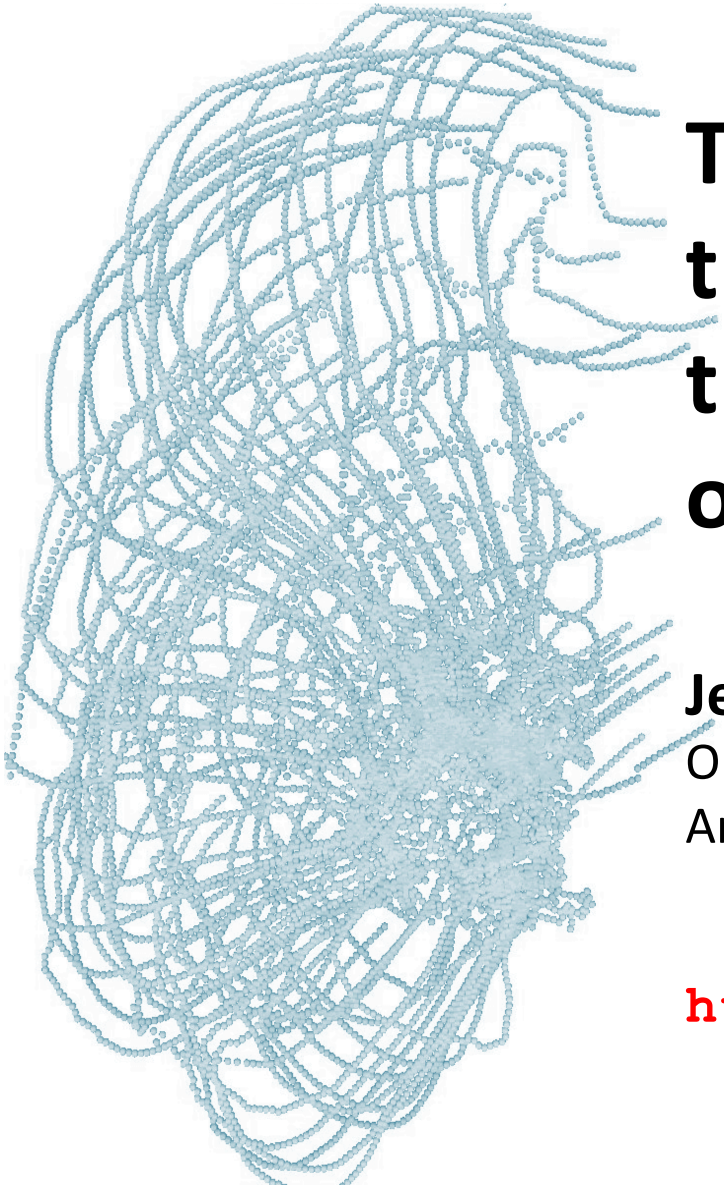
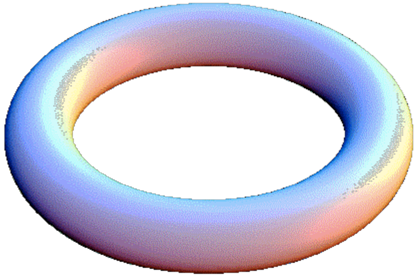


The Modeling and the Simulation of the Fluid Machines of Systems/Synthetic Biology

Jean-Louis Giavitto
Olivier Michel
Antoine Spicher



http://mgs.spatial-computing.org

TOC

1. Introduction
from silicon based information processing
to fluid (biological) machines

2. What are the good abstractions for fluide machinery ?
towards the engineering of biological systems

3. The MGS project
declarative spatial computing

4. Modeling morphogenesis
the growth of the meristem

5. The iGEM-Paris 2008 device
a synthetic « multicellular bacterium »

Home message

1. **New kind of computational machines with "dynamic structure"**
   strange loop between structure and processes
   not new (program = data)
   but not understood (*e.g.*, type discipline to avoid that)

2. **Space matters**
   compartmentalization and beyond

3. **Killer app.**
   systems & synthetic biology
   the nano-world: form=function

4. **Usual tools of computer science are relevant**
   but the focus, the questions and the answers are new
   e.g., termination in rewriting

5. **Versatility of the MGS approach**
   time evolution = rewriting strategy
   kind of space = kind of objects to be rewritten

from
silicon based information processing
to
fluid (biological) machines

A story that could have been parallel...

- 1936 : the Turing machine

- 1928 : DNA as the support of the genetic information

1944 E. Schrödinger: « program » and genetic « code »

- 1947 : the first transistor
- 1958 : first integrated circuit
- 1962 : idea of computer network
- 1967 ARPANET
- 1971 : first microprocessor
- 1972 : electronic mail
- 89~90 : WWW
- 90 : start of the commercial internet
- November 2009 : Cray Jaguar
  1.75 petaflops (10^{15} flop/s)

- 1953 : DNA structure
- 72~73 : RNA sequencing
- 75 : DNA sequencing
- 83 : PCR
- 1989 → 2001 then 2003 :
  the human genome project

… but whose initial motivations are greatly different

Some common concerns: concepts of

- storing
- replicating
- communicating
- modifying

(genetic) informations are studied in computer science (biology).

But there is a great difference between computer science and bio:

- Computer science leads to the engineering of artefactual devices everything is developed from scratch, from hardware to software
- Biology is a *natural science* studying preexisting systems shaped by evolution

However, the *technology* making possible the design and the building or the synthesis of biological machines *now* exists

Computing, yes but why?

- Linear algebra with wetware?
  parallelism? Cf. the moral of the DNA computing story!

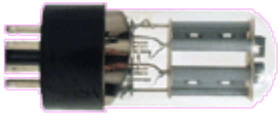
- Cyberphysical systems
  tight coupling between computational processing
  and physical behavior (i/o)

  - 1st generation: embedded systems
    aerospace, automotive, chemical processes, civil infrastructure, energy, healthcare, manufacturing, transportation, entertainment, consumer appliances, etc.

  - 2nd generation: *nanomachines*
    cells as programmable and replicable chemical plants

Technological developments

1906 Triode (L.D. Forest)



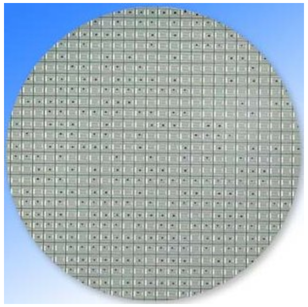
1925 FET transistor (Julius Edgar Lilienfeld)

1947 *Bell Labs*, John Bardeen, Walter Houser Brattain
& William Bradford Shockley:
the first *contact transistor*

1958 First integrated circuit (TI)

1973 PDP8 (DEC)

1971 1st microprocessor (Intel 4004 – 2250 transistors)



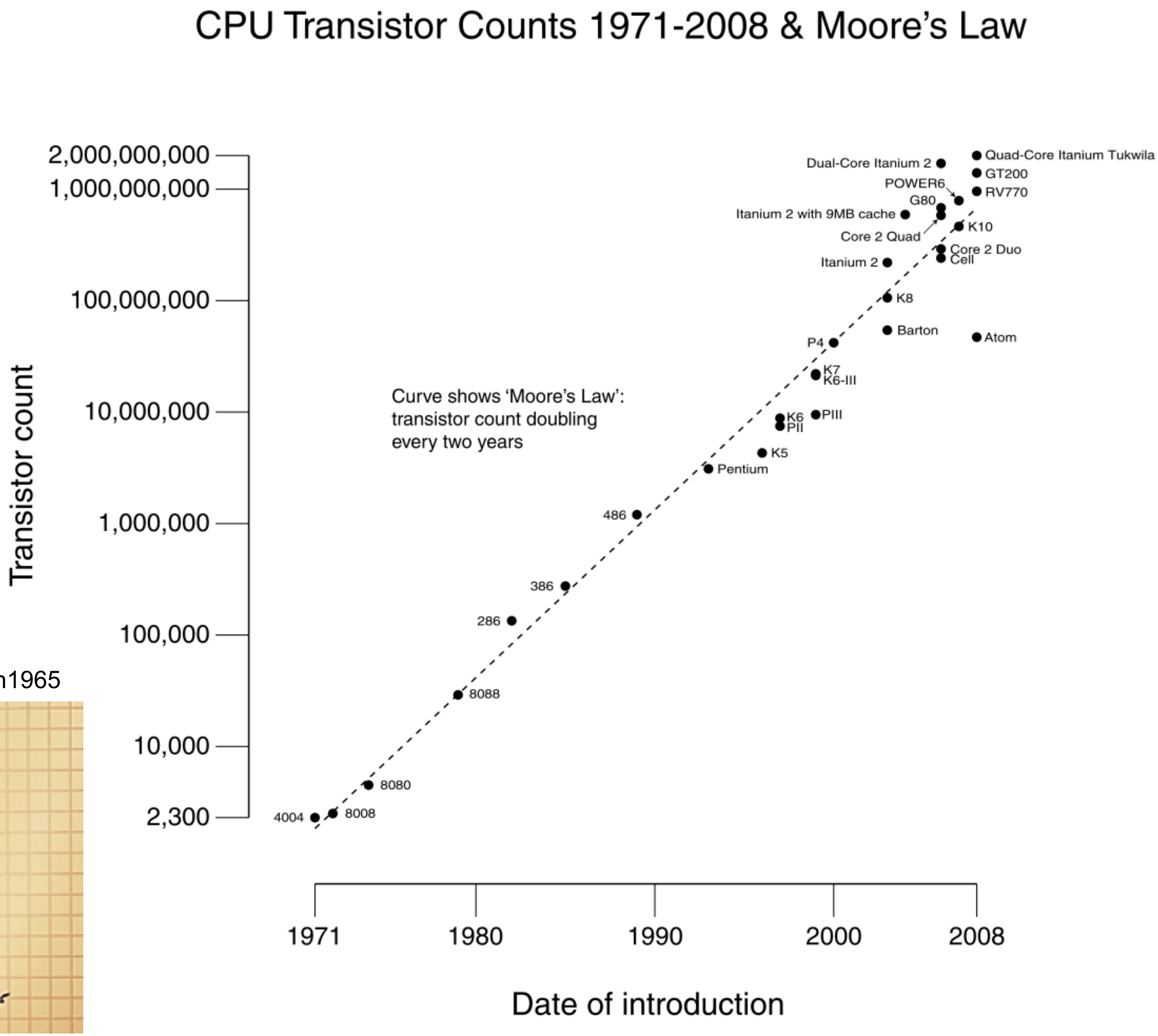
1972 Intel 8008 (8 bits)

1978 Intel 8086 (16 bits)

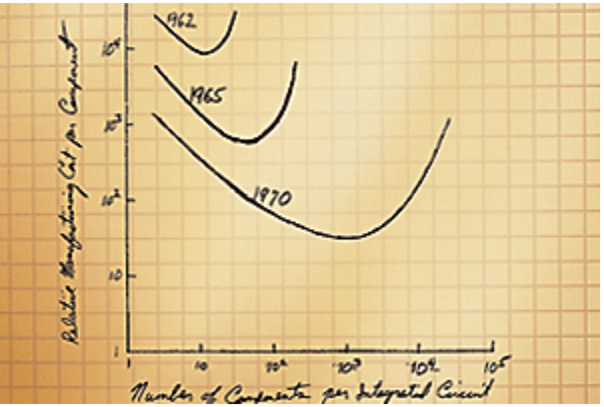
2007 Intel Xeon 7150, 1.3x10^9 transistors, 3.5 GHz, **150 watt**

Simulation on Blue Gene (4096 pe, 256Mo RAM - 360 T-flops)
of 1 second of half mouse brain functioning

The law of Mr. Moore



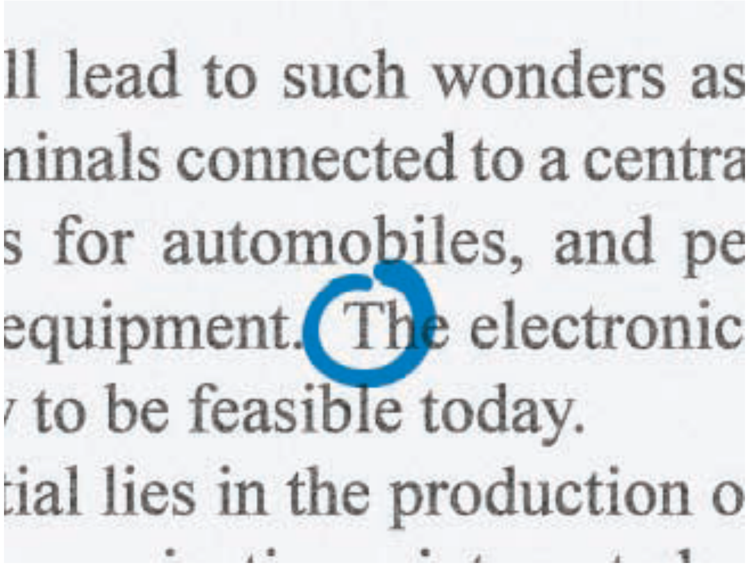
Graphe original de Gordon Moore en1965



An extraordinary decrease

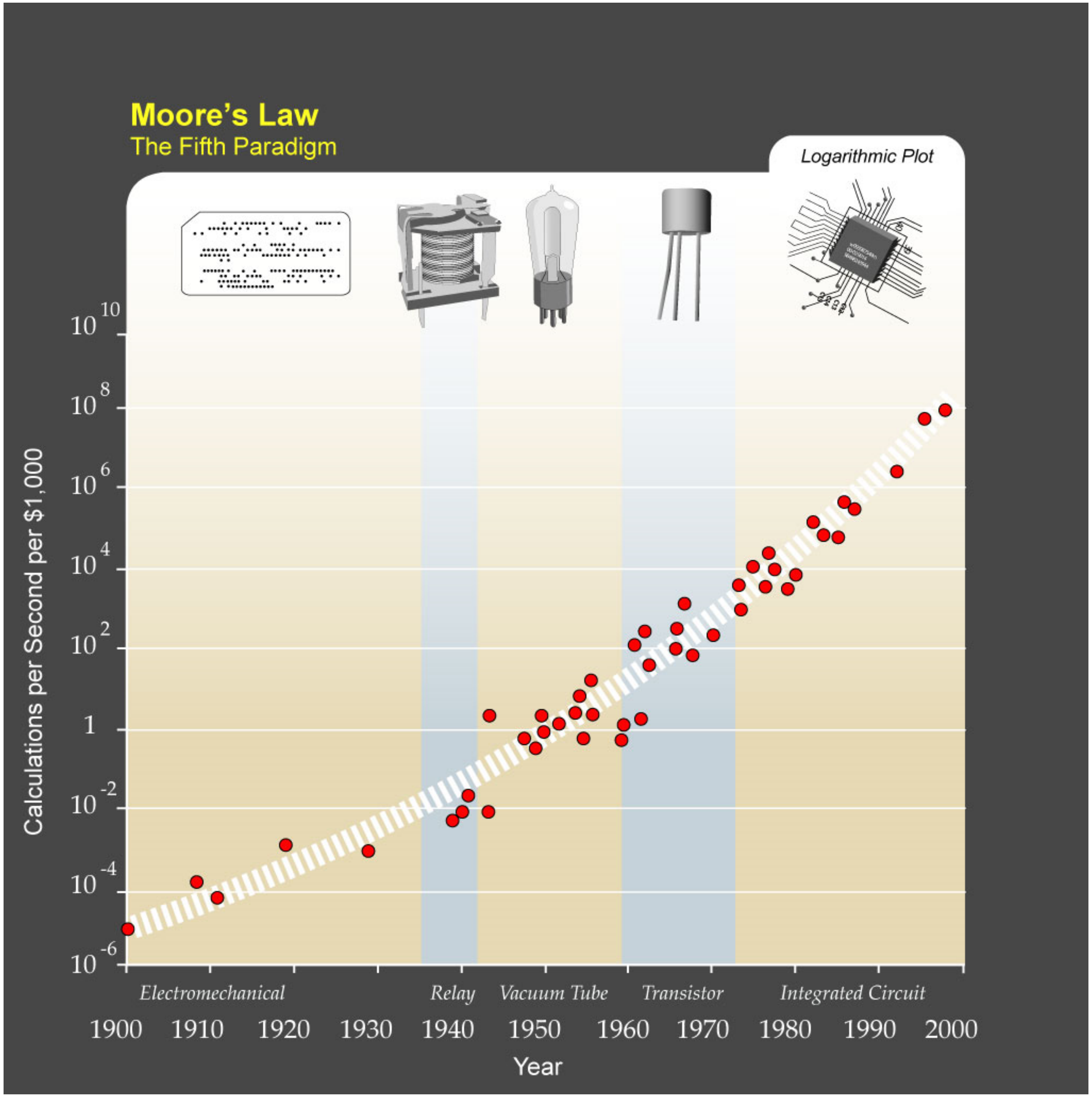
On the road to a billion transistors per chip, transistors are so small that about *200 million of them could fit on the head of each of these pins*.

The price per transistor on a chip has dropped dramatically. Some people estimate that the price of a transistor is now about the same as that of one printed newspaper character.



ll lead to such wonders as
ninals connected to a centra
s for automobiles, and pe
equipment. The electronic
to be feasible today.
tial lies in the production o

That goes back *before* the transistor…

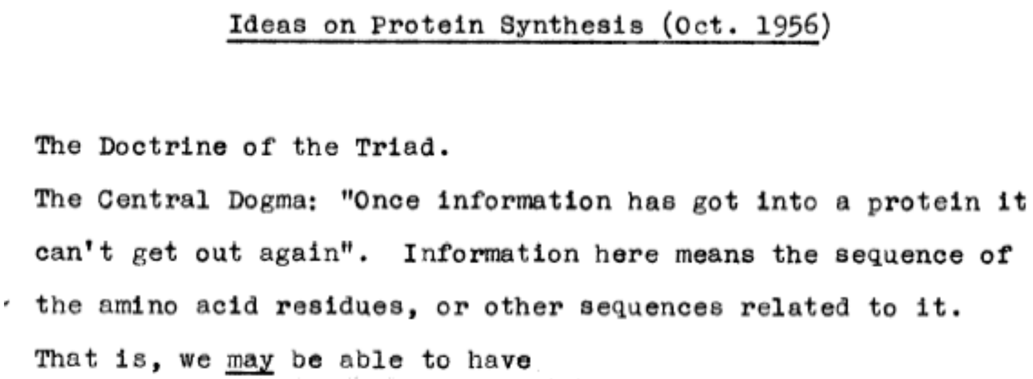


If we compare to molecular biology…

From a computer scientist perspective!

The three periods of molecular bio (1)

- Francis Crick – 1956/58 – The central dogma of mol bio

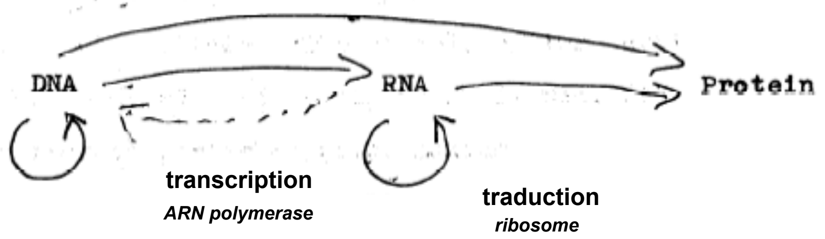


Ideas on Protein Synthesis (Oct. 1956)

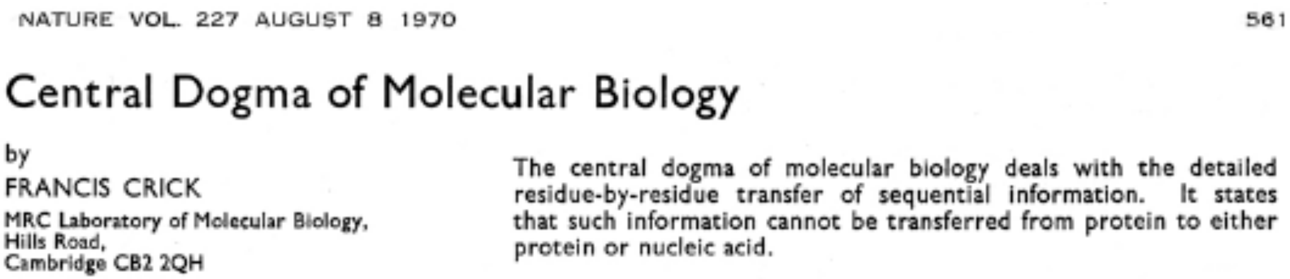
The Doctrine of the Triad.

The Central Dogma: "Once information has got into a protein it can't get out again". Information here means the sequence of the amino acid residues, or other sequences related to it.

That is, we *may* be able to have



- Restated in 1970



NATURE VOL. 227 AUGUST 8 1970 561

Central Dogma of Molecular Biology

by
FRANCIS CRICK
MRC Laboratory of Molecular Biology,
Hills Road,
Cambridge CB2 2QH

The central dogma of molecular biology deals with the detailed residue-by-residue transfer of sequential information. It states that such information cannot be transferred from protein to either protein or nucleic acid.

The three periods of molecular bio (2)

- Analysis
  - of the fine structure of DNA
    - topology
    - irregularity of the double Helix
    - …
  - instability, *dynamics*
    - Gene transfer
    - Spatial conformation dynamics

II : from the DNA structure to genome sequencing & synthesis

  - sequence identification
    - qualitative and quantitative analysis of proteines at various phases of the cell
    - genome wide sequencing (yeast(1997), C-elegans (98), humain (2001), …)

- Synthesis
  - genetic engineering (*e.g.*, yeast producing human proteins)
  - Copy and synthesis of DNA
    - Restriction enzymes
    - recombinant DNA (insertion d'un brin d'ADN dans un ADN existant)
    - PCR ([Mullis 86], …)
    - Gene synthesis: *phosphoramidite chemistry* [Beaucage & Caruthers – 1981] (de 1.25$/bp, jusqu'à 45 kbp, 2 semaines [2006])

The three periods of molecular bio (3)

What is missing for bio-engineering?

The three periods of molecular bio (3)

The conceptual revolution of « engineerization »

Application to biology of engineering principles:

- standardisation (function specification, standardization, libraries, …)
- abstraction (functionnal levels, organization)
- decoupling (conception / implementation)

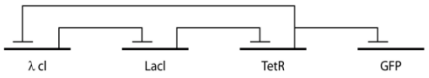
“The work on restriction nucleases not only permits us easily to construct recombinant DNA molecules and to analyse individual genes, but also has led us into the new era of ‘**synthetic biology**’ where not only *existing genes* are *described* and *analyzed* but also new gene arrangements can be constructed and evaluated.”

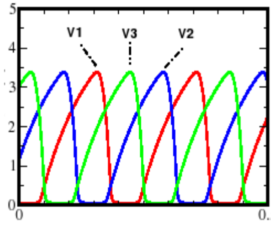
[Szybalski, W. & Skalka, A. Nobel prizes and restriction enzymes. Gene 4, 181–-182 (1978)]

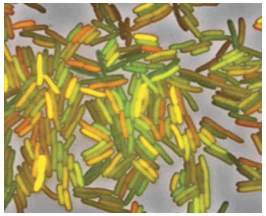
⇨ **Synthetic biology**

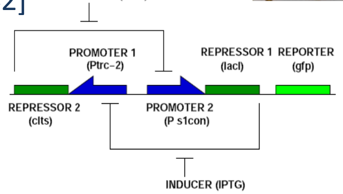
A realistic vision?

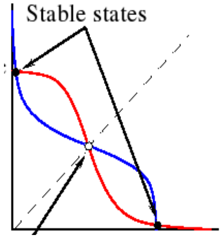
- Proof of concept
  - repressilator [Elowitz & Leibler (2000) Nature 403, 335-338]
  - Flashing E. Coli [Ferber (2004) Science 303, 158-161]
  - toggle switch [Gardner & al (2000) Nature 403, 339-342]



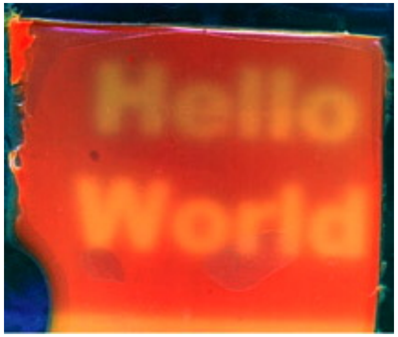








- Sensors
  - Sensing chemicals [TNT (Gibbs (2004) Scientific American 75-81], caffeine [iGEM]
  - Sensing radiation
    (biofilms – [Kobayashi et al. (2004) PNAS 10:8414-841], [iGEM'04])
  - cell-cell communications
    [Weiss & al (2003) Natural Computing 2, 47-84]



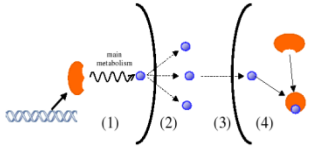
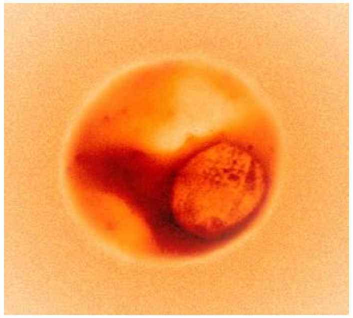


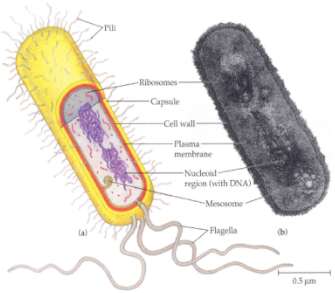
Figure 17. Cell-cell communication schematics: (1) The sender cell produces small signal molecules using certain metabolic pathways. (2) The small molecules diffuse outside the membrane and into the environment. (3) The signals then diffuse into neighboring cells (4) and interact with proteins in the receiver cells, and thereby change signal values.

