and collections

- Sequence(x) $=_{\mathrm{df}}$ Substance(x) $\wedge$ $\forall y,z.$ (Part(x,y) $\wedge$ Part(x,z)) $\rightarrow$ TransitiveConnection(y,z) $\wedge$ $\exists j,k.$ Part(x,j) $\wedge$ Part(x,k) $\wedge$ StrongConnection(j,k) $\wedge$ DirectSuccessor(j,k)
- * Sequence(x) $\rightarrow$ $\forall y,z.$ (Part(x,y) $\wedge$ Part(x,z)) $\rightarrow$ $\neg$(Successor(y,z) $\wedge$ Successor (z,y))
- dnaSequence(x) $\rightarrow$ $\forall y.$ PartOf(y,x) $\rightarrow$ Deoxyrybosenucleotide(y)
- Gene(x) $\rightarrow$ $\forall y.$ PartOf(y,x) $\rightarrow$ (dnaSequence(y) $\vee$ Deoxyribosenucleotide(y))
- Gene(x) $\rightarrow$ $\exists c,n,o.$ CodingSequence(c) $\wedge$ NonCodingSequence(n) $\wedge$ (= ($c \oplus n$), x) $\wedge$ Organism(o) $\wedge$ in(x,o) $\wedge$ $\neg\exists z.$ ComponentOf(z,c) $\wedge$ ComponentOf(z,n)

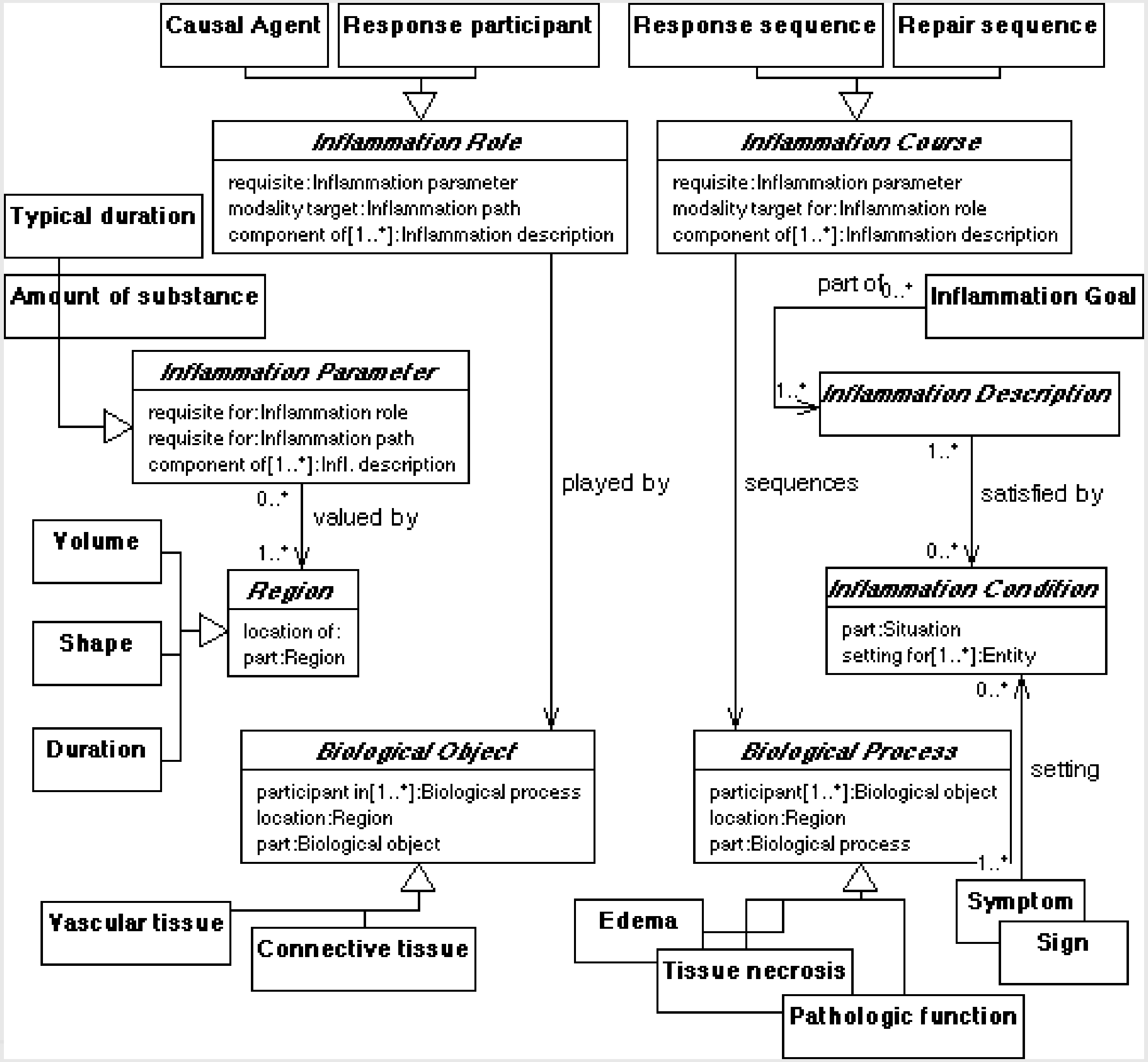
Other axioms. Processes, time, roles.