A realistic vision?

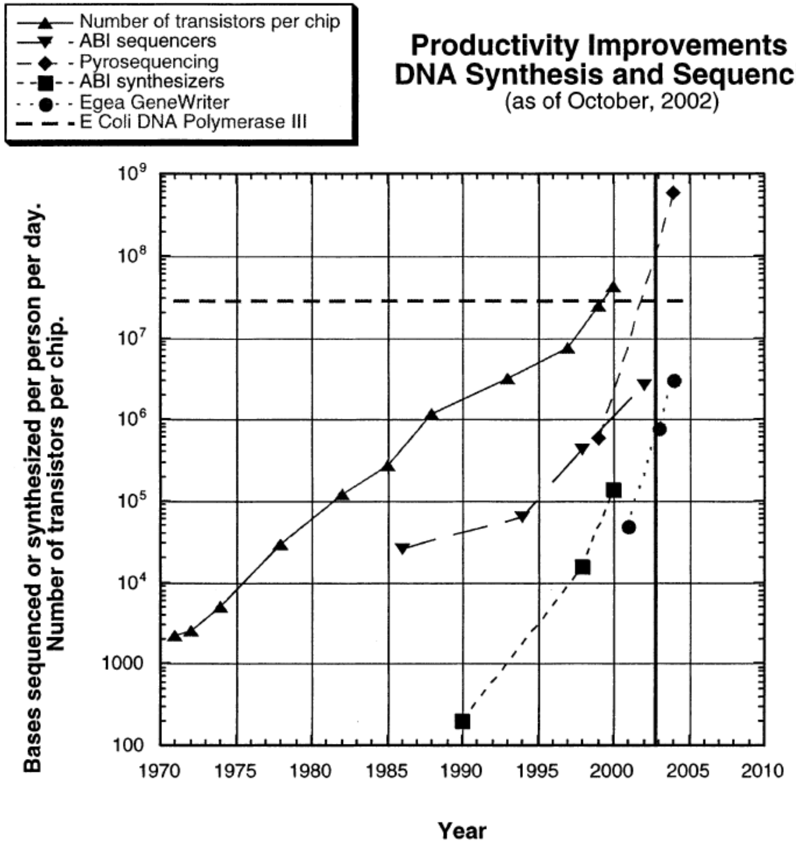
- Bio-medical/Bio-production
  - Drugs (Keasling's *artemisinin* [Martin & al (2003) Nature Biotech 21, 796-802])
  - Targeted delivery [Anderson & al (2005) J. Mol Biol 335, 619-627]
  - Nanobot [Weiss, Knight DNA06 – LNCS 2054, 1-16]
  - timed-drug delivery [F. Molina et al.– 2007]

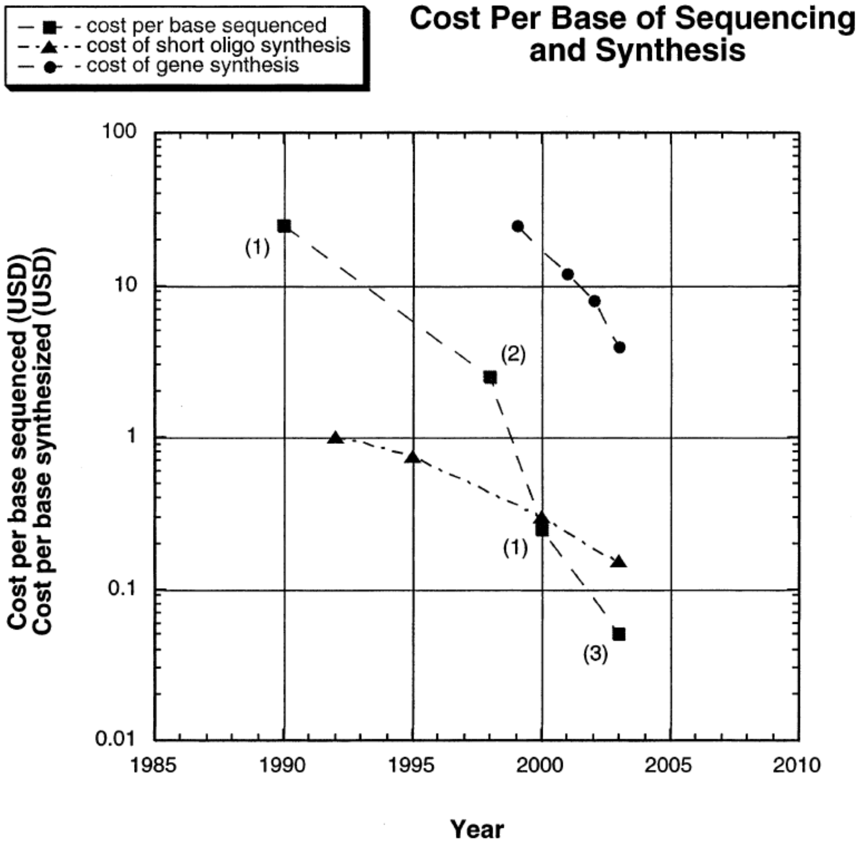


- *Bio-computing*
  - inverter, nand gate, self-repressor, bi-stable switch, oscillators… [R. Weiss]
  - cell-cell communication, signal processing [R. Weiss]



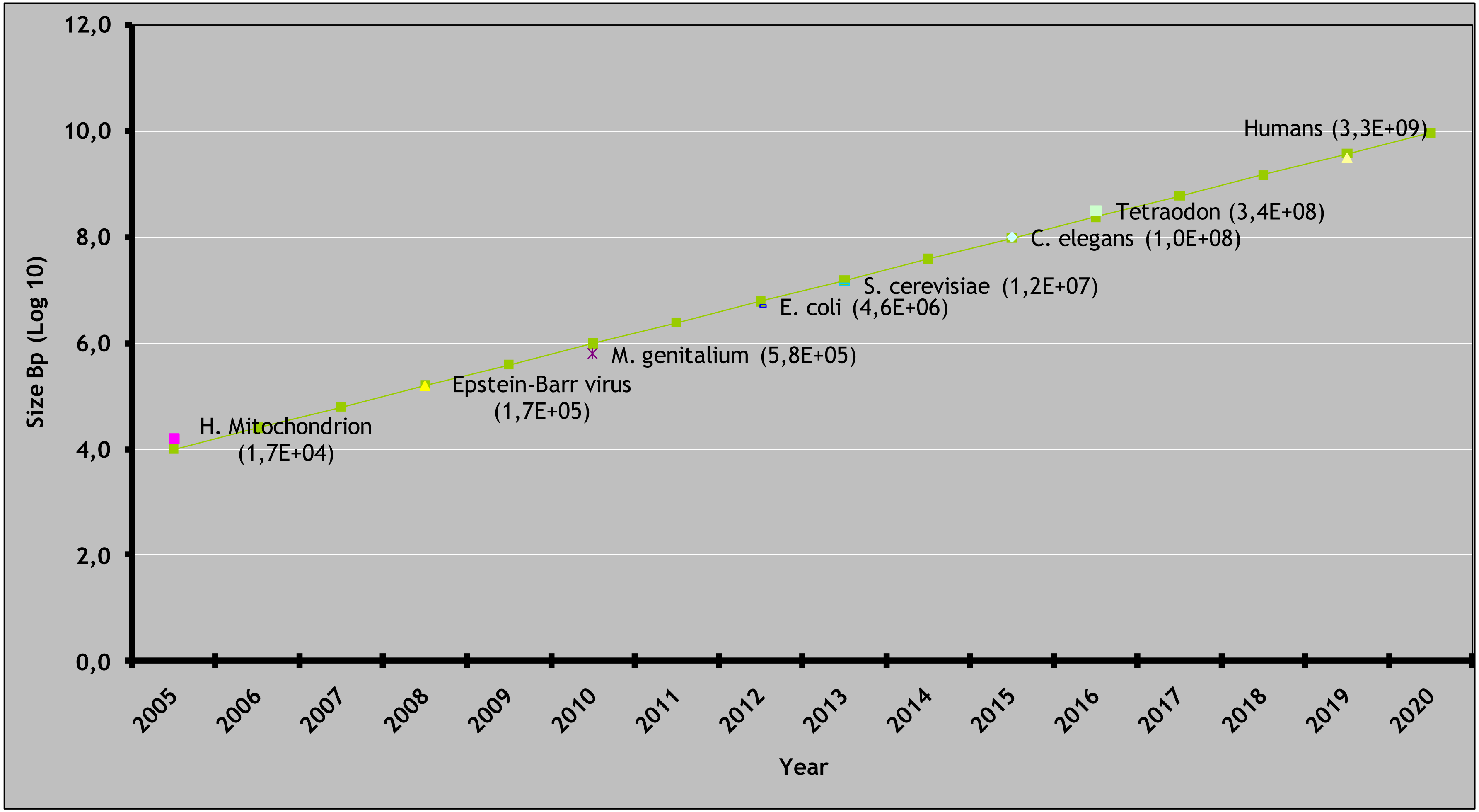
From the Moore law to the Carlson law





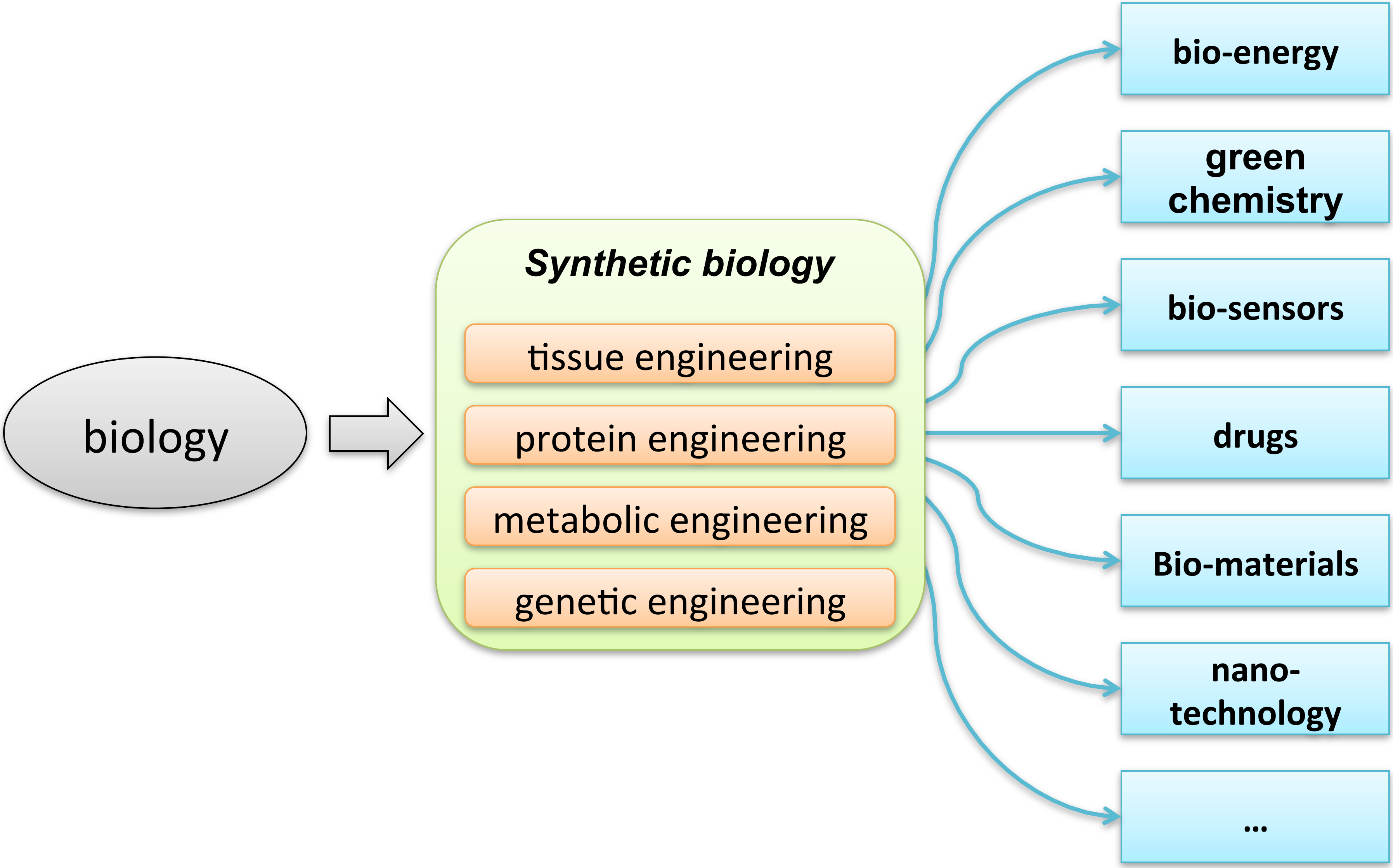
[Carlson, R. (2003).
The pace and proliferation of biological technologies.
Biosecurity and Bioterrorism: Biodefense Strategy, Practice and Science, *1*(3), 203-214.]

The projections are verified



2010 – 2015 : projections for the synthesis of a genome

Applications



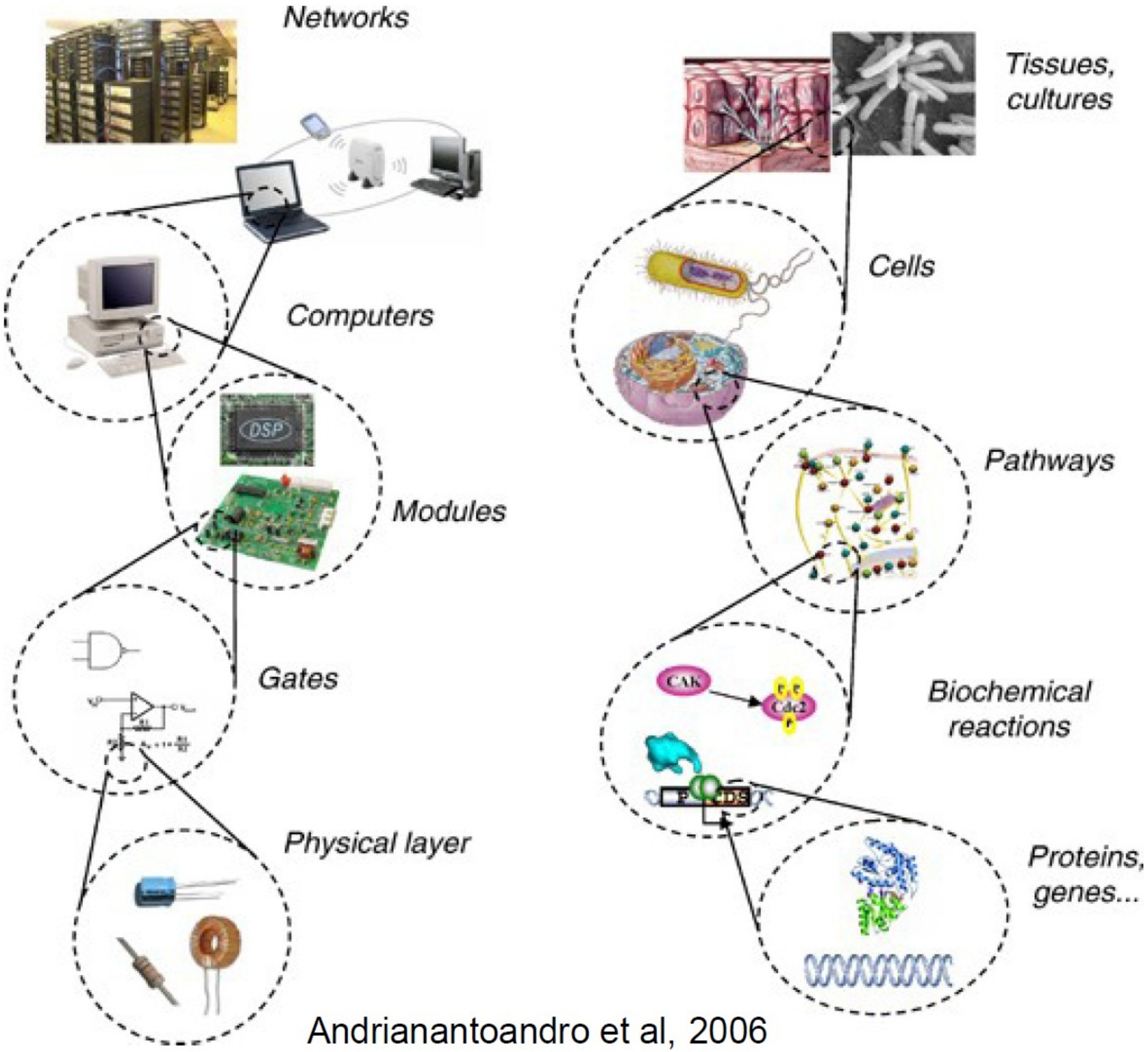
What are the good abstractions to program a cell?

A low level code

```
ACAGGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGCTTCGGAATTTGACCTAAG
ACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGCTTCGGAATTTGACCTAAG
ACCTTATTGTCTAATTGAGATGACCTAAGACCTTATTGTCTAATTGAGAAGGAGGGCC
GGTTGAACAGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGGCTTCGGAATTTGAC
CTAAGACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGCTTCGGAATTTGACC
TAAGACCTTATTGTCTTGACCTAAGACCTTATTGTCTAATTGAGAAGGAATTGAGAAGG
GCCGGTTGAACAGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGGCTTCGGAATTT
GACCTAAGACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGCTTCGGAATTTG
ACTGACCTAAGACCTTATTGTCTAATTGAGAAGGCTAAGACCTTATTGTCTAATTGAGA
AGGGCCGGTTGAACAGGGCTTCGGAATTTGACCTAAGACCTTATTGTCTAATTGAGAAG
GGCCGGTTGAACAGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGGCTTCGGAATTT
GACCTAAGACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGTGACCTAAGAC
ACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGCTTCGGAATTTGACCTAAG
ACCTTATTGTCTAATTGAGATGACCTAAGACCTTATTGTCTAATTGAGAAGGAGGGCC
GGTTGAACAGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGGCTTCGGAATTTGAC
CTAAGACCTTATTGTCTAATTGAGAAGGGCCGGTTGAACAGGGCTTCGGAATTTGACC
TAAGACCTTATTGTCTTGACCTAAGACCTTATTGTCTAATTGAGAAGGAATTGAGAAGG
GCCGGTTGAACAGGTGACCTAAGACCTTATTGTCTAATTGAGAAGGGCTTCGGAATTT
GACCTAAGACCTTATTGTCTAATTGAGAAGGGCCGGTTG
```

Breaking the complexity barrier

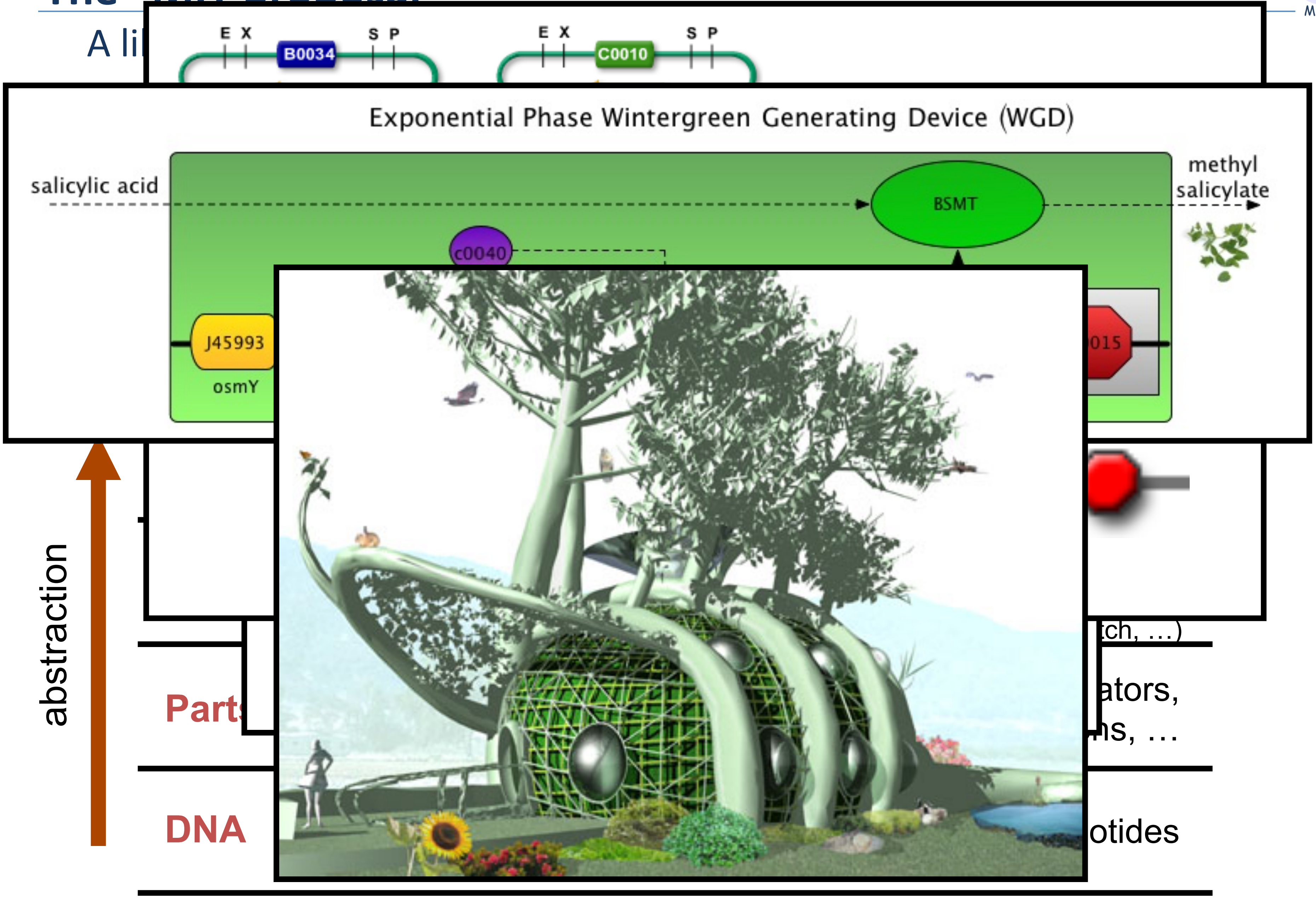
- Levels of abstraction
- Hierarchy
- Decoupling
- Modularity
- I/O



Andrianantoandro et al, 2006

The "MIT proposal"

A li



abstraction

tch, …)

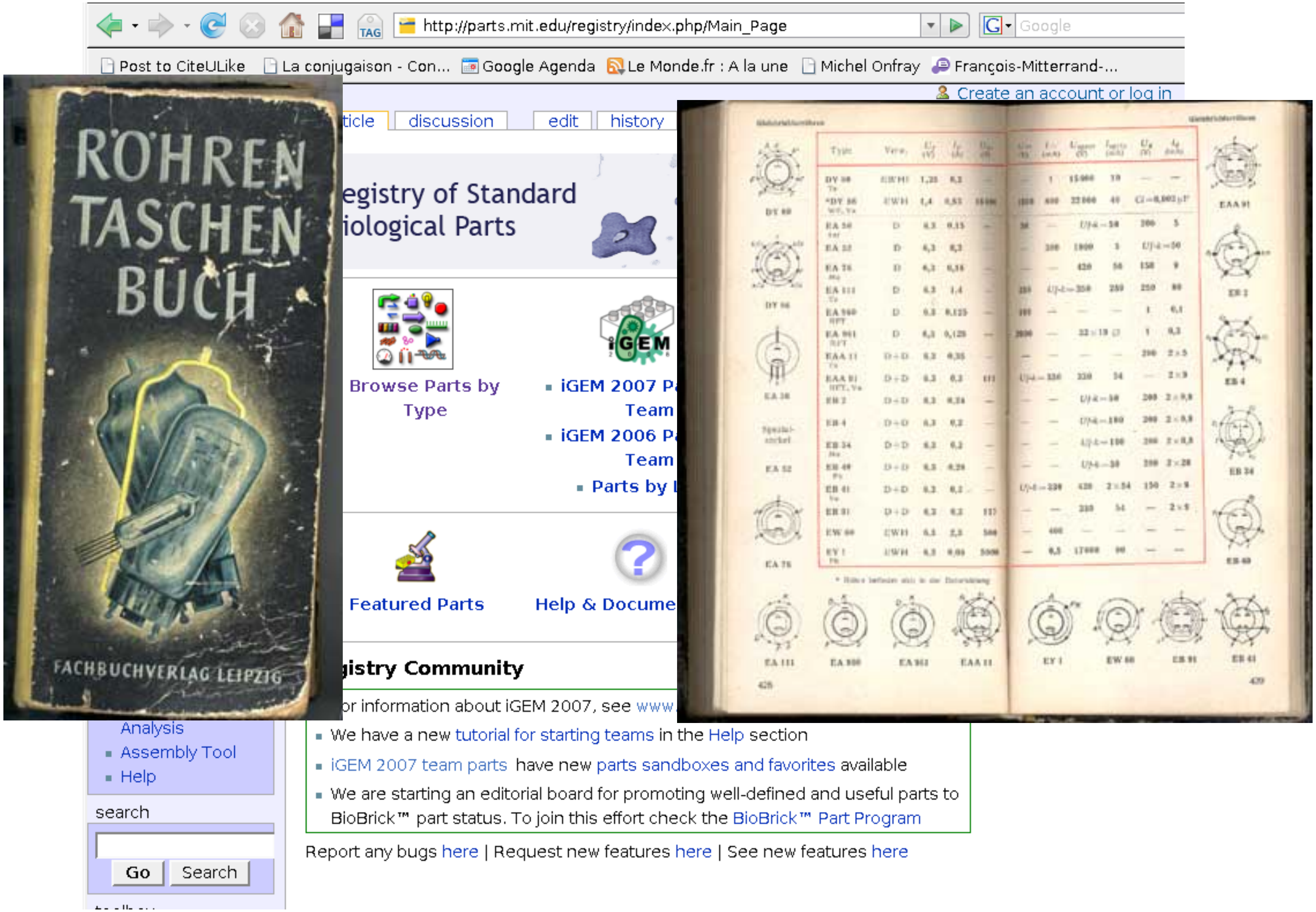
Parts

ators,
ns, …

DNA

otides

A library of standardized componants



http://parts.mit.edu/registry/index.php/Main_Page

Google

Post to CiteULike | La conjugaison - Con... | Google Agenda | Le Monde.fr : A la une | Michel Onfray | François-Mitterrand-...

Create an account or log in

ticle | discussion | edit | history

egistry of Standard
iological Parts

Browse Parts by Type

- iGEM 2007 P
Team
- iGEM 2006 P
Team
- Parts by I

Featured Parts

Help & Docume

gistry Community

or information about iGEM 2007, see www

- We have a new tutorial for starting teams in the Help section
- iGEM 2007 team parts have new parts sandboxes and favorites available
- We are starting an editorial board for promoting well-defined and useful parts to BioBrick™ part status. To join this effort check the BioBrick™ Part Program

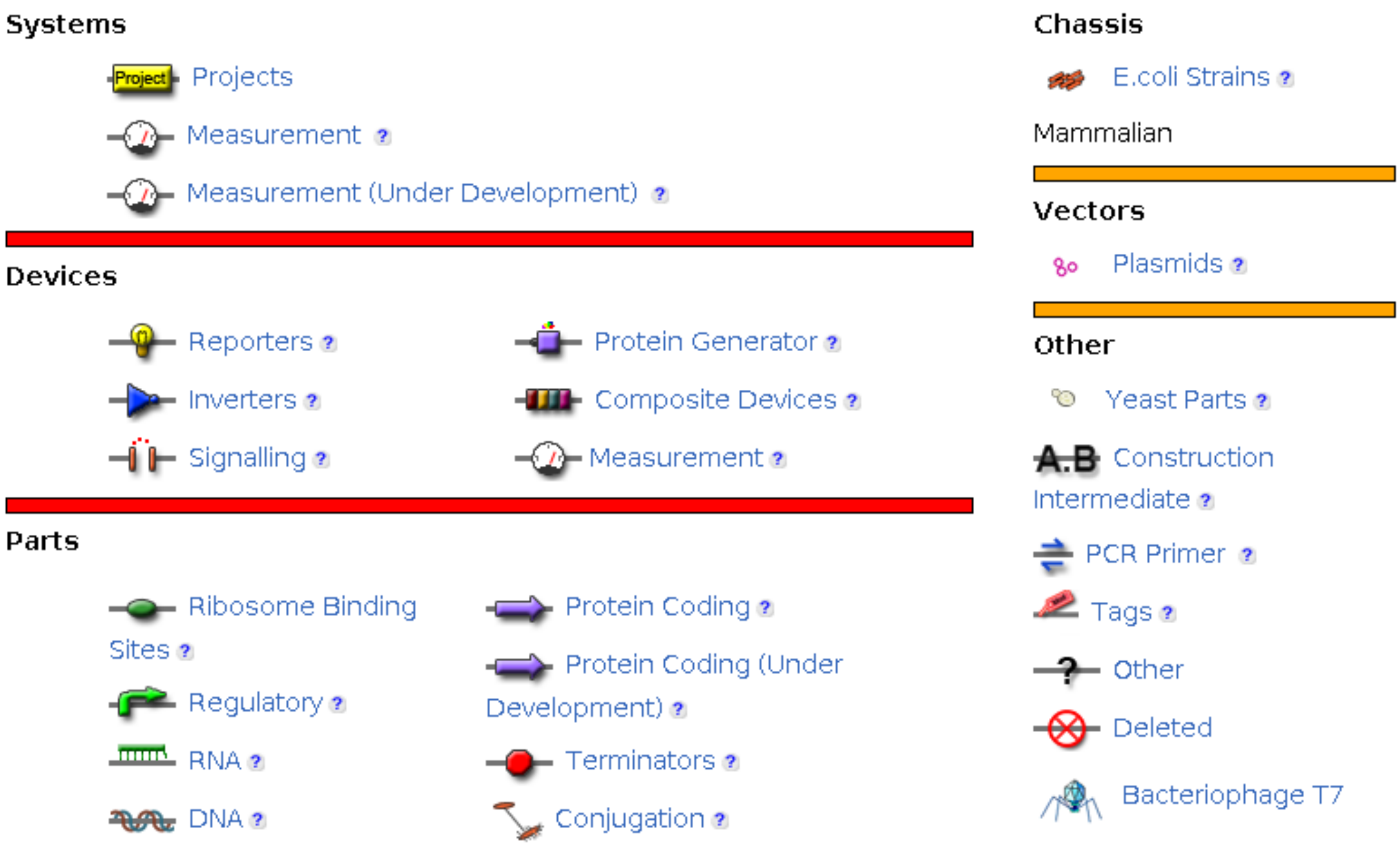
Report any bugs here | Request new features here | See new features here

- Analysis
- Assembly Tool
- Help

search

Go | Search

Several kind of componants



Systems

- Projects
- Measurement
- Measurement (Under Development)

Devices

- Reporters
- Inverters
- Signalling
- Protein Generator
- Composite Devices
- Measurement

Parts

- Ribosome Binding Sites
- Regulatory
- RNA
- DNA
- Protein Coding
- Protein Coding (Under Development)
- Terminators
- Conjugation

Chassis

- E.coli Strains
- Mammalian

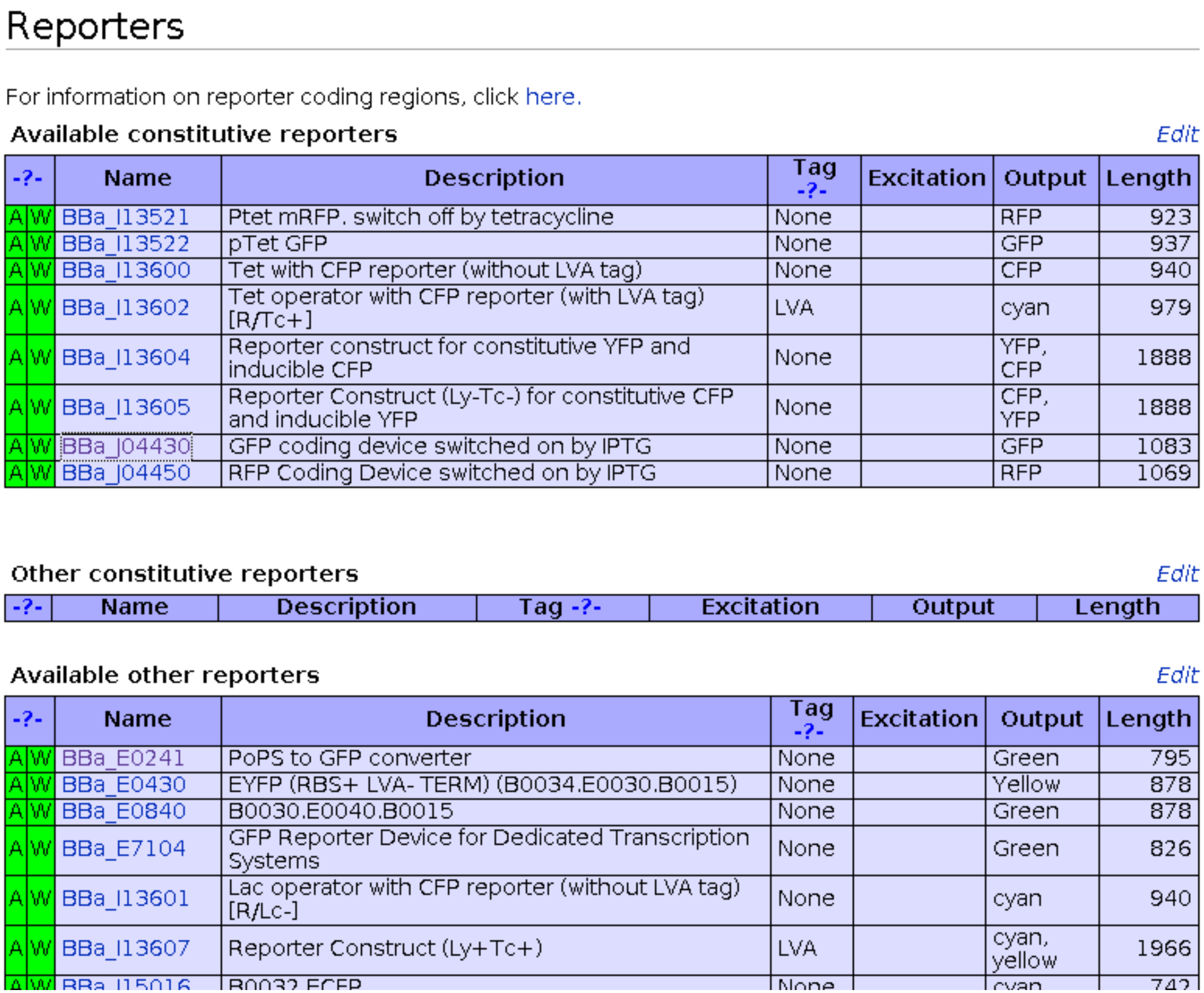
Vectors

- Plasmids

Other

- Yeast Parts
- Construction Intermediate
- PCR Primer
- Tags
- Other
- Deleted
- Bacteriophage T7

A device



Reporters

For information on reporter coding regions, click here.

Available constitutive reporters *Edit*

-?-	Name	Description	Tag -?-	Excitation	Output	Length
A W	BBa_I13521	Ptet mRFP. switch off by tetracycline	None		RFP	923
A W	BBa_I13522	pTet GFP	None		GFP	937
A W	BBa_I13600	Tet with CFP reporter (without LVA tag)	None		CFP	940
A W	BBa_I13602	Tet operator with CFP reporter (with LVA tag) [R/Tc+]	LVA		cyan	979
A W	BBa_I13604	Reporter construct for constitutive YFP and inducible CFP	None		YFP, CFP	1888
A W	BBa_I13605	Reporter Construct (Ly-Tc-) for constitutive CFP and inducible YFP	None		CFP, YFP	1888
A W	BBa_J04430	GFP coding device switched on by IPTG	None		GFP	1083
A W	BBa_J04450	RFP Coding Device switched on by IPTG	None		RFP	1069

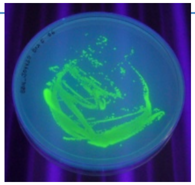
Other constitutive reporters *Edit*

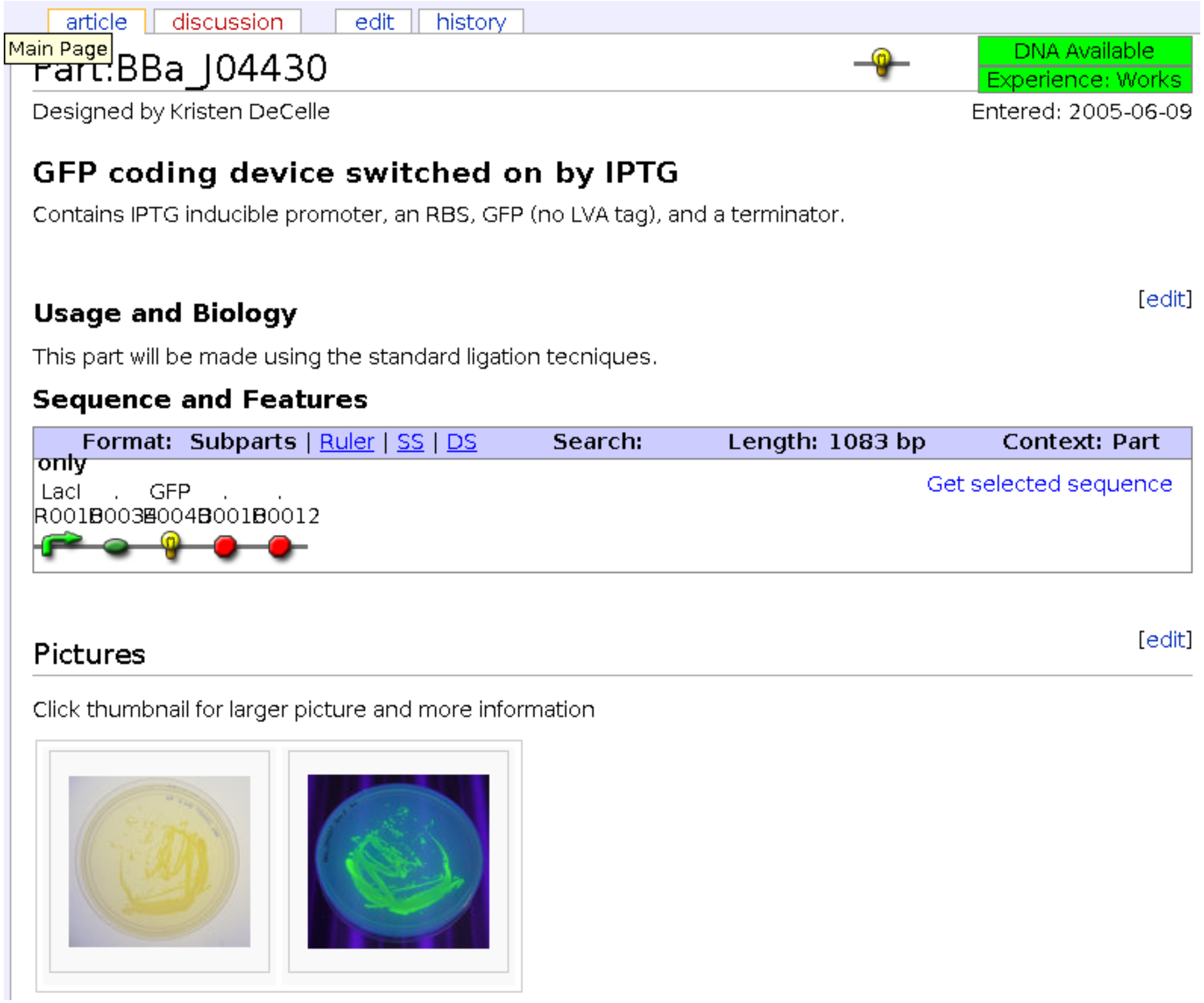
-?-	Name	Description	Tag -?-	Excitation	Output	Length

Available other reporters *Edit*

-?-	Name	Description	Tag -?-	Excitation	Output	Length
A W	BBa_E0241	PoPS to GFP converter	None		Green	795
A W	BBa_E0430	EYFP (RBS+ LVA- TERM) (B0034.E0030.B0015)	None		Yellow	878
A W	BBa_E0840	B0030.E0040.B0015	None		Green	878
A W	BBa_E7104	GFP Reporter Device for Dedicated Transcription Systems	None		Green	826
A W	BBa_I13601	Lac operator with CFP reporter (without LVA tag) [R/Lc-]	None		cyan	940
A W	BBa_I13607	Reporter Construct (Ly+Tc+)	LVA		cyan, yellow	1966
A W	BBa_I15016	B0032.ECFP	None		cyan	742

A device





Part:BBa_J04430

DNA Available
Experience: Works

Designed by Kristen DeCelle

Entered: 2005-06-09

GFP coding device switched on by IPTG

Contains IPTG inducible promoter, an RBS, GFP (no LVA tag), and a terminator.

Usage and Biology

[edit]

This part will be made using the standard ligation tecniques.

Sequence and Features

Format: Subparts | Ruler | SS | DS Search: Length: 1083 bp Context: Part only

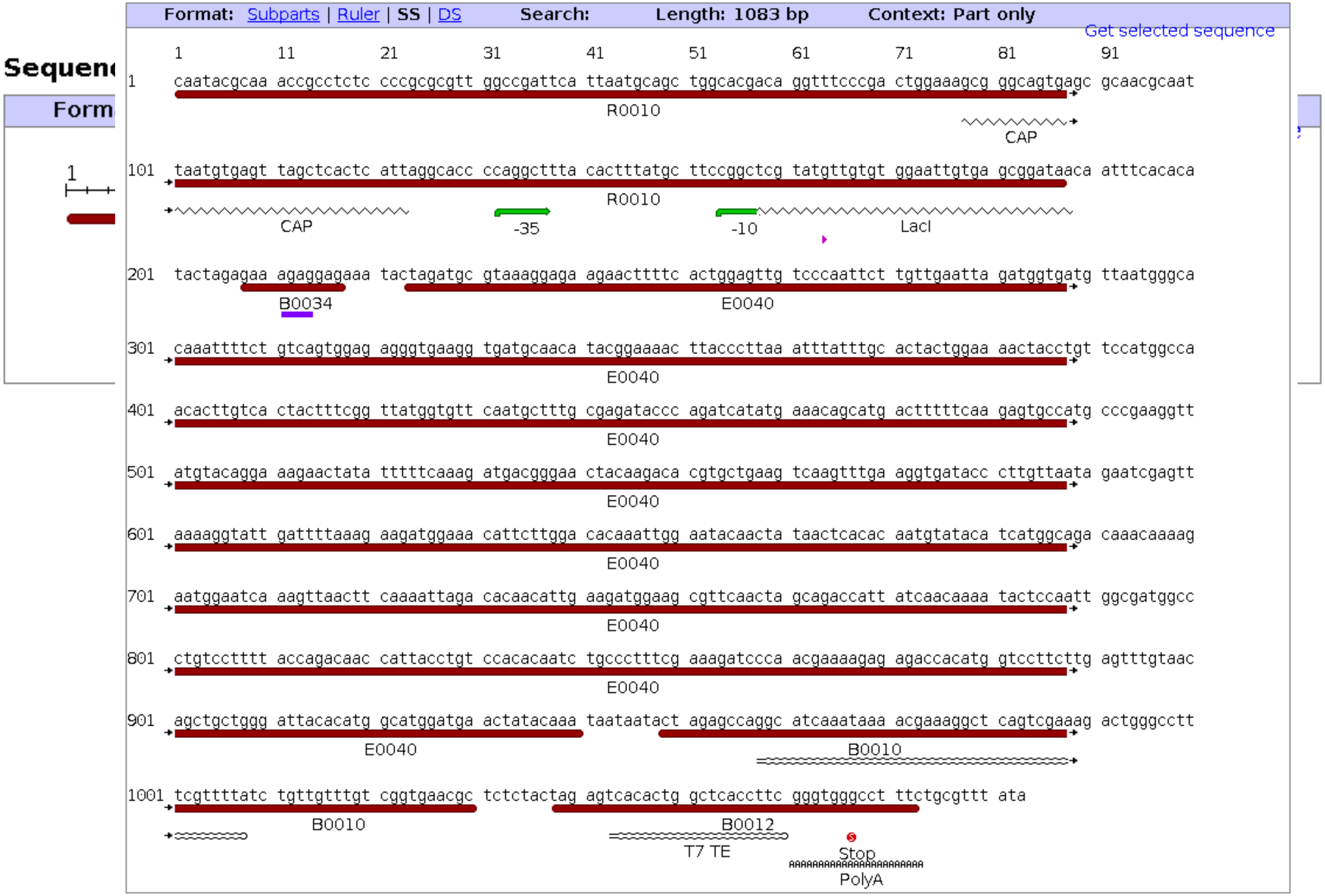
Get selected sequence

Pictures

[edit]

Click thumbnail for larger picture and more information

Un *device*



Format: Subparts | Ruler | SS | DS   Search: Length: 1083 bp   Context: Part only

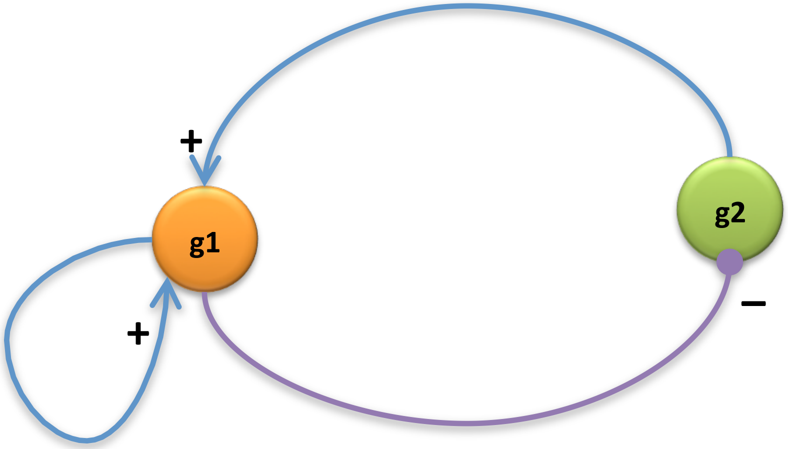
Get selected sequence

```
     1          11         21         31         41         51         61         71         81         91
1    caatacgcaa accgcctctc cccgcgcgtt ggccgattca ttaatgcagc tggcacgaca ggtttcccga ctggaaagcg ggcagtgagc gcaacgcaat
101  taatgtgagt tagctcactc attaggcacc ccaggcttta cactttatgc ttccggctcg tatgttgtgt ggaattgtga gcggataaca atttcacaca
201  tactagagaa agaggagaaa tactagatgc gtaaaggaga agaacttttc actggagttg tcccaattct tgttgaatta gatggtgatg ttaatgggca
301  caaattttct gtcagtggag agggtgaagg tgatgcaaca tacggaaaac ttacccttaa atttatttgc actactggaa aactacctgt tccatggcca
401  acacttgtca ctactttcgg ttatggtgtt caatgctttg cgagataccc agatcatatg aaacagcatg actttttcaa gagtgccatg cccgaaggtt
501  atgtacagga aagaactata tttttcaaag atgacgggaa ctacaagaca cgtgctgaag tcaagtttga aggtgatacc cttgttaata gaatcgagtt
601  aaaaggtatt gattttaaag aagatggaaa cattcttgga cacaaattgg aatacaacta taactcacac aatgtataca tcatggcaga caaacaaaag
701  aatggaatca aagttaactt caaaattaga cacaacattg aagatggaag cgttcaacta gcagaccatt atcaacaaaa tactccaatt ggcgatggcc
801  ctgtcctttt accagacaac cattacctgt ccacacaatc tgccctttcg aaagatccca acgaaaagag agaccacatg gtccttcttg agtttgtaac
901  agctgctggg attacacatg gcatggatga actatacaaa taataatact agagccaggc atcaaataaa acgaaaggct cagtcgaaag actgggcctt
1001 tcgttttatc tgttgtttgt cggtgaacgc tctctactag agtcacactg gctcaccttc gggtgggcct ttctgcgttt ata
```

The underlying model

Finding circuits

Basics of genetic regulation



	promoter	start	RBS	protein code	stop	
TGC	CAGGAC	TTGCCA	GGAAG	CCCAGATGCAGTCTTTTACGT	GCCGAAA	C