- $\mathrm{Transcription}(x) \rightarrow \mathrm{ChemicalReaction}(x) \wedge \exists g,o,prom,ts,gt,enz,tf,compl.\ \mathrm{Gene}(g) \wedge \mathrm{in}(g,o) \wedge \mathrm{Substrate}(x,g) \wedge \mathrm{Promoter}(prom) \wedge \mathrm{Substrate}(x,prom) \wedge \mathrm{TerminationSequence}(ts) \wedge \mathrm{Substrate}(x,ts) \wedge \mathrm{Transcript}(gt) \wedge \mathrm{Product}(x,gt) \wedge \mathrm{rnaPolymerase}(enz) \wedge \mathrm{Catalyzer}(x,enz) \wedge \mathrm{TranscriptionFactor}(tf) \wedge \mathrm{Regulator}(x,tf)$
- $\mathrm{Translation}(x) \rightarrow \mathrm{ChemicalReaction}(x) \wedge \exists mr,tr,rib,pep.\ \mathrm{mRNA}(mr) \wedge \mathrm{TemplateFor}(mr,x) \wedge \mathrm{tRNA}(tr) \wedge \mathrm{Substrate}(x,tr) \wedge \mathrm{Ribosome}(rib) \wedge \mathrm{Catalyzer}(x,rib) \wedge \mathrm{Peptide}(pep) \wedge \mathrm{Product}(x,pep)$
- $\mathrm{TemplateFor}(x,y) \rightarrow \mathrm{mRNA}(x) \rightarrow \forall z,w,pep.\ [\mathrm{Codon}(z) \wedge \mathrm{Component}(x,z) \wedge \mathrm{Aminoacid}(w) \wedge \mathrm{Peptide}(pep) \wedge \mathrm{Component}(pep,w) \wedge \mathrm{Product}(y,pep)] \rightarrow \mathrm{Maps}(w,z)$
- $\mathrm{Meets}(x,y) \rightarrow \exists t_1,t_2.\ \mathrm{Loc}(x,t_1) \wedge \mathrm{Loc}(y,t_2) \wedge t_1<t_2$
- $\mathrm{Translation}(x) \rightarrow \exists y.\ \mathrm{Transcription}(y) \wedge \mathrm{Meets}(x,y)$

Foundational issues

- Gene as a "knowledge object": functional collection, what unity criterion? (Inferred from transcript results? Characters? Evolutionary constraints?)
- Gene for an organism: type or token? What is the prototypical gene, given individual variability? Similarly for genome:
- $\mathrm{Genome}(x) \rightarrow \exists y.\ \mathrm{Organism[type]}(y) \wedge \forall z.\ \mathrm{Gene}(z) \wedge \mathrm{in}[*](z,y) \rightarrow \mathrm{Member}(x,z)$
- *Formal* vs. *material* relations: e.g. *connection* vs. *covalent binding*
  - Two different layers in the ontology?
  - Sequences are at the functional or at the substantial layer?
- How to formalize interaction btw different layers/systems?
  - E.g. membrane topology and gene processes
  - E.g. gene functional sequences and protein biochemical structure
- Should we be engaged in these issues?

Further work: Ontology design patterns for functional ontologies



Causal Agent

Response participant

Response sequence

Repair sequence

Inflammation Role
requisite:Inflammation parameter
modality target:Inflammation path
component of[1..*]:Inflammation description

Inflammation Course
requisite:Inflammation parameter
modality target for:Inflammation role
component of[1..*]:Inflammation description

Typical duration

Amount of substance

Inflammation Goal

Inflammation Parameter
requisite for:Inflammation role
requisite for:Inflammation path
component of[1..*]:Infl. description

Inflammation Description

Volume

Shape

Duration

Region
location of:
part:Region

Inflammation Condition
part:Situation
setting for[1..*]:Entity

Biological Object
participant in[1..*]:Biological process
location:Region
part:Biological object

Biological Process
participant[1..*]:Biological object
location:Region
part:Biological process

Vascular tissue

Connective tissue

Edema

Tissue necrosis

Pathologic function

Symptom

Sign