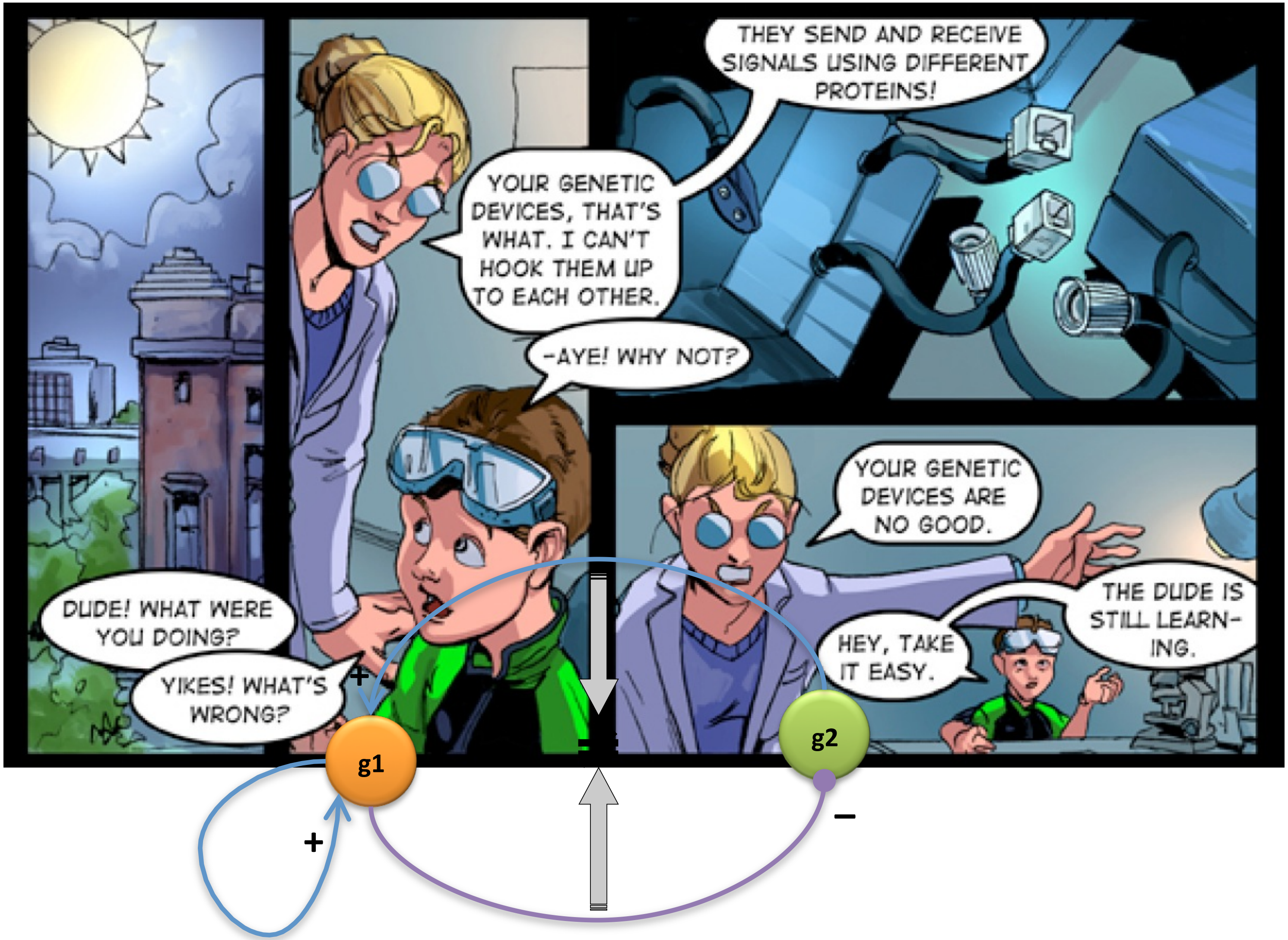
promoter

ribosome binding site

sequence coding for the protein

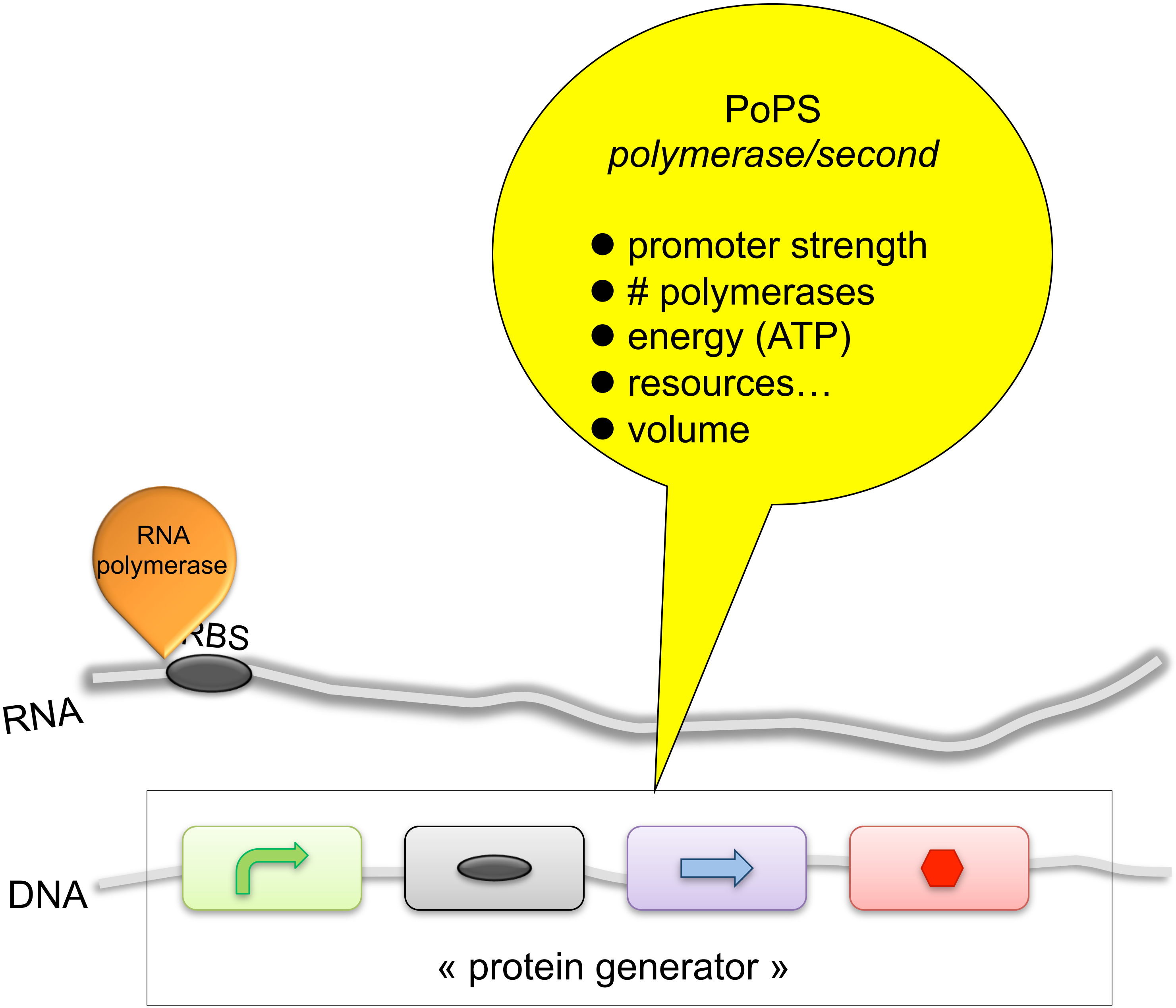
stop codon

Compositionality?


THEY SEND AND RECEIVE SIGNALS USING DIFFERENT PROTEINS!
YOUR GENETIC DEVICES, THAT'S WHAT. I CAN'T HOOK THEM UP TO EACH OTHER.
-AYE! WHY NOT?
YOUR GENETIC DEVICES ARE NO GOOD.
THE DUDE IS STILL LEARN-ING.
HEY, TAKE IT EASY.
DUDE! WHAT WERE YOU DOING?
YIKES! WHAT'S WRONG?
g1
g2
+
+
–

→ compartmentalization

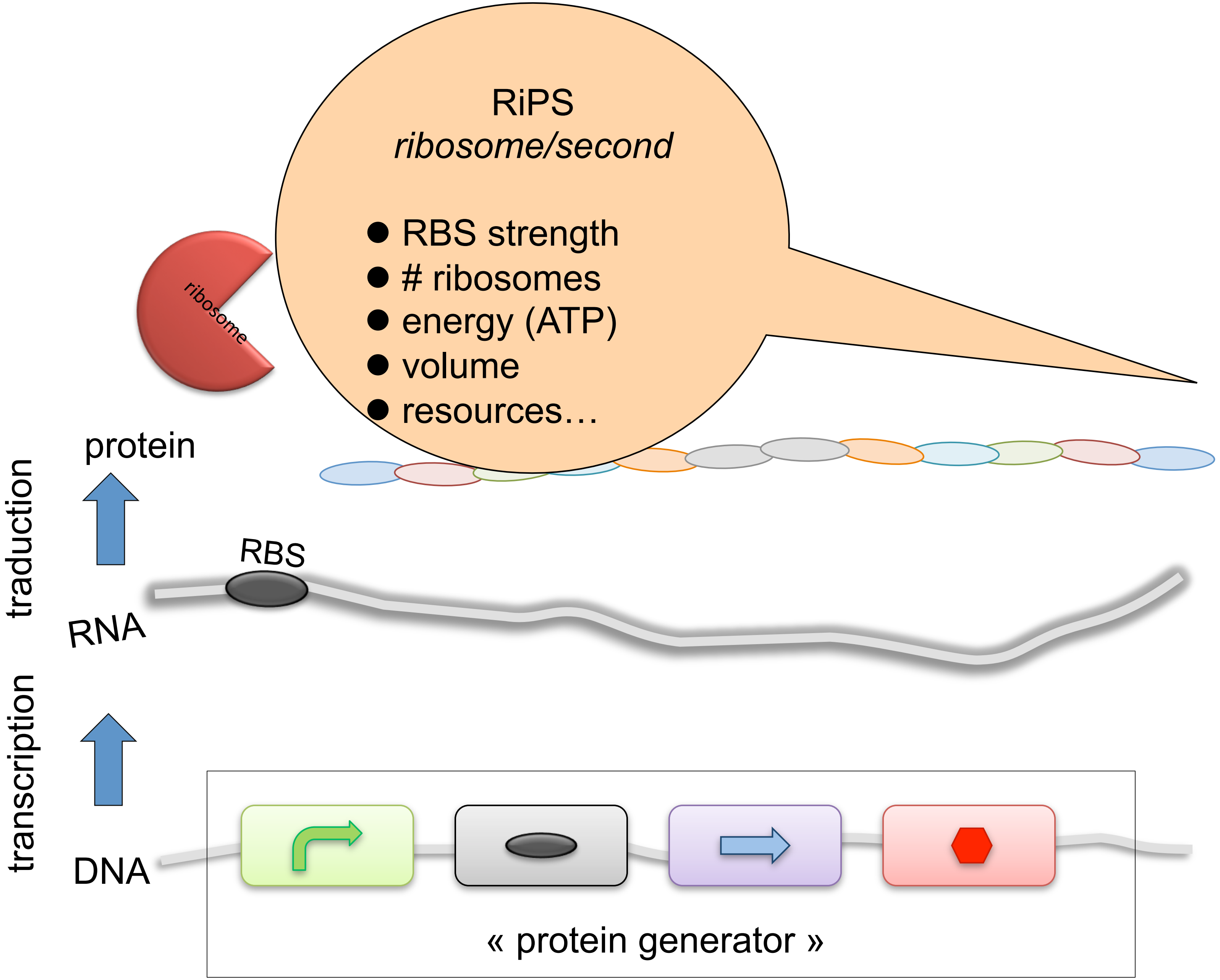
« Signal transmission » in transcription



PoPS
polymerase/second

- promoter strength
- # polymerases
- energy (ATP)
- resources…
- volume

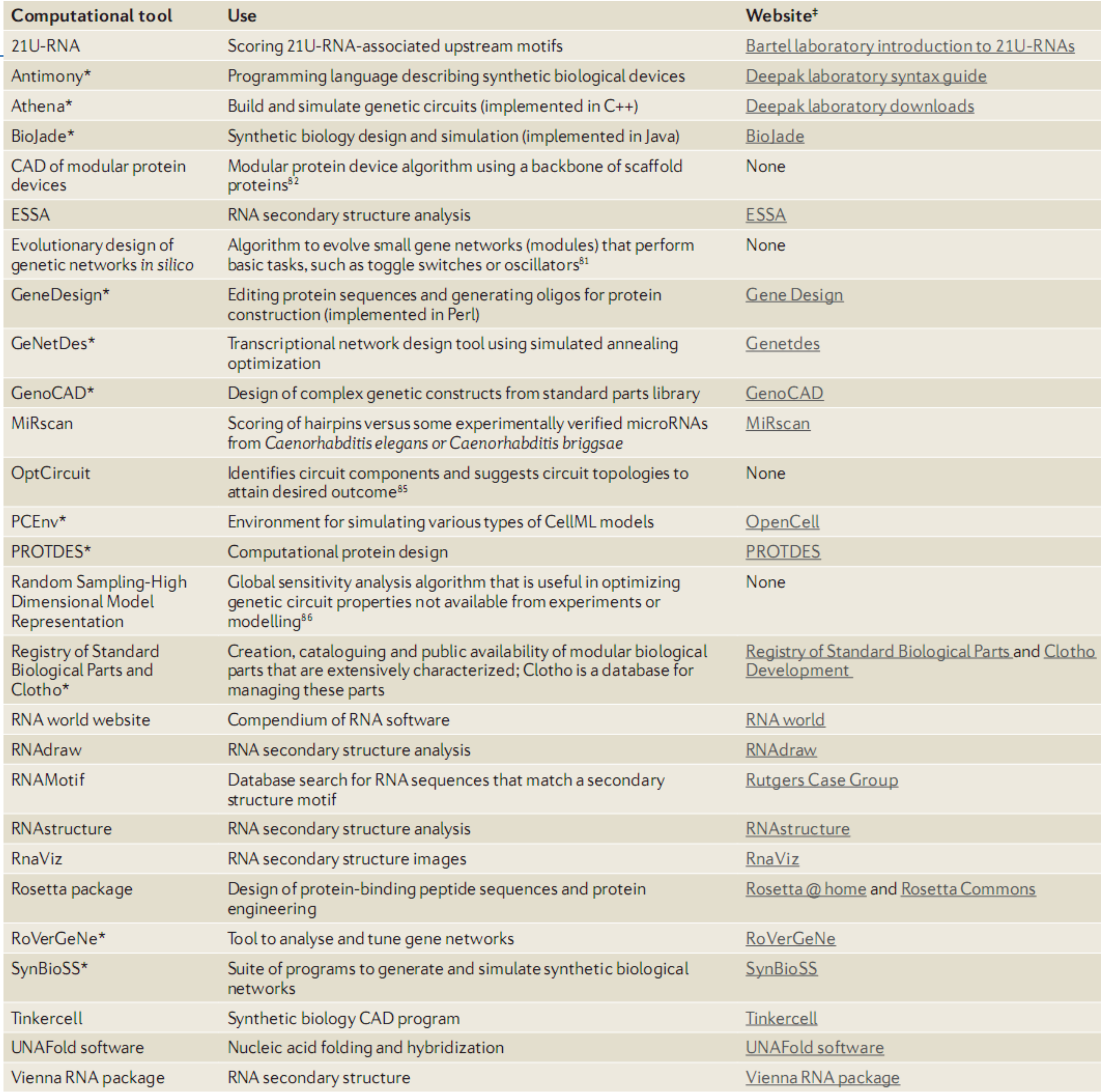
« Signal transmission » in transcription



RiPS
ribosome/second

- RBS strength
- # ribosomes
- energy (ATP)
- volume
- resources…

« protein generator »



Computational tool	Use	Website‡
21U-RNA	Scoring 21U-RNA-associated upstream motifs	Bartel laboratory introduction to 21U-RNAs
Antimony*	Programming language describing synthetic biological devices	Deepak laboratory syntax guide
Athena*	Build and simulate genetic circuits (implemented in C++)	Deepak laboratory downloads
BioJade*	Synthetic biology design and simulation (implemented in Java)	BioJade
CAD of modular protein devices	Modular protein device algorithm using a backbone of scaffold proteins[82]	None
ESSA	RNA secondary structure analysis	ESSA
Evolutionary design of genetic networks *in silico*	Algorithm to evolve small gene networks (modules) that perform basic tasks, such as toggle switches or oscillators[81]	None
GeneDesign*	Editing protein sequences and generating oligos for protein construction (implemented in Perl)	Gene Design
GeNetDes*	Transcriptional network design tool using simulated annealing optimization	Genetdes
GenoCAD*	Design of complex genetic constructs from standard parts library	GenoCAD
MiRscan	Scoring of hairpins versus some experimentally verified microRNAs from *Caenorhabditis elegans or Caenorhabditis briggsae*	MiRscan
OptCircuit	Identifies circuit components and suggests circuit topologies to attain desired outcome[85]	None
PCEnv*	Environment for simulating various types of CellML models	OpenCell
PROTDES*	Computational protein design	PROTDES
Random Sampling-High Dimensional Model Representation	Global sensitivity analysis algorithm that is useful in optimizing genetic circuit properties not available from experiments or modelling[86]	None
Registry of Standard Biological Parts and Clotho*	Creation, cataloguing and public availability of modular biological parts that are extensively characterized; Clotho is a database for managing these parts	Registry of Standard Biological Parts and Clotho Development
RNA world website	Compendium of RNA software	RNA world
RNAdraw	RNA secondary structure analysis	RNAdraw
RNAMotif	Database search for RNA sequences that match a secondary structure motif	Rutgers Case Group
RNAstructure	RNA secondary structure analysis	RNAstructure
RnaViz	RNA secondary structure images	RnaViz
Rosetta package	Design of protein-binding peptide sequences and protein engineering	Rosetta @ home and Rosetta Commons
RoVerGeNe*	Tool to analyse and tune gene networks	RoVerGeNe
SynBioSS*	Suite of programs to generate and simulate synthetic biological networks	SynBioSS
Tinkercell	Synthetic biology CAD program	Tinkercell
UNAFold software	Nucleic acid folding and hybridization	UNAFold software
Vienna RNA package	RNA secondary structure	Vienna RNA package

Purnt & Weiss, Nature Reviews, June 2009.

Analysis & Design of Fluid Machines as Non Conventionnal Programing

What is the "VHDL"
for molecular interaction network
and beyond?

The general picture

① Design requires a model

synthetic biology as the constructive part of systems biology

a model is a program in some programing language

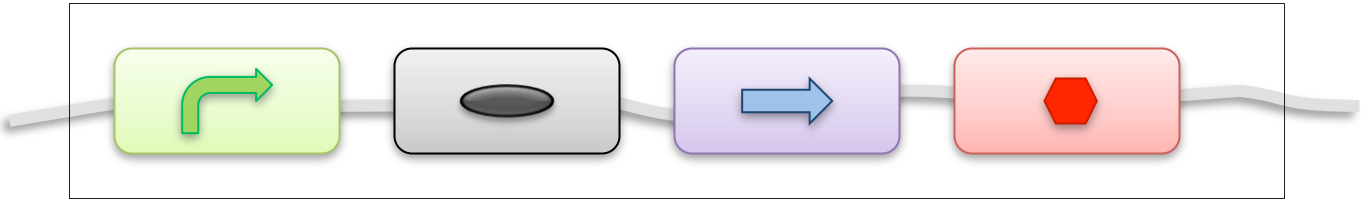
② Abstraction level: alternation of

- deterministic / stochastic
  (noise, chemical fluctuation, etc.)
- continuous / discrete
  aggregate versus agent-based models

③ The structure of fluid machines changes in time :

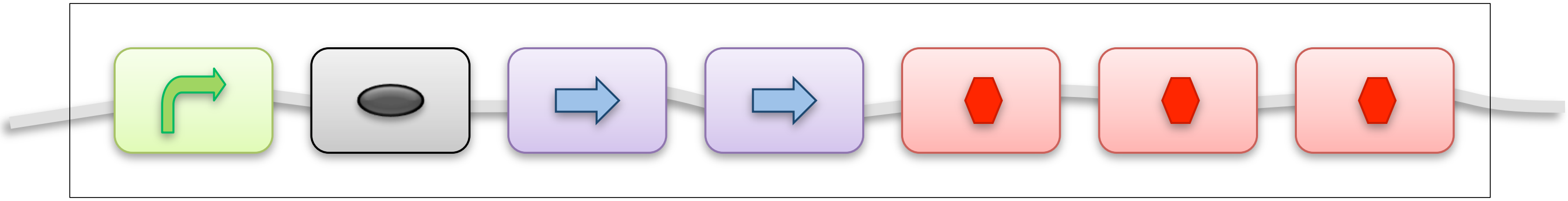
- dynamical systems with a dynamic structure (DS)^2
- Think globally, program locally:
  - datasheet (static) *vs.* rules set
  - space matters

① A programing language for SB



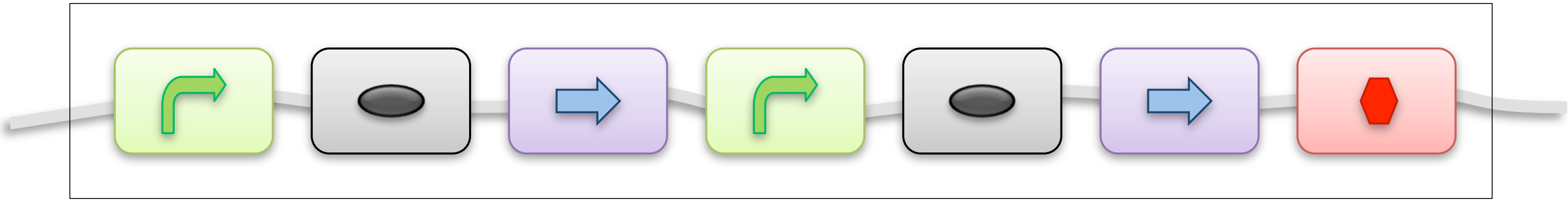
Valide or not?

① A programing language for SB



Valide or not?

① A programing language for SB



Valide or not?

① A programing language for SB

Can we restrict the construction

- a priori (syntax/type discipline)
- a posteriori (verification, validation, test)

Such that compositions of biobricks:

- have the expected functional properties
- make an autonomous system (modularity)
- Can themselves be composed (reuse)

➔ This are problems addressed in a programming language

example : register allocation by a compiler and choice of genes

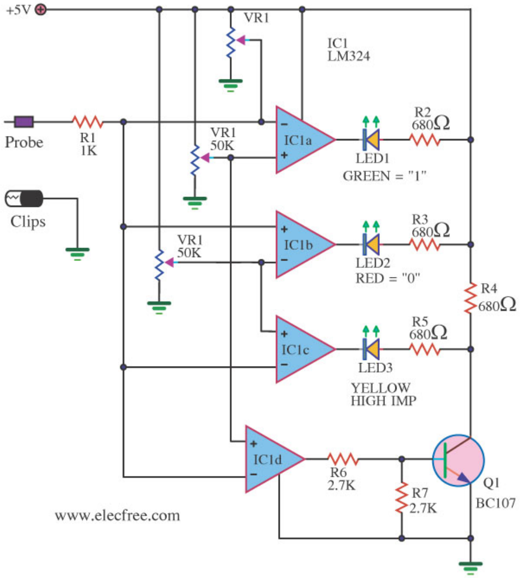
Is this language a “Biobricks composition language” ?

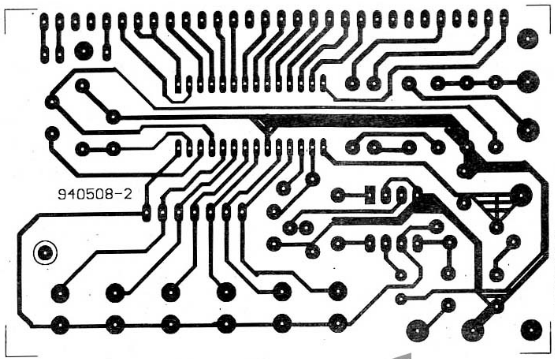
A programing language for SB

The composition of biobricks does not address all problems:

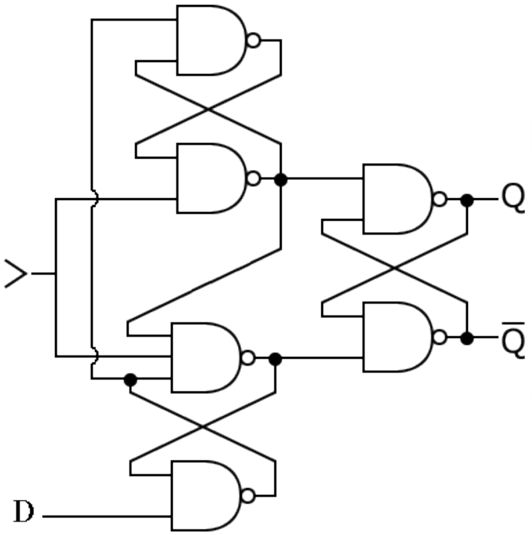
- Which semantics for/characterization of the biobricks ?
- Cell *vs.* population
- Bioprocesses as a special kind of dynamical systems

② A tower of languages





electronic
(electric currents)
[spice]



logic
(gates)
[sequential logic]

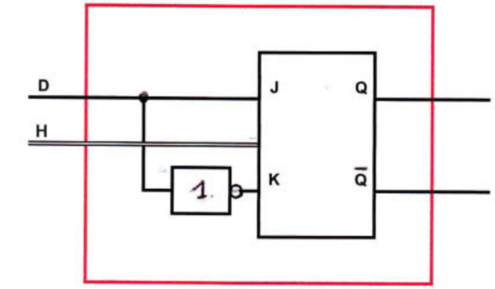


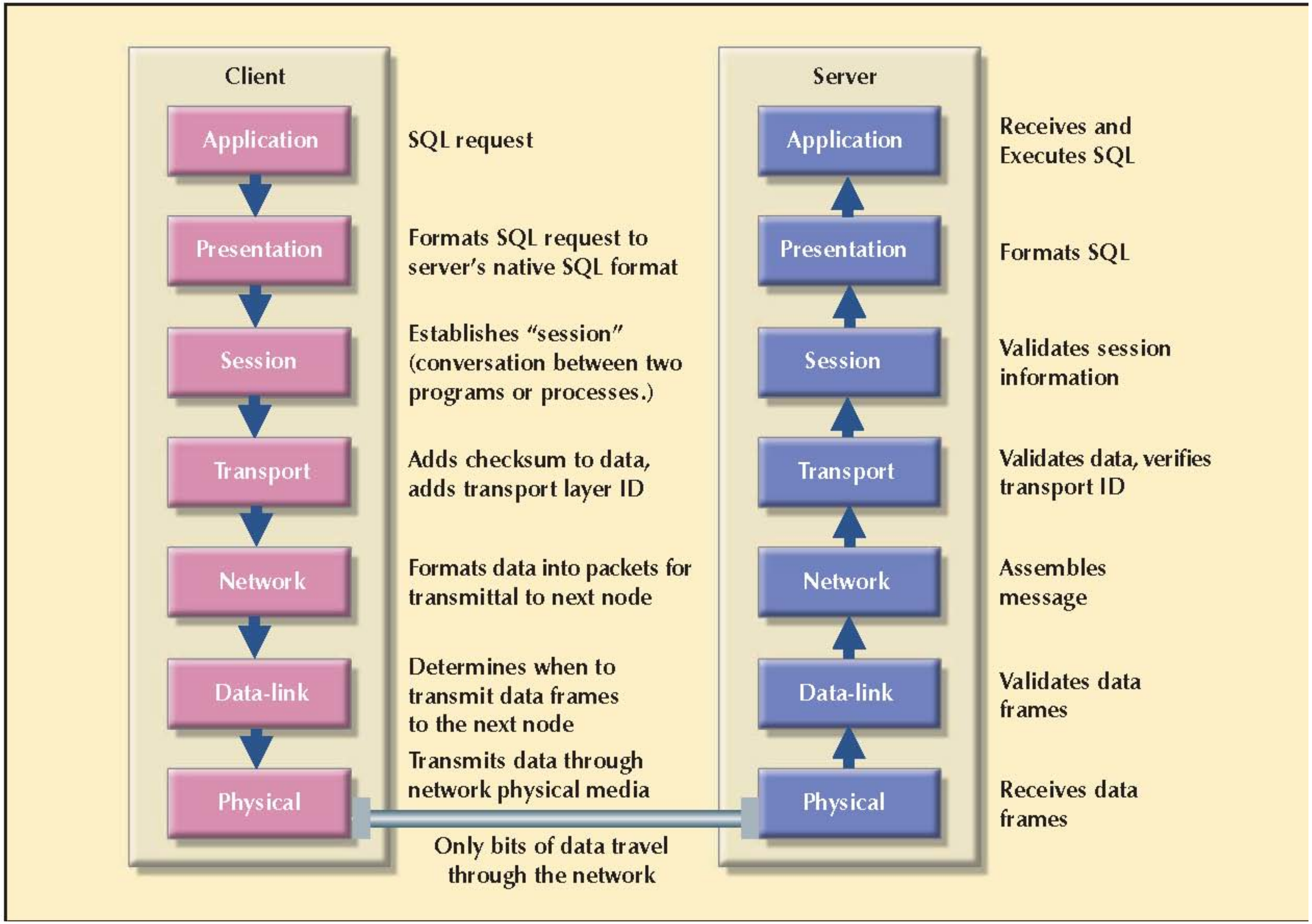
Table d'états

H	D	Qn+1	$\overline{Qn+1}$
↑	0	0	1
↑	1	1	0
↓	×	Qn	$\overline{Qn}$

function
(register)
[transition system]

② A tower of languages with a lot of levels

FIGURE F.7 Information Flow Through the OSI Model



Client

Server

Application — SQL request — Application — Receives and Executes SQL

Presentation — Formats SQL request to server's native SQL format — Presentation — Formats SQL

Session — Establishes "session" (conversation between two programs or processes.) — Session — Validates session information

Transport — Adds checksum to data, adds transport layer ID — Transport — Validates data, verifies transport ID

Network — Formats data into packets for transmittal to next node — Network — Assembles message

Data-link — Determines when to transmit data frames to the next node — Data-link — Validates data frames

Physical — Transmits data through network physical media — Physical — Receives data frames

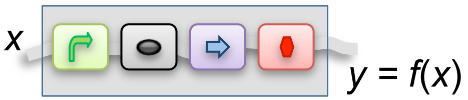
Only bits of data travel through the network

Vertical abstraction *or*
horizontal compositionality **?**

Systems

Long term
Applications

Devices



Assembly of *biobricks*
implementing a function
(inverters, oscillateurs, switch, …)

Parts

BIOBRICKS

DNA

ACGTTGCCAGGACCTTGCCAGGAA
ACCTGCCAGGAATGCAGTCCCTTTT

Sequences of nucleotides

② What are the good abstractions?

The composition of biobricks does not address all problems:

Which semantics for the biobricks ?

- Chemical interactions are fundamentally stochastic
  ➔ stochastic modeling
  ➔ amorphous computation

- Some bioprocesses implies only few entities:
  ➔ discrete, agent-based modeling

- Some processes are aggregation of a lot of lower level processes
  ➔ continuous modeling

③ What are the good abstractions?

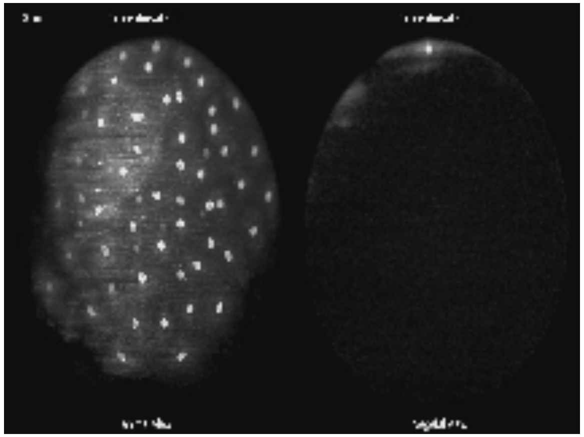
The composition of biobricks does not address all problems:

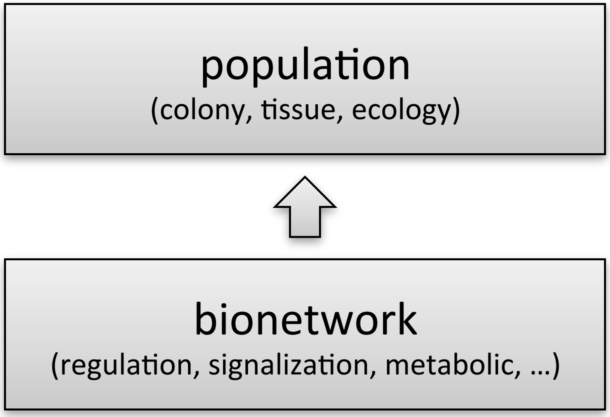
- *cell vs. population*: colony, biofilm, tissue, organism, ecology

→ global behavior vs. local behavior

→ spatial abstraction

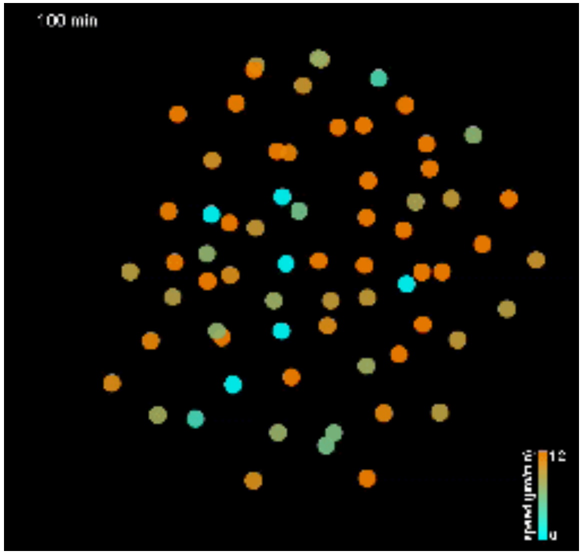
→ morphogenesis





population
(colony, tissue, ecology)

bionetwork
(regulation, signalization, metabolic, …)



Nadine Peyrieras

③ What are the good abstractions?

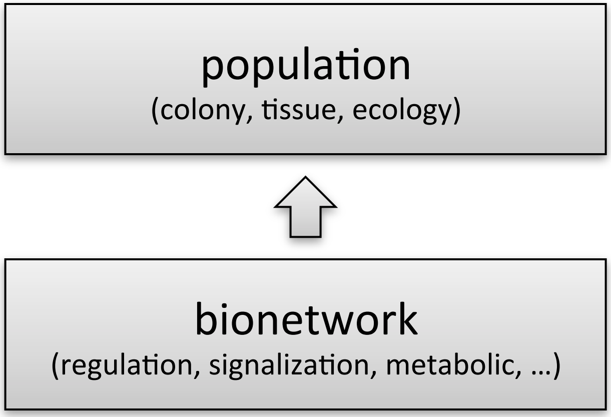
The composition of biobricks does not address all problems:

- *cell vs. population*: colony, biofilm, tissue, organism, ecology

→ global behavior vs. local behavior

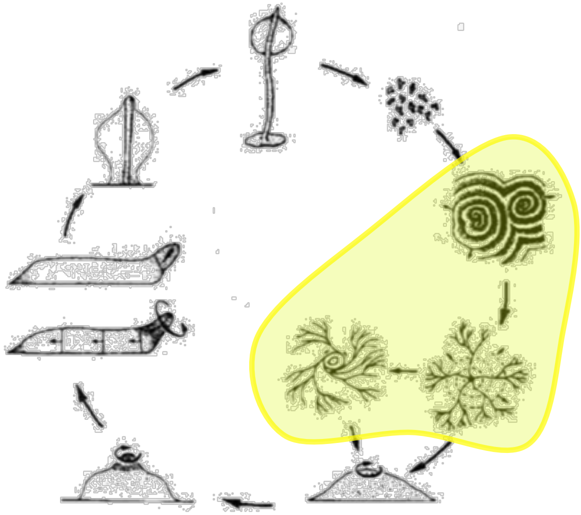
→ spatial abstraction

→ morphogenesis

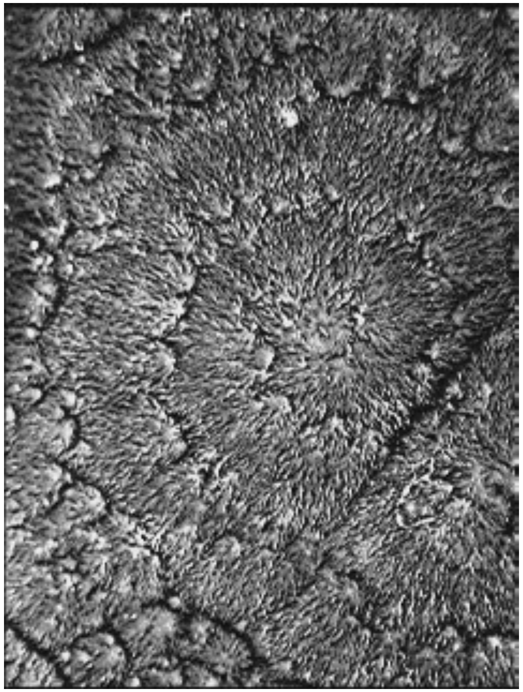


population
(colony, tissue, ecology)

bionetwork
(regulation, signalization, metabolic, …)



Dictyostelium discoideum



Marcus Hauser

http://ecologie.snv.jussieu.fr/eem/berder2009/Presentations_Berder09/

③ What are the good abstractions?

The composition of biobricks does not address all problems:

- *cell vs. population*: colony, biofilm, tissue, organism, ecology

➔ global behavior vs. local behavior

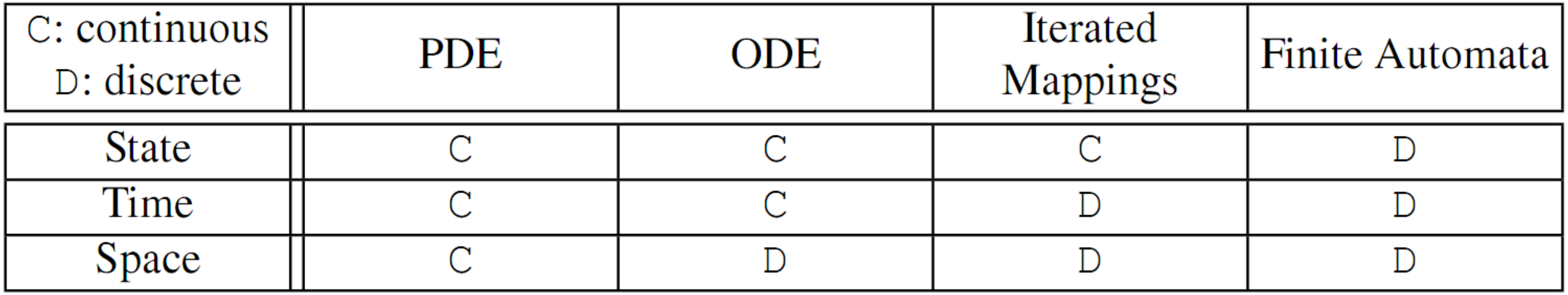
splitting a function over a population of bacteria

- – to reduce the cell overload
- – because some sub-functions are chemically incompatibles
- – because some sub-functions are sequential
- – for resources mapping (compiler)
- – for reuse (modularity at bacteria level)
- – for security (auxotrophic device)
- – etc.

③ What are the good abstractions?

The composition of biobricks does not address all problems:

- *Bioprocesses as dynamical systems*?



C: continuous D: discrete	PDE	ODE	Iterated Mappings	Finite Automata
State	C	C	C	D
Time	C	C	D	D
Space	C	D	D	D

③ What are the good abstractions?

The composition of biobricks does not address all problems:

- *Bioprocesses as dynamical systems*?
  - Entities involved are dynamic and cannot be listed *a priori*
    chemical complex (EGF with receptor network = 10^{33} species)
    cellular duplication, apoptosis, reproduction, …
  - Interactions are localized (compartment)
    (vesicle, cargo, membrane, etc.)
    and compartments are dynamically created

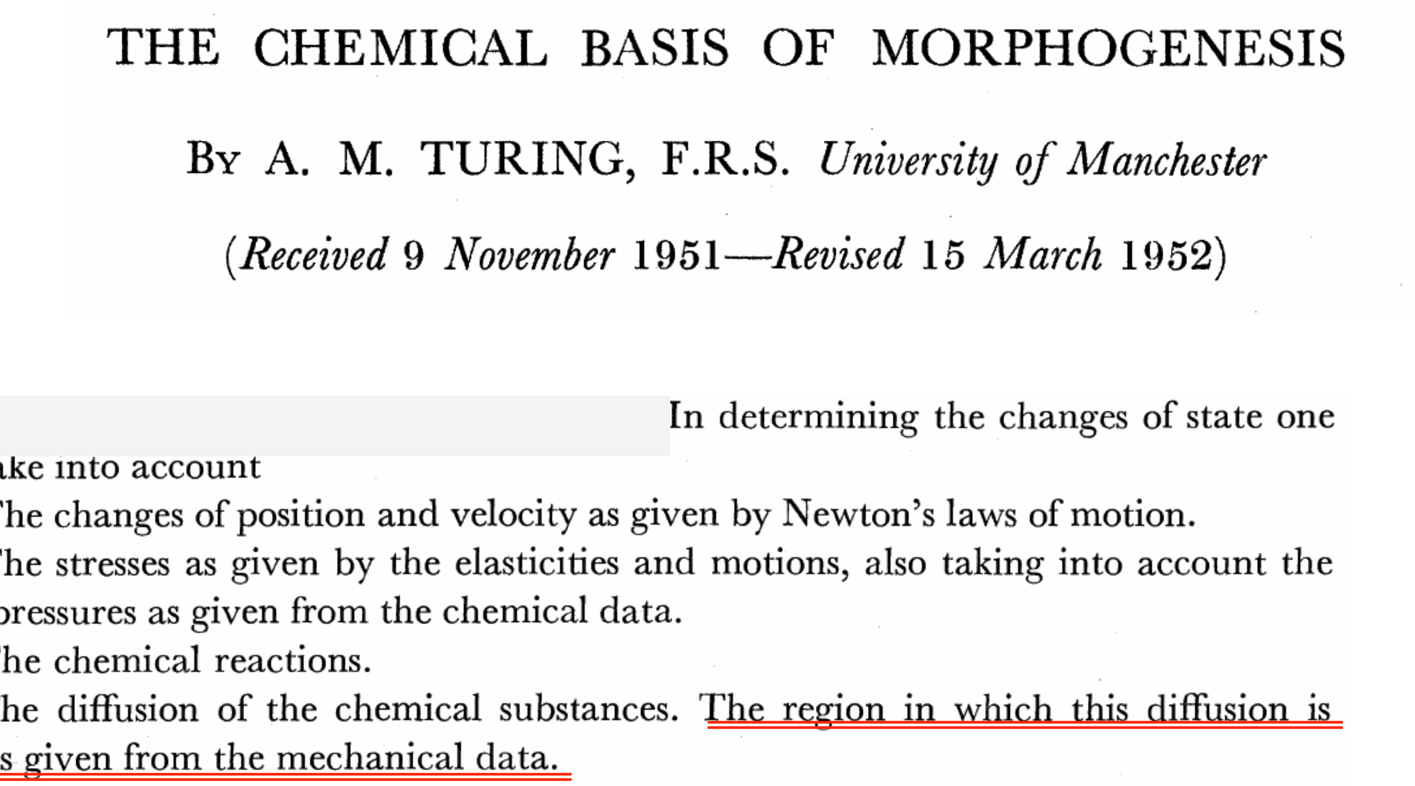
➔Dynamical systems with a dynamic structure :
the state space is build with the process itself

➔ no *a priori* global description

➔ evolution rules must be local

➔ the global evolution is the "integration" of local evolutions

The need for (DS)^2 ?



THE CHEMICAL BASIS OF MORPHOGENESIS

By A. M. TURING, F.R.S. *University of Manchester*

(*Received* 9 *November* 1951—*Revised* 15 *March* 1952)

In determining the changes of state one should take into account

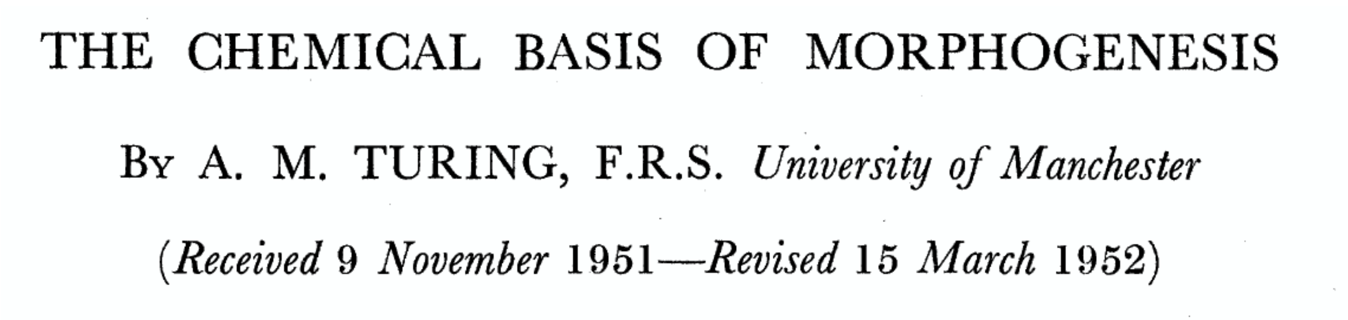
(i) The changes of position and velocity as given by Newton's laws of motion.

(ii) The stresses as given by the elasticities and motions, also taking into account the osmotic pressures as given from the chemical data.

(iii) The chemical reactions.

(iv) The diffusion of the chemical substances. The region in which this diffusion is possible is given from the mechanical data.

The need for (DS)² ?

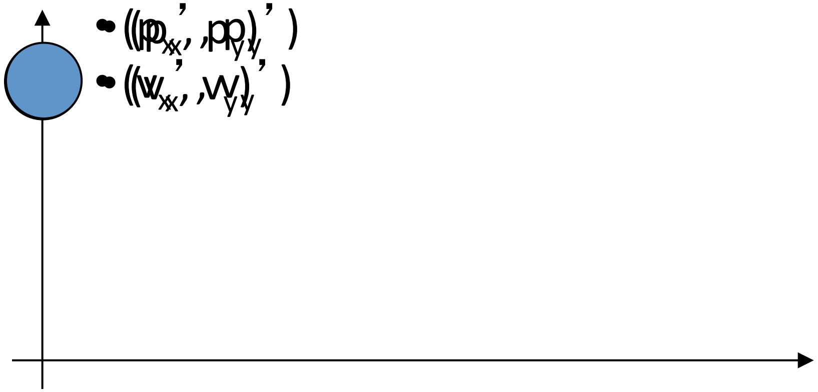


THE CHEMICAL BASIS OF MORPHOGENESIS

By A. M. TURING, F.R.S. *University of Manchester*

(*Received* 9 *November* 1951—*Revised* 15 *March* 1952)

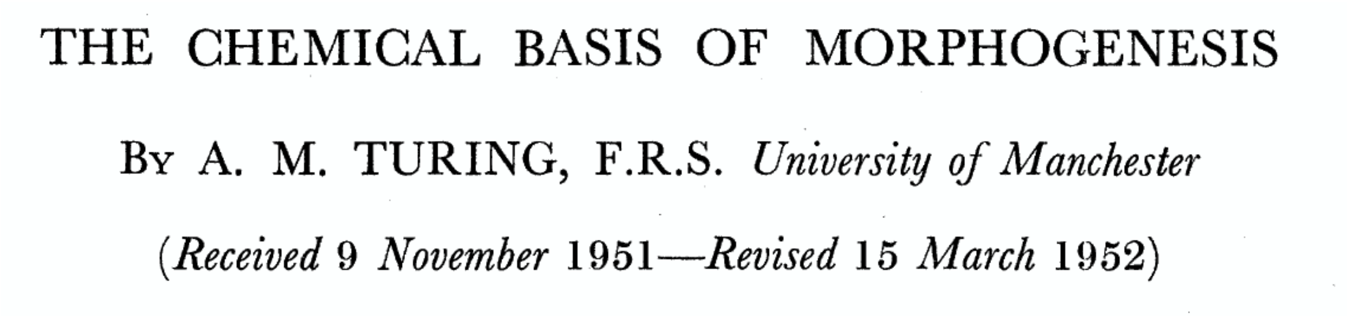
a falling ball



at any time a state is a position and a speed

A dynamical system (DS)

The need for (DS)^2 ?



THE CHEMICAL BASIS OF MORPHOGENESIS

By A. M. TURING, F.R.S. *University of Manchester*

(*Received* 9 *November* 1951—*Revised* 15 *March* 1952)

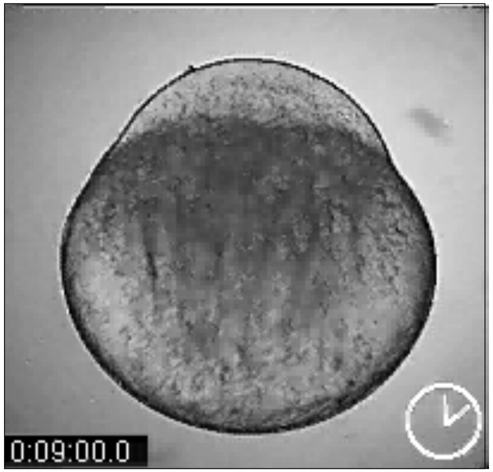
a falling ball

- (p_x' , p_y')
- (v_x' , v_y')

at any time a state is a position and a speed

A dynamical system (DS)

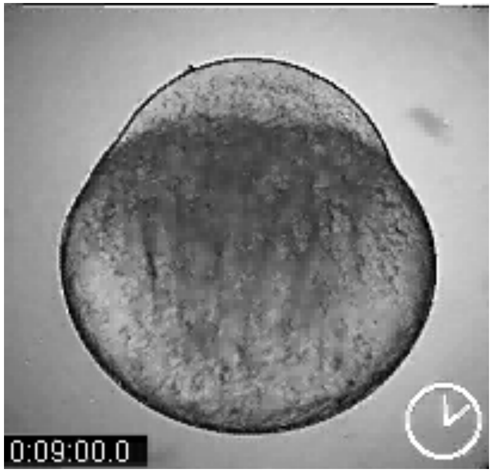
a developing embryo



the structure of the state
(chemical and mechanical state of each cell)
is changing in time

A dynamical system with a dynamical structure (DS)^2

The interplay between state and form

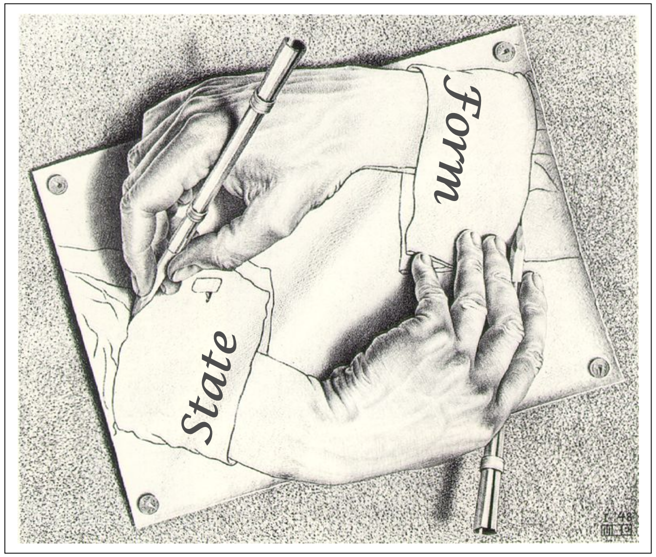


a developing embryo

the state **as well as the structure of the state** is changing in time

*(chemical and mechanical state of each cell **as well as the arrangement of the cells**)*

“Strange loop”



Dynamics ON form

Dynamics OF form

A dynamical system with a dynamical structure

$(DS)^2$

The need for $(DS)^2$?

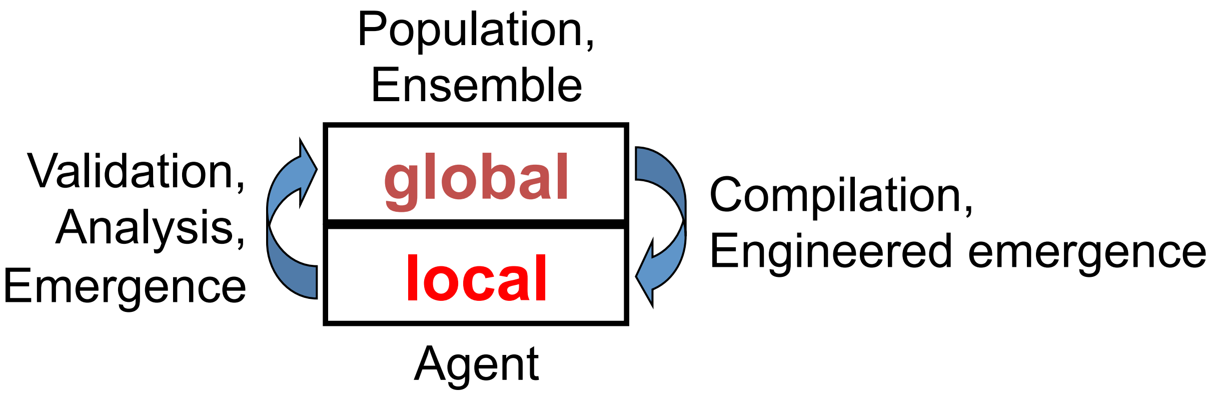
THE CHEMICAL BASIS OF MORPHOGENESIS

By A. M. TURING, F.R.S. *University of Manchester*

(*Received* 9 *November* 1951—*Revised* 15 *March* 1952)

It might be possible, however, to treat a few particular cases in detail with the aid of a digital computer. This method has the advantage that it is not so necessary to make simplifying assumptions as it is when doing a more theoretical type of analysis. It might even be possible to take the mechanical aspects of the problem into account as well as the chemical, when applying this type of method. The essential disadvantage of the method is that one only gets results for particular cases. But this disadvantage is probably of comparatively little importance.

Three other challenges for a BS language



Population, Ensemble

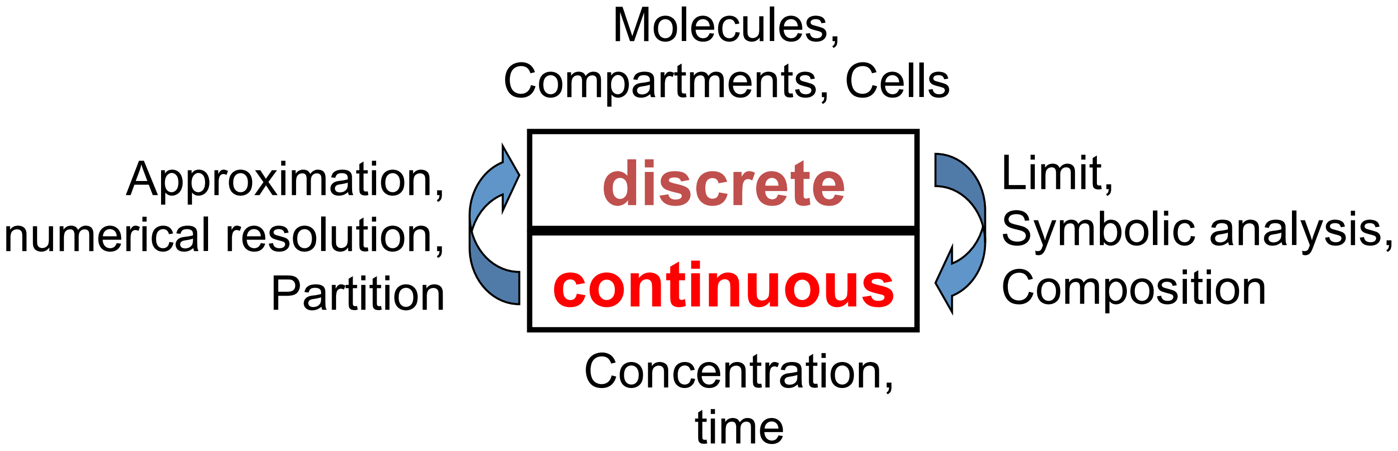
global

local

Agent

Validation, Analysis, Emergence

Compilation, Engineered emergence



Molecules, Compartments, Cells

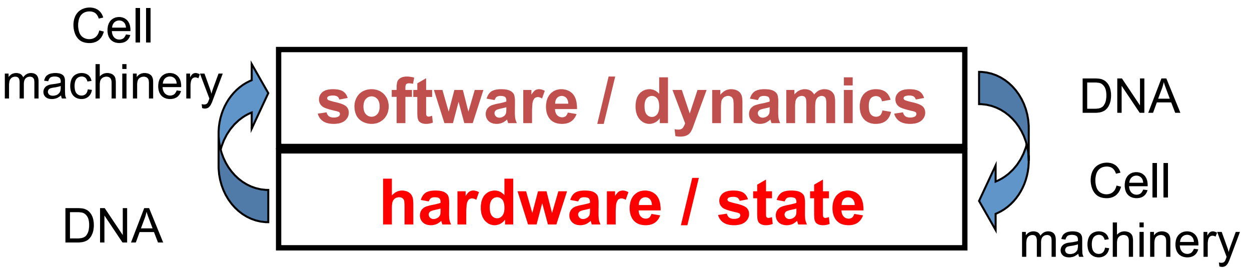
discrete

continuous

Concentration, time

Approximation, numerical resolution, Partition

Limit, Symbolic analysis, Composition



Cell machinery

DNA

software / dynamics

hardware / state

DNA

Cell machinery

The MGS project

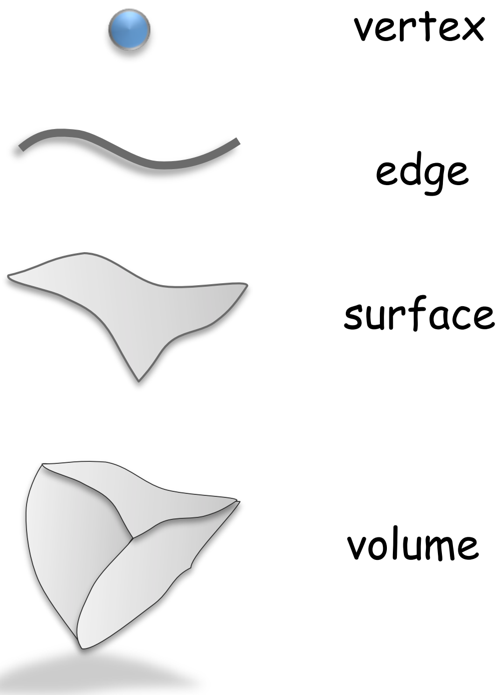
- Language dedicated to the simulation of $(DS)^2$
- Declarative (declarative simulation *vs* procedural)
- Abstract rewriting of complex spatial structures:
  - Data structure = topological collections

    sequence, generalized array, (multi-)set, arbitrary graph, Delaunay triangulation, g-map, …, cell complexes

  - Control structure = transformation
    - two powerful languages to specify sub-collections (elements in interaction)
    - Various rule application strategies: maximal parallel, asynchronous, stochastic, Gillespie-like, …

Topological collections

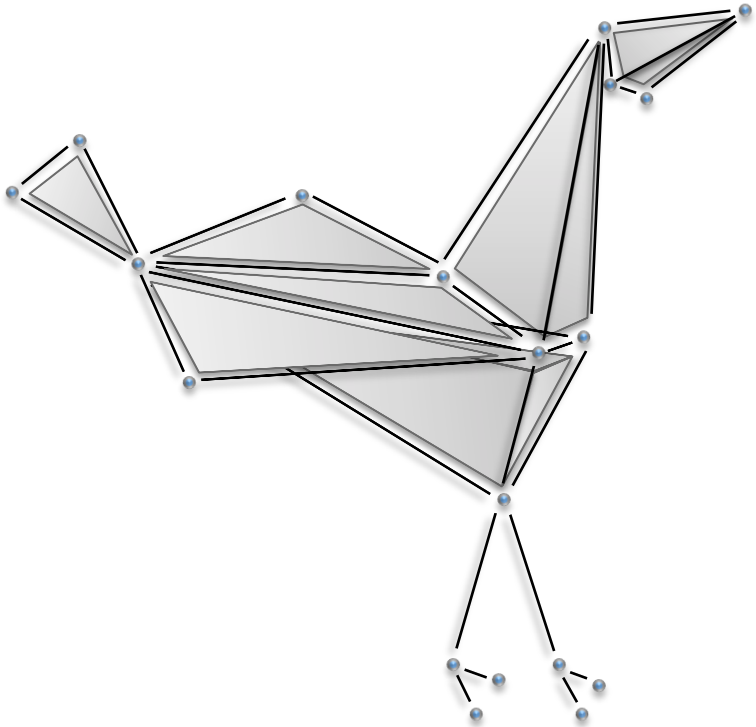
- Structure
  - A collection of topological cells
  - An ***incidence relationship***


vertex

edge

surface

volume

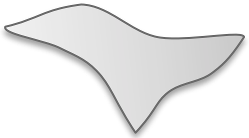


Topological collections

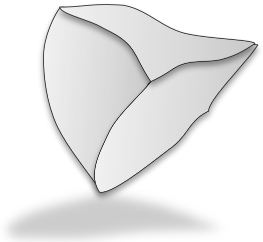
- Structure
  - A collection of topological cells
  - An incidence relationship
  - Data: **association of a value with each cell**

0-cell

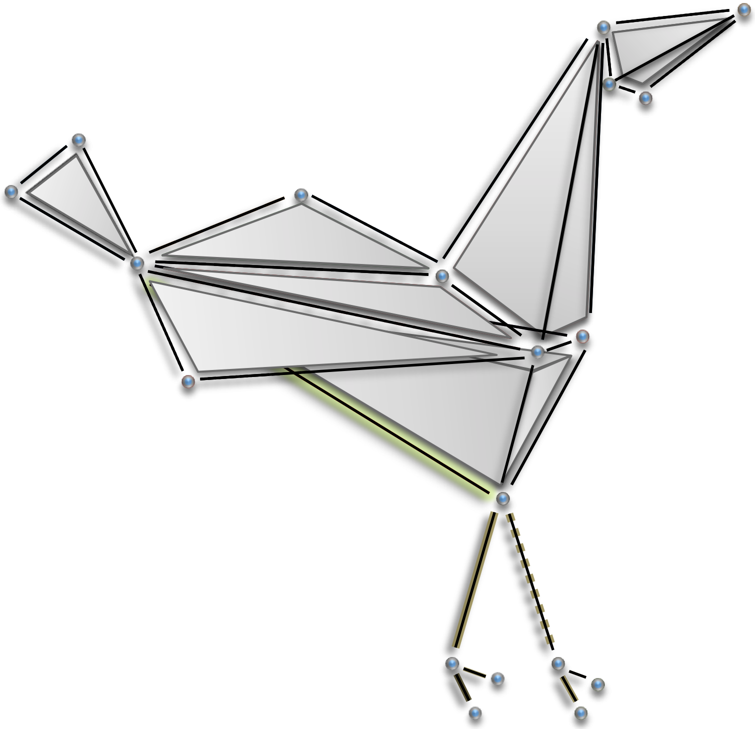
1-cell



2-cell



3-cell



Higher dimensional objects for complex simulations

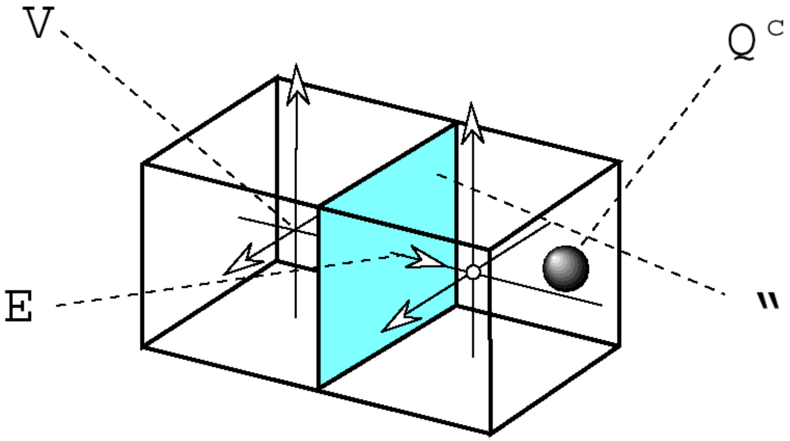
Example of electrostatic Gauss law [**Tonti** 74]

- Electric charge content ρ : dimension 3
- Electric flux $\boldsymbol{\Phi}$: dimension 2
- Law available on a arbitrary complex domain

$$\phi = \oiint w \cdot dS = \frac{Q^c}{\varepsilon_0} = \iiint_{(V)} \frac{\rho}{\varepsilon_0} d\tau$$

electric field in space:

- V: electric potential (dim 0)
- E: voltage (dim 1)
- w: electric flux (dim 2)
- Qc: electric charge (dim 3)



A Direct Discrete Formulation of Field Laws: The Cell Method

Transformations

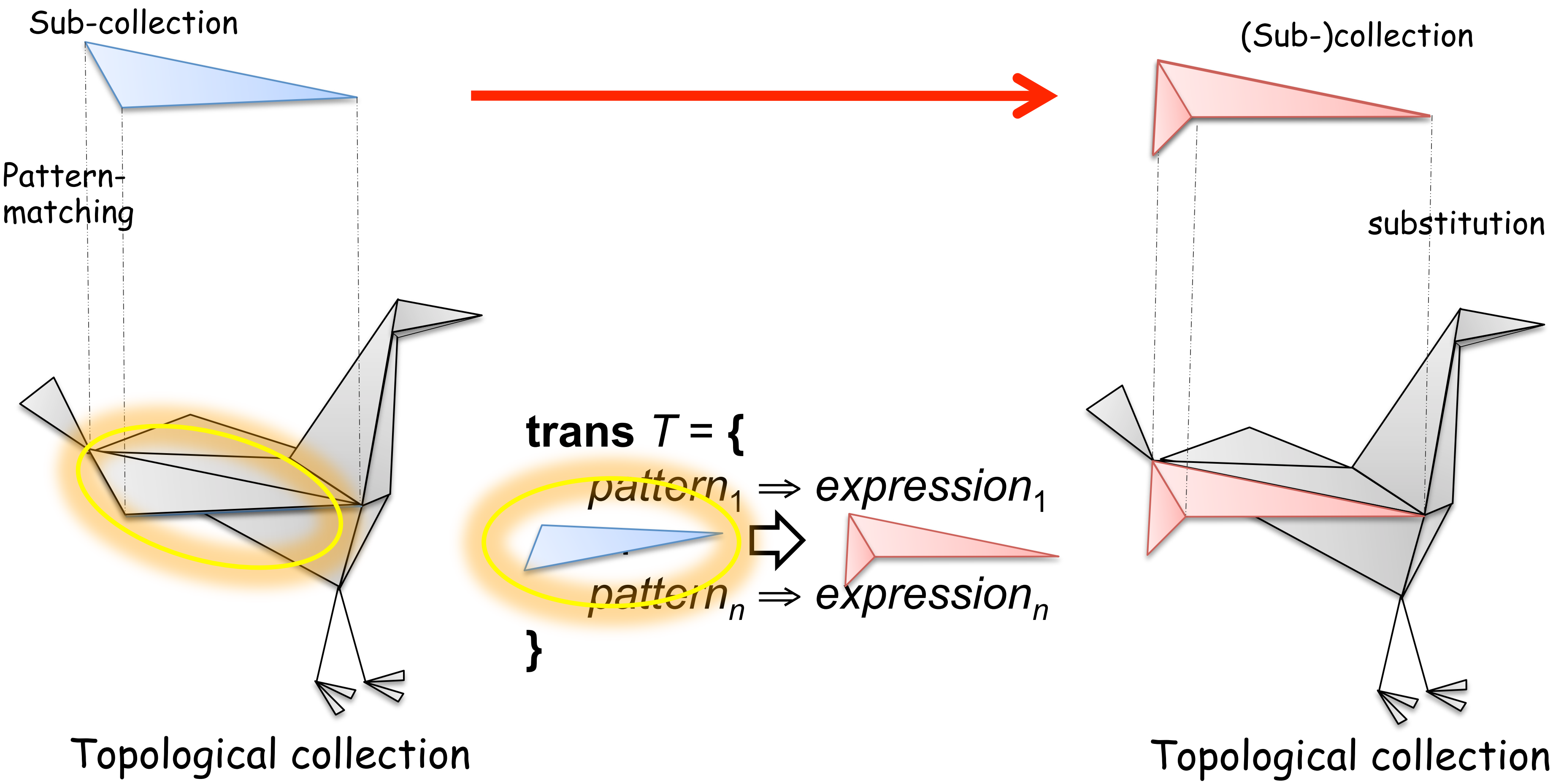
- Functions defined by case on collections

  Each case (pattern) matches a sub-collection

- Defining a rewriting relationship: ***topological rewriting***

```
trans $T$ = {
    $pattern_1 \Rightarrow expression_1$
    …
    $pattern_n \Rightarrow expression_n$
}
```

Transformations



trans *T* = {

$\mathit{pattern}_1 \Rightarrow \mathit{expression}_1$

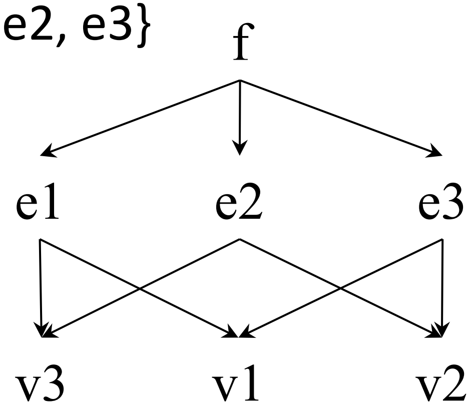
$\mathit{pattern}_n \Rightarrow \mathit{expression}_n$

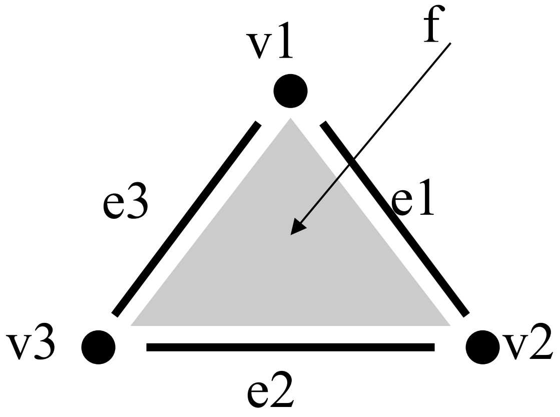
}

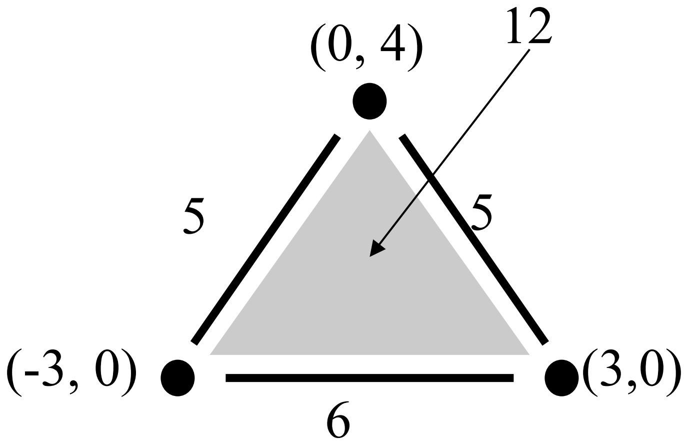
Abstract Simplicial Complex and simplicial chains

Incidence relationship and lattice of incidence:

- boundary(f) = {v1, v2, v3, e1, e2, e3}
- faces(f) = {e1, e2, e3}
- cofaces(v1) = {e1, e3}







Topological chain

- coordinates with vertices
- lengths with edges
- area with f

$$\begin{pmatrix}0\\4\end{pmatrix}.v_1 + \begin{pmatrix}3\\0\end{pmatrix}.v_2 + \begin{pmatrix}-3\\0\end{pmatrix}.v_3 + 5.e_1 + 6.e_2 + 5.e_3 + 12.f$$

Topological rewriting = transformation

$1 + 2 \rightarrow \ldots$ (arithmetic) term rewriting

arithmetic operation

$a \,.\, b \rightarrow \ldots$ string rewriting (~ L systems)

string concatenation: « . » is a formal associative operation

$2H + O \rightarrow H_2O$ multiset rewriting (~ chemistry)

multiset concatenation (= the chemical soup): « . » is AC

$v_1.\sigma_1 + v_2.\sigma_2 \rightarrow \ldots$ topological rewriting (MGS)

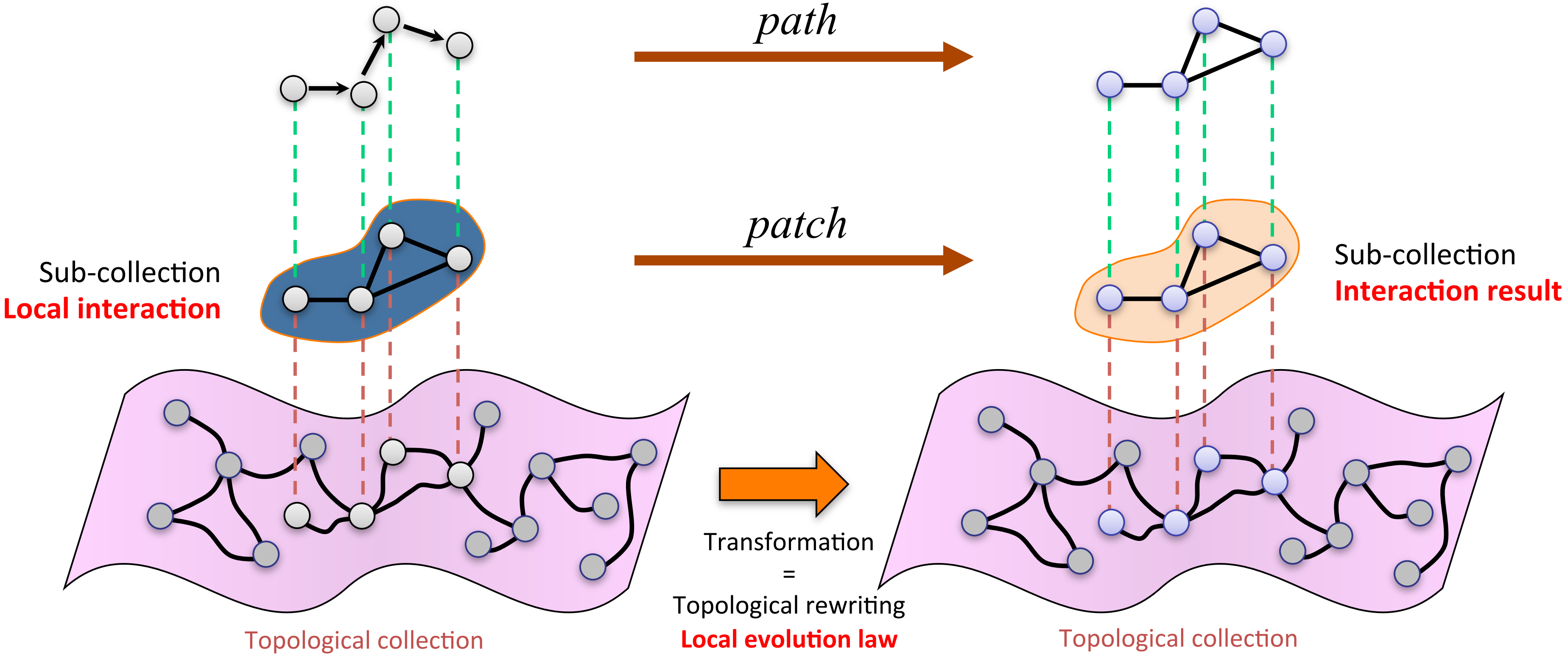
gluing cell in a cell complex: … (AC and algebraic machinery)

Transformation

Pattern matching : specifying a sub-collection of elements in interaction

- *Path transformation* (path = sequence of neighbor elements)
  - Concise but limited expressiveness
- *Patch transformation* (arbitrary shape)
  - Longer but higher expressiveness

Rule application strategy: maximal parallel, asynchronous, stochastic, etc.

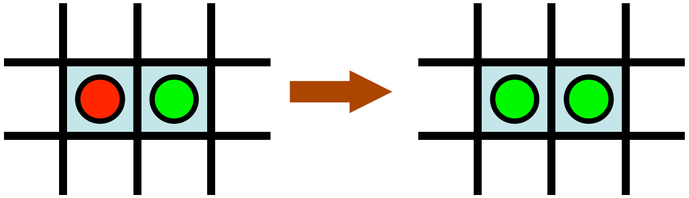


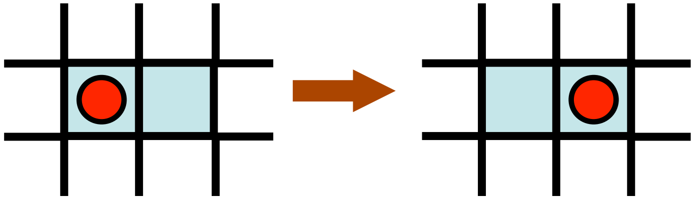
Example: Diffusion Limited Aggregation (DLA)

- Diffusion: some particles are randomly diffusing; others are fixed
- Aggregation: if a mobile particle meets a fixed one, it stays fixed

```
trans dla = {
      `mobile , `fixed  => `fixed, `fixed ;
      `mobile , <undef> => <undef>, `mobile
}
```

NEIGHBOR OF



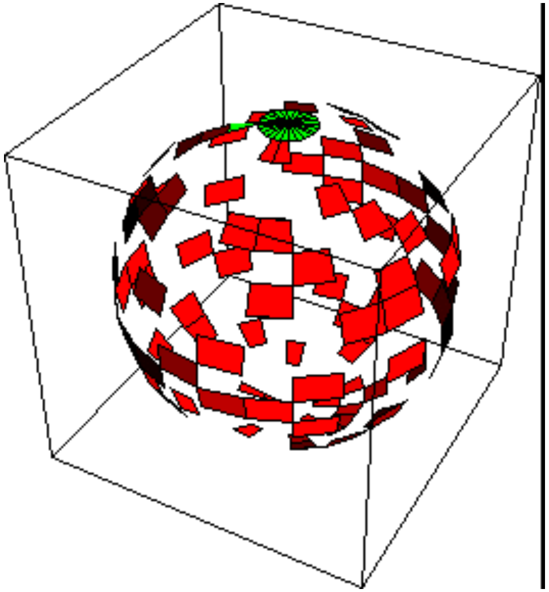


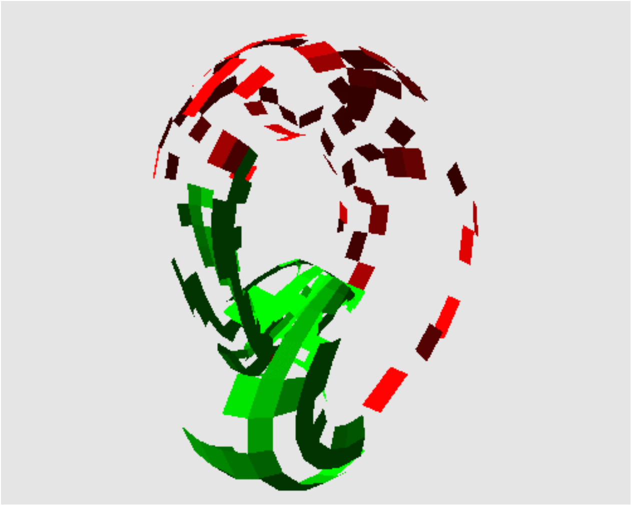
Example: Diffusion Limited Aggregation (DLA)

- Diffusion: some particles are randomly diffusing; others are fixed
- Aggregation: if a mobile particle meets a fixed one, it stays fixed

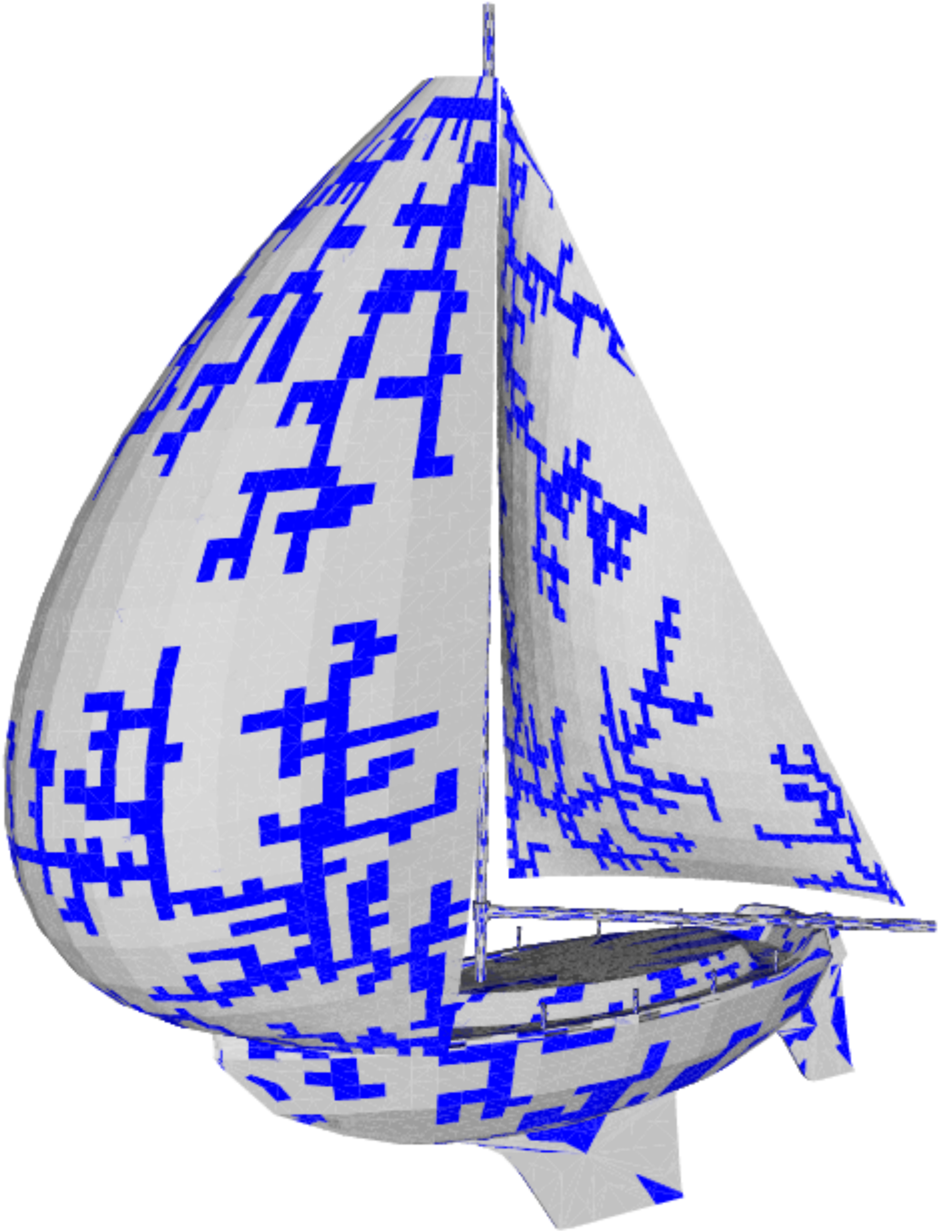
```
trans dla = {
    `mobile , `fixed   => `fixed, `fixed ;
    `mobile , <undef> => <undef>, `mobile
}
```

this transformation is an abstract process that can be applied to any kind of space

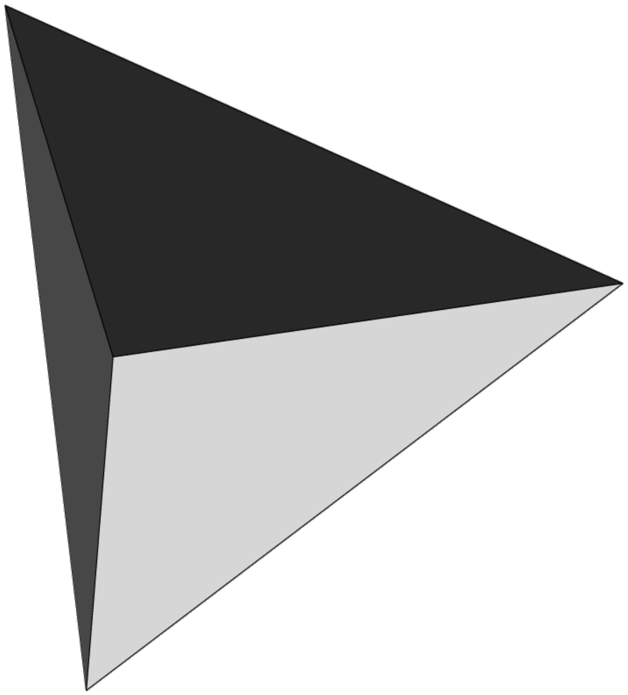




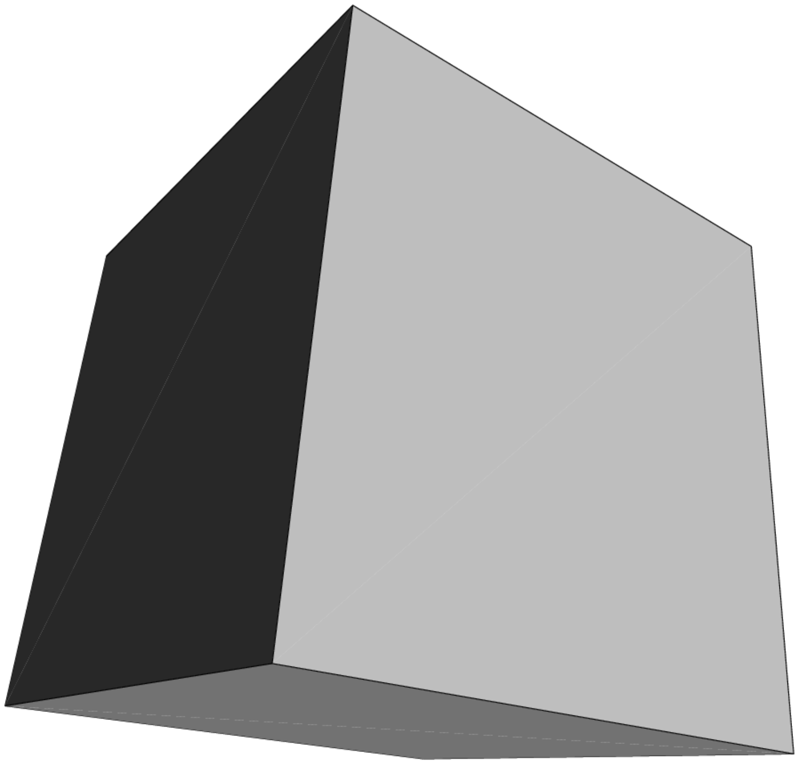
Polytypism



Fractal construction by carving



Sierpinsky sponge (4 steps)



Menger sponge (2 steps)

Two examples

1. Meristem growth
2. An iGEM project

The Growth of a Meristem

[PNAS 103(5), 1627-1632, 2006]

Pierre Barbier de Reuille

Mikaël Lucas

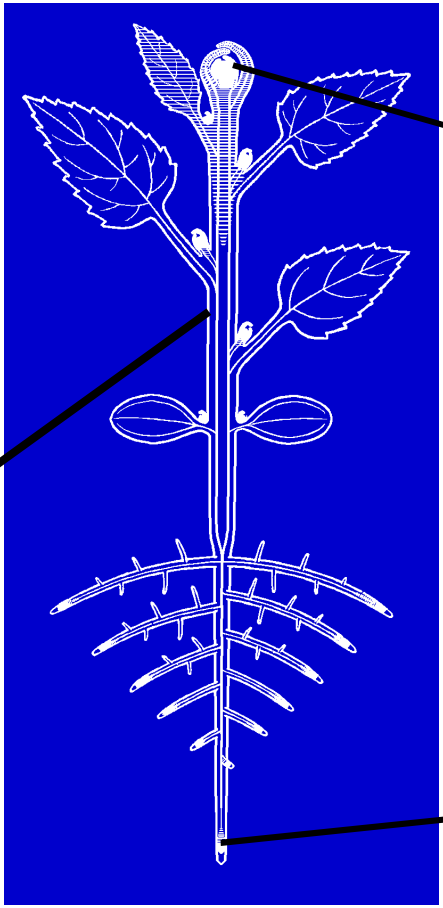
Jan Traas

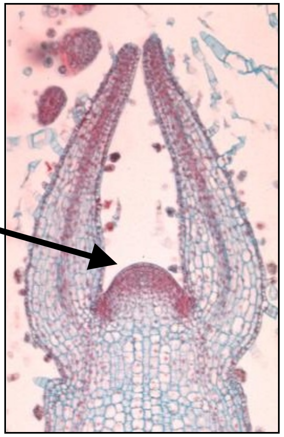
Christophe Godin

CIRAD/INRA/INRIA

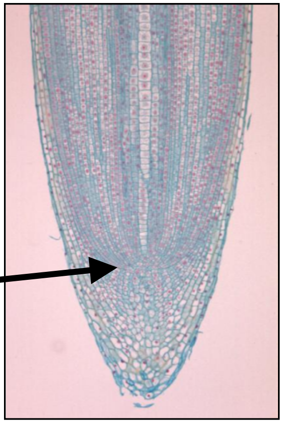
Organs positionning at the shoot apex

Cambium



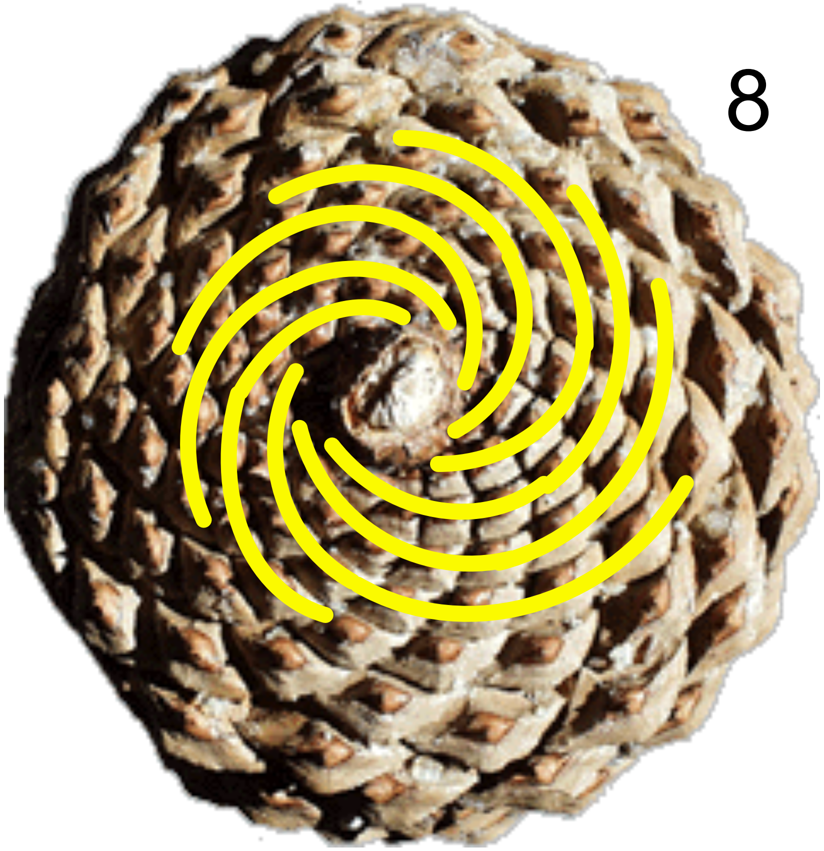


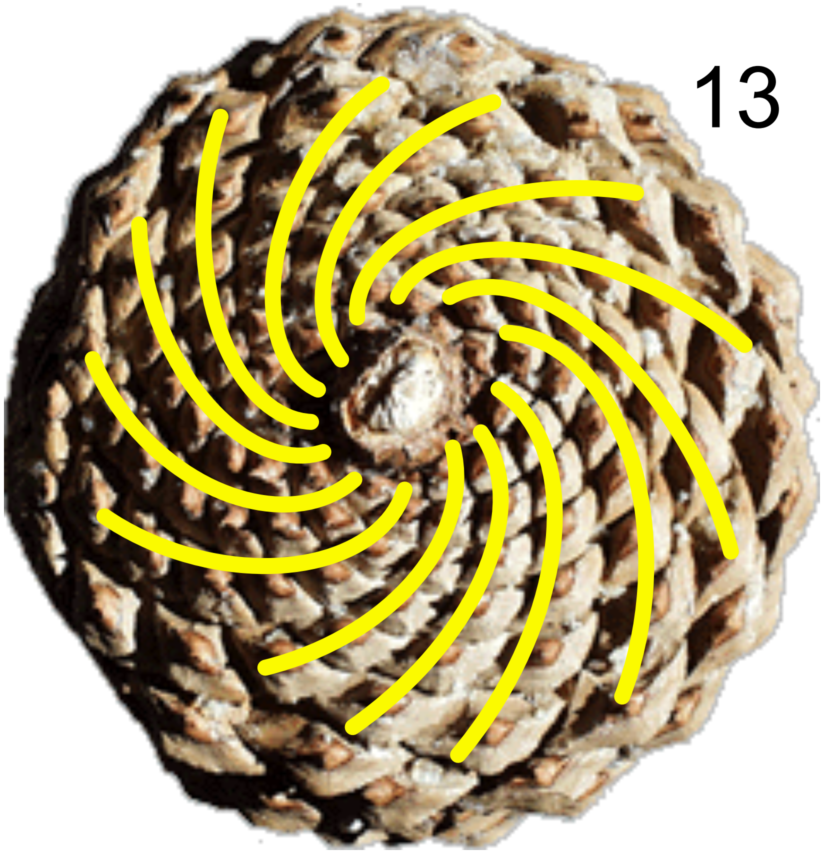
Shoot apical meristem



Root apical meristem

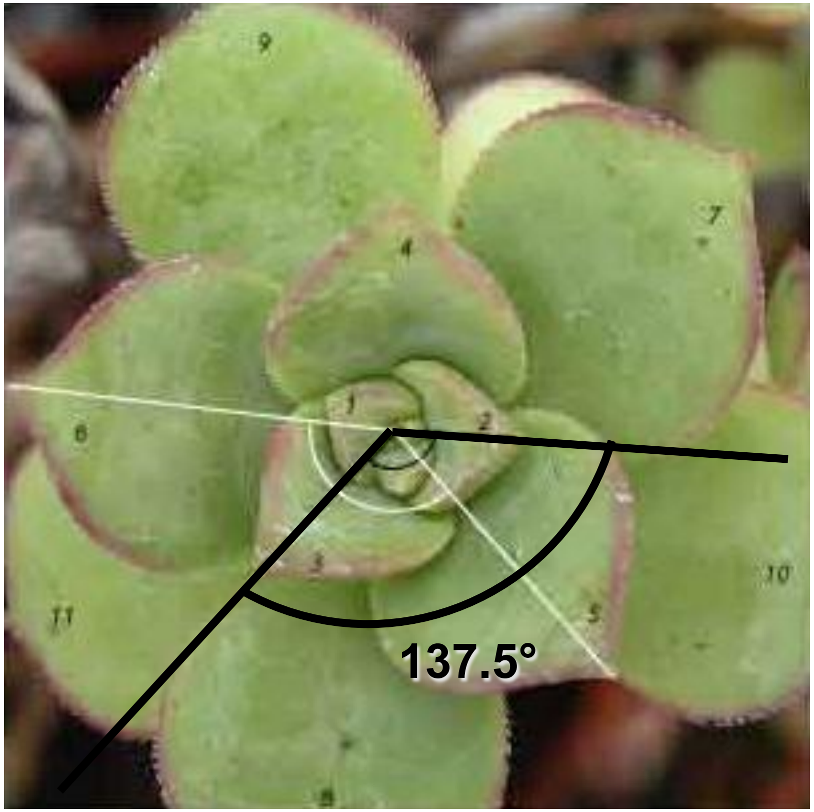
Fibonacci and phyllotaxis





8,13 → Two successive numbers of the Fibonacci series

Phyllotaxis : divergence angle

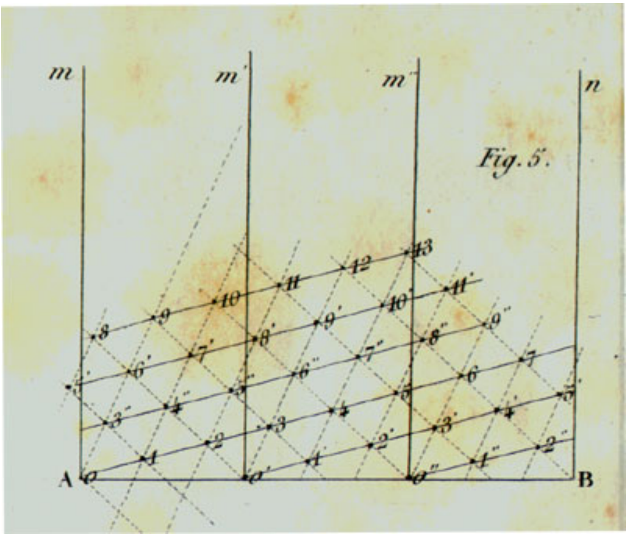


Phyllotaxis models: three kinds of approaches

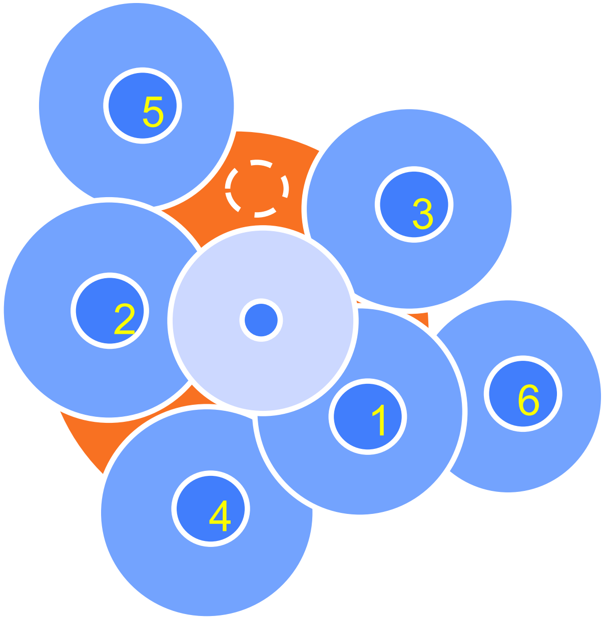
Geometrical

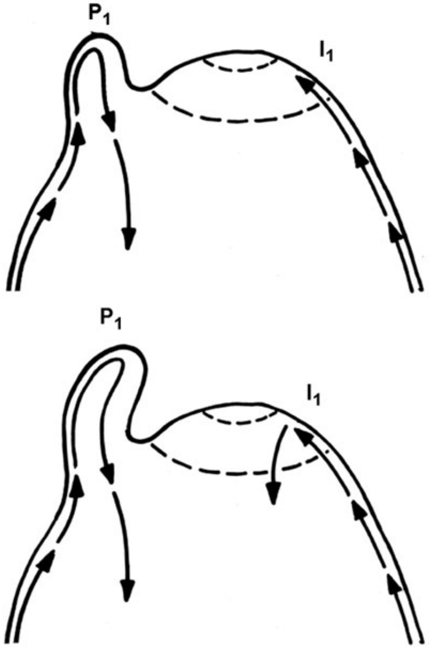
Dynamical

Physiological



(Bravais & Bravais,1837)



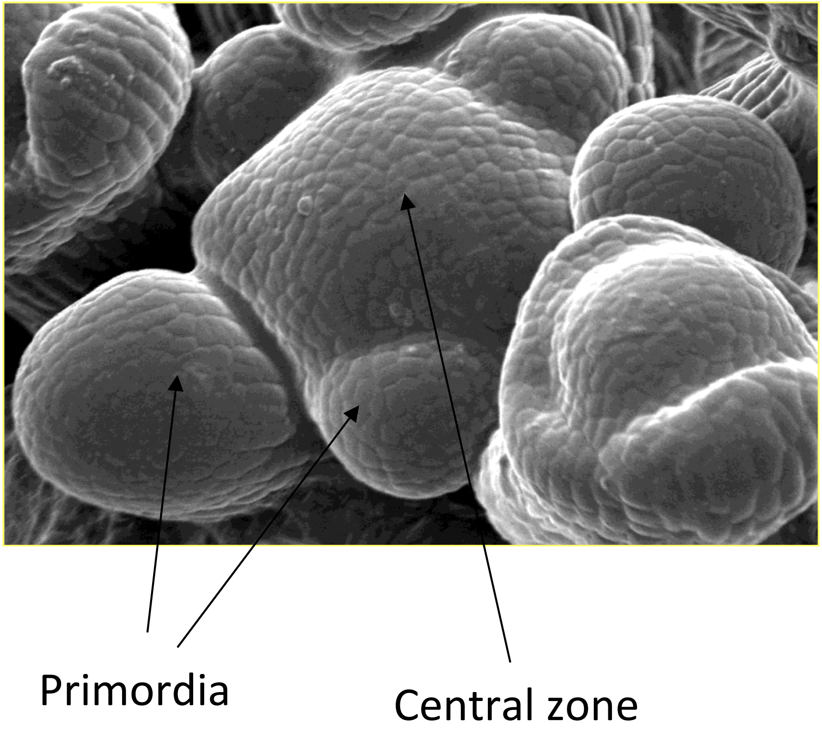


(Hofmeister, 1868)

(Snow and Snow, 1962)

(Reinhardt et al., 2000)

A shoot apical meristem



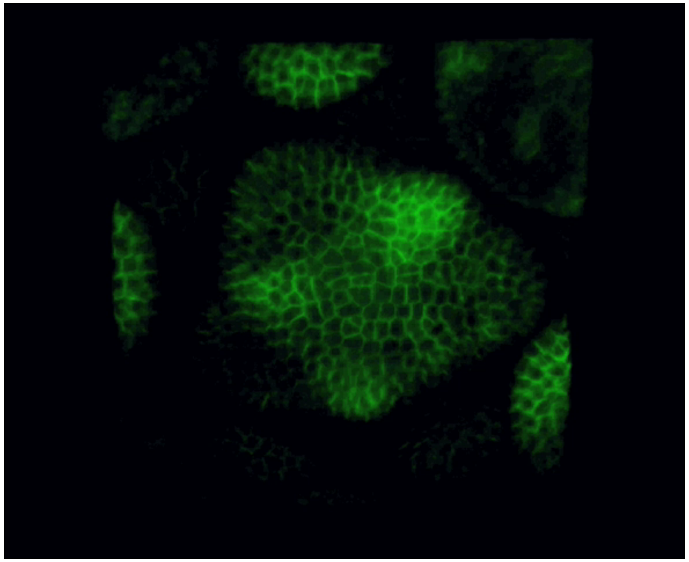
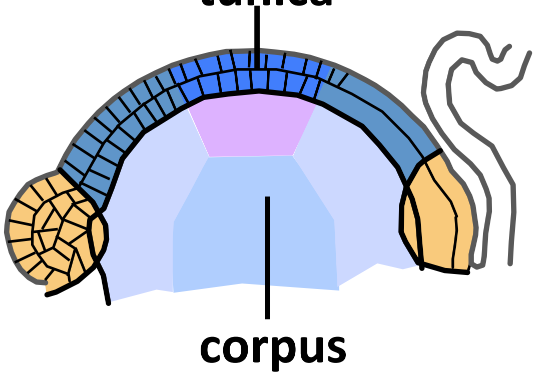
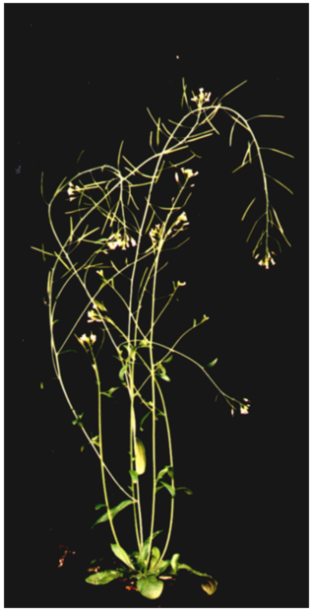


Image sequence showing cell division patterns via membrane-bound PIN1, in Shoot Apical Meristem (SAM), nearby floral meristems, and the boundaries between them (M. Heisler).
http://computableplant.ics.uci.edu/ (E. Mjolness)

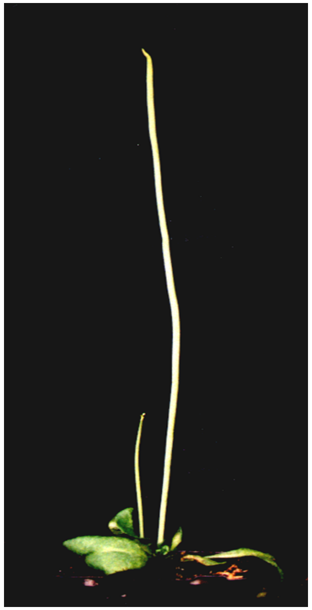


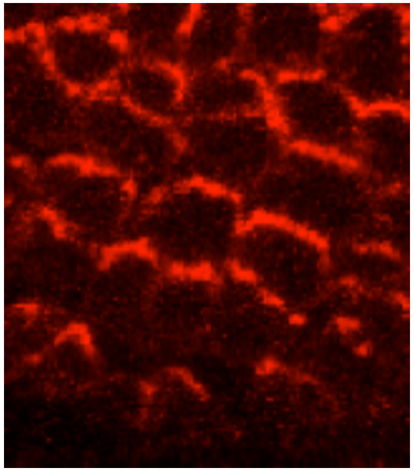
Active transport of auxine

wild type

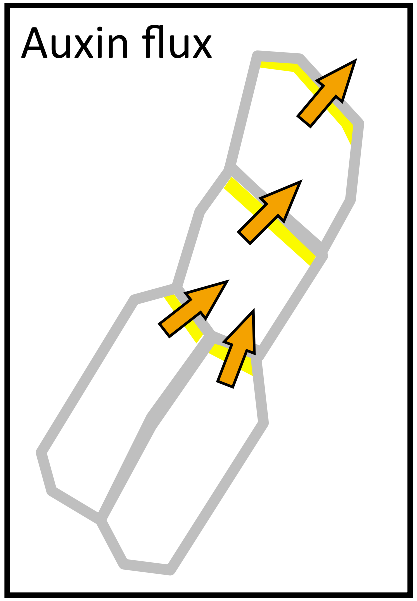


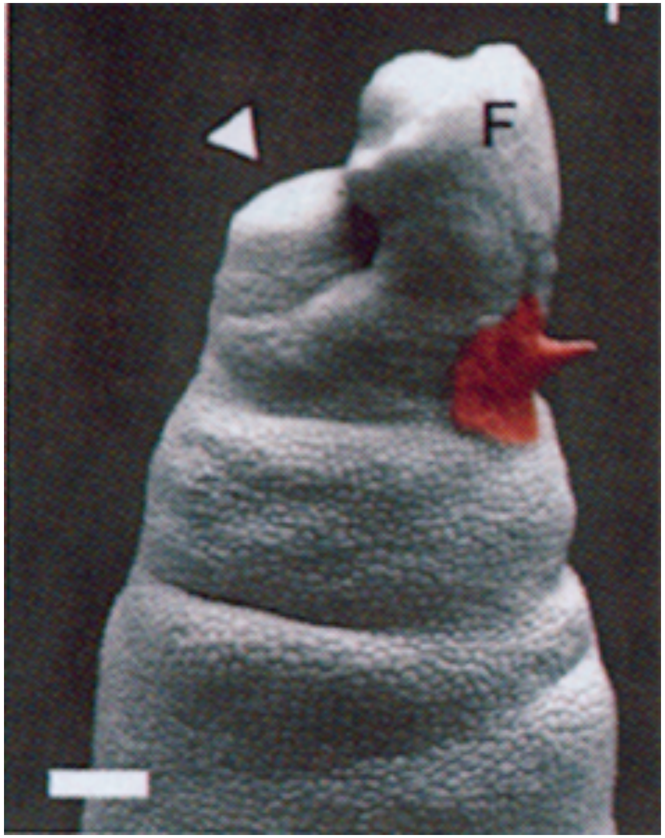
pin-1 mutant





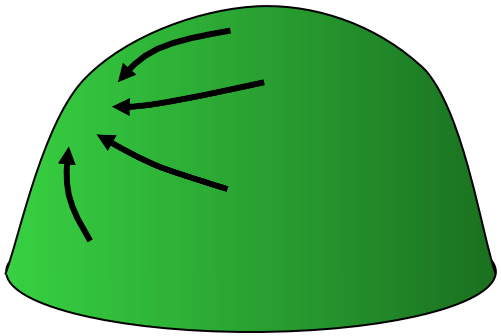
Immunolabelling of PIN-FORMED1 protein



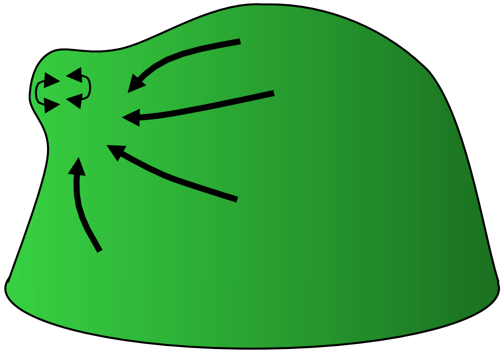


high concentration of auxine induces organ initiation

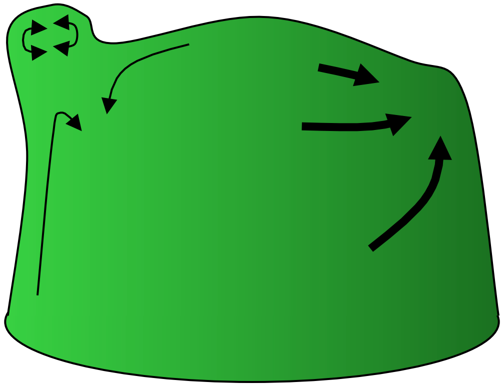
(DS)^2



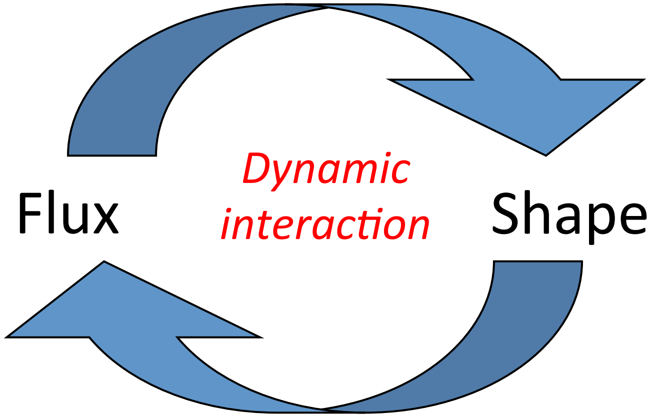
flux…



changes form…



which changes flux…



Flux

Dynamic interaction

Shape

Model

- ***Cell internal state and processes***
  capacity of division, springs relaxed length, primordium/center,
  concentration of auxin, auxin degradation / evacuation, inhibitor
  promotion to primordium, “pump magnetism”

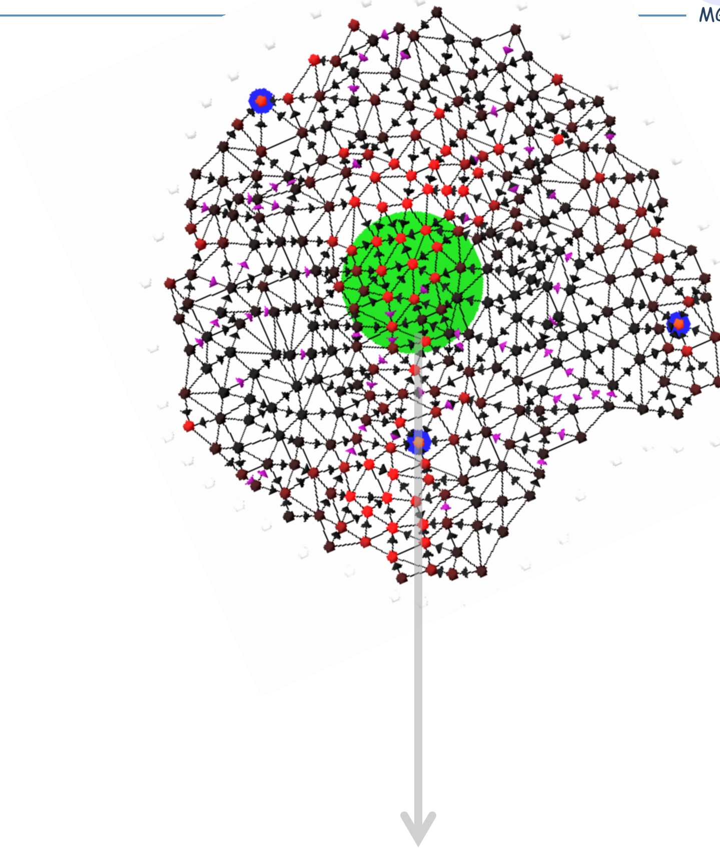
- ***Movement*** (due to cell growth)

- ***Growth***: increase of spring relaxed length

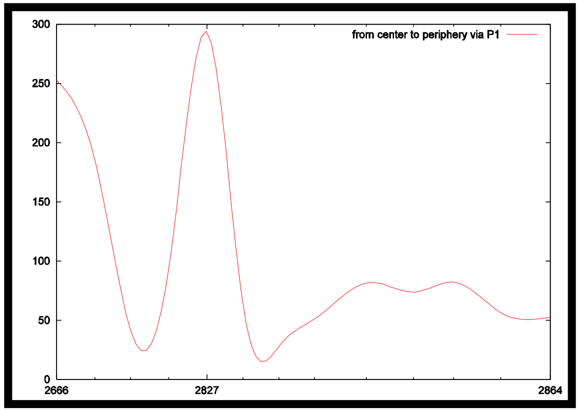
- ***Division***: when size > threshold

- ***Cell interaction***
  Passive diffusion of auxin, active pumping of auxin

```
trans Auxin = {
      x, y / pump(x,y)
      → x+{x.auxin -= δ}, y+{y.auxin += δ}
}
```

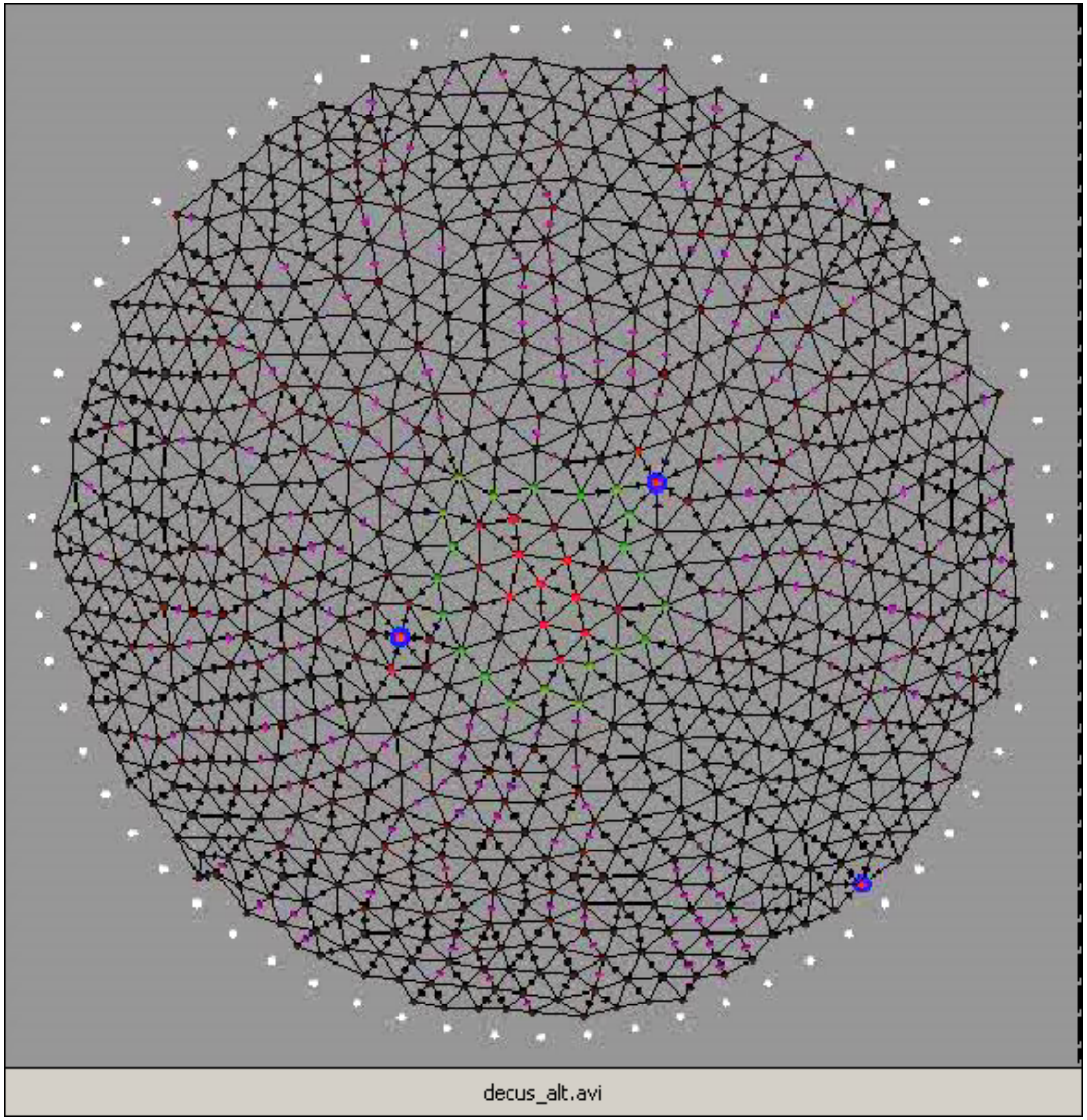


P. Barbier de Reuille

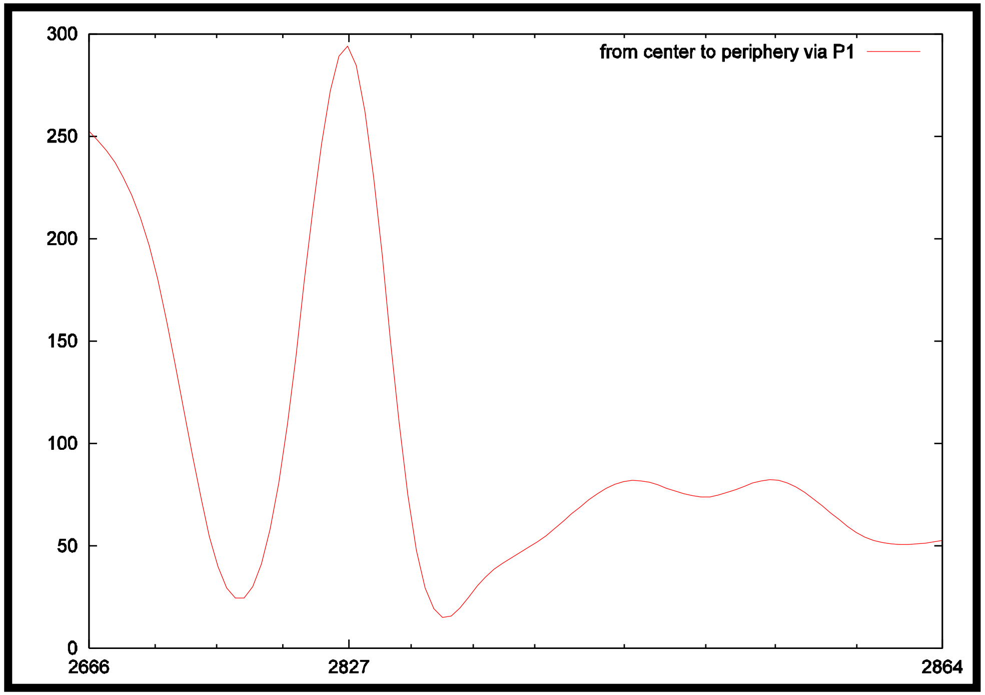


Auxin level

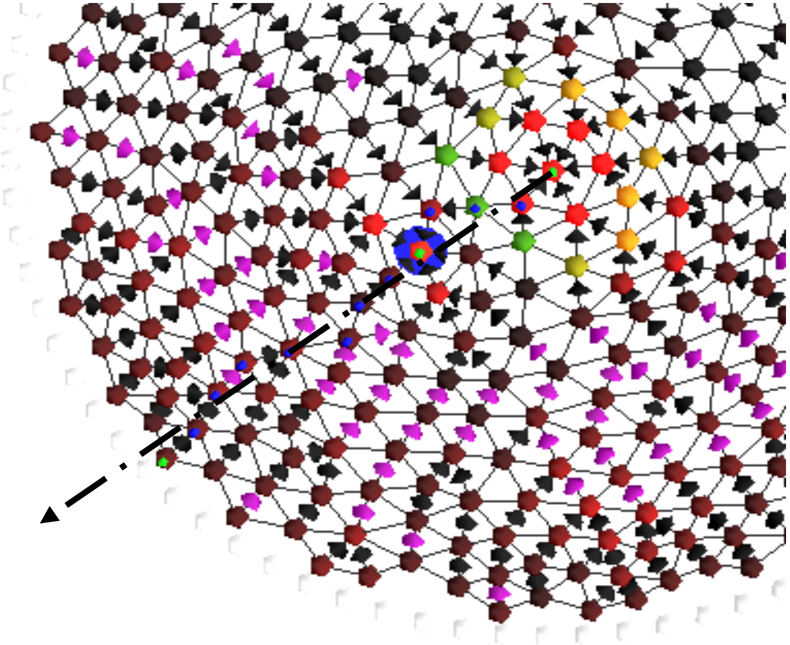
Simulation



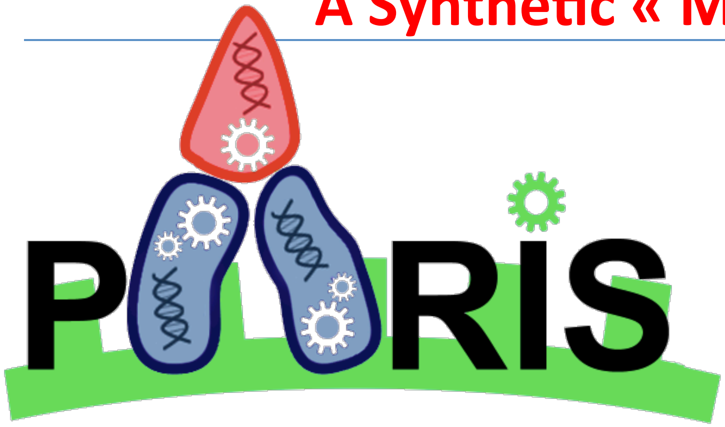
Primordium local inhibition

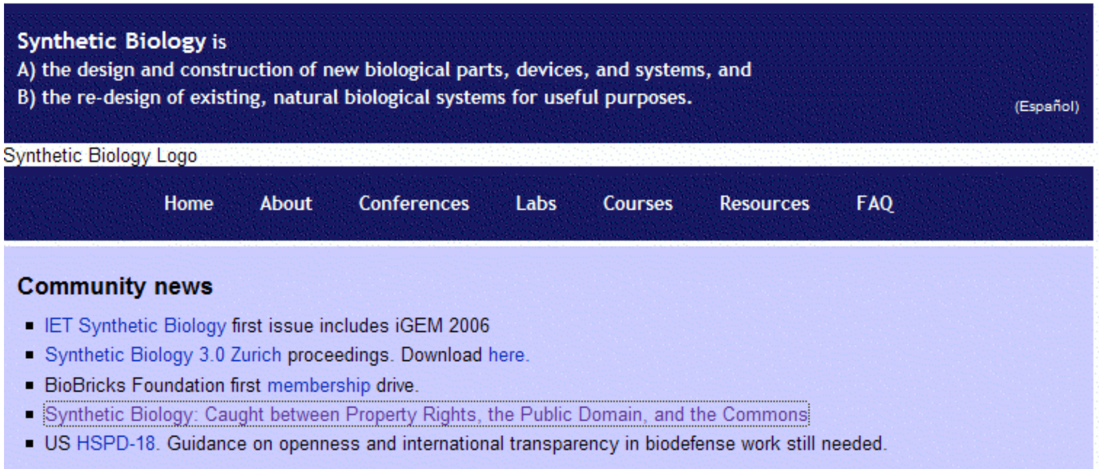


Auxin level



A Synthetic « Multicellular Bacterium »





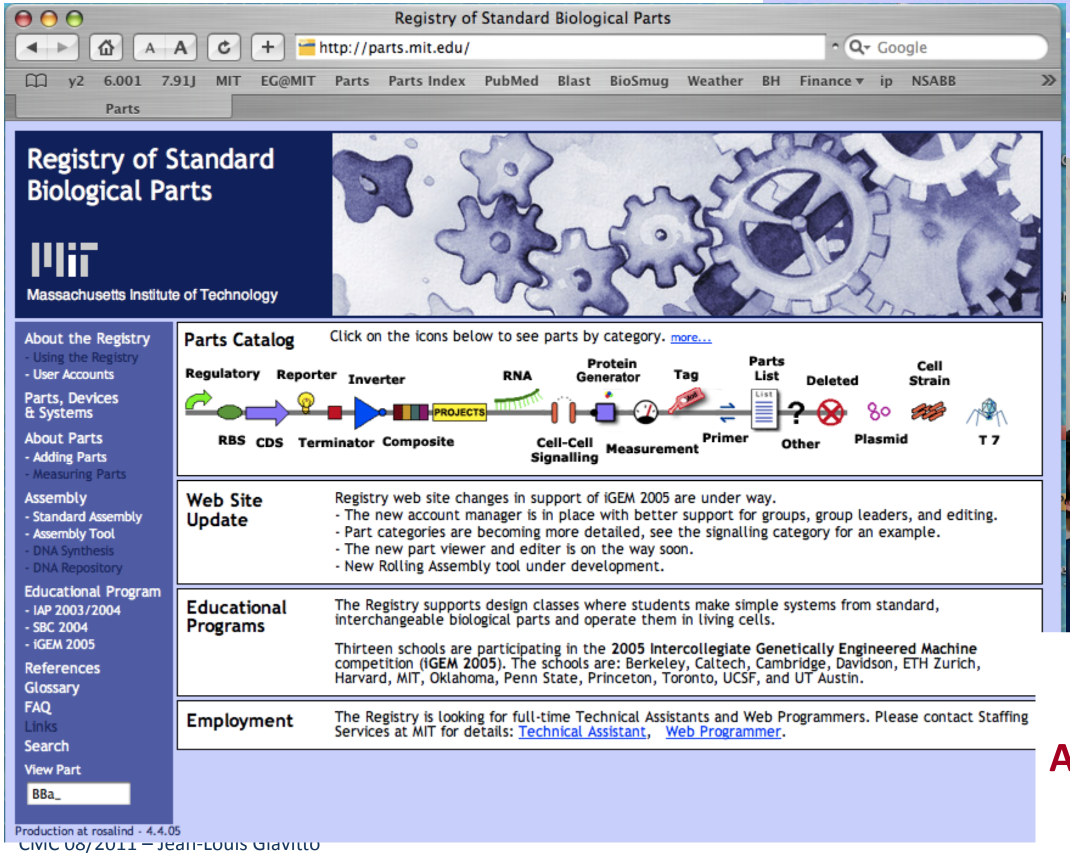
Synthetic Biology is
A) the design and construction of new biological parts, devices, and systems, and
B) the re-design of existing, natural biological systems for useful purposes.

Community news

- IET Synthetic Biology first issue includes iGEM 2006
- Synthetic Biology 3.0 Zurich proceedings. Download here.
- BioBricks Foundation first membership drive.
- Synthetic Biology: Caught between Property Rights, the Public Domain, and the Commons
- US HSPD-18. Guidance on openness and international transparency in biodefense work still needed.

Resources

- Press articles
- Publications: citeulike, connotea, PubMed



Registry of Standard Biological Parts

Parts Catalog Click on the icons below to see parts by category. more...

Web Site Update

Registry web site changes in support of iGEM 2005 are under way.
- The new account manager is in place with better support for groups, group leaders, and editing.
- Part categories are becoming more detailed, see the signalling category for an example.
- The new part viewer and editer is on the way soon.
- New Rolling Assembly tool under development.

Educational Programs

The Registry supports design classes where students make simple systems from standard, interchangeable biological parts and operate them in living cells.

Thirteen schools are participating in the 2005 **Intercollegiate Genetically Engineered Machine** competition (**iGEM 2005**). The schools are: Berkeley, Caltech, Cambridge, Davidson, ETH Zurich, Harvard, MIT, Oklahoma, Penn State, Princeton, Toronto, UCSF, and UT Austin.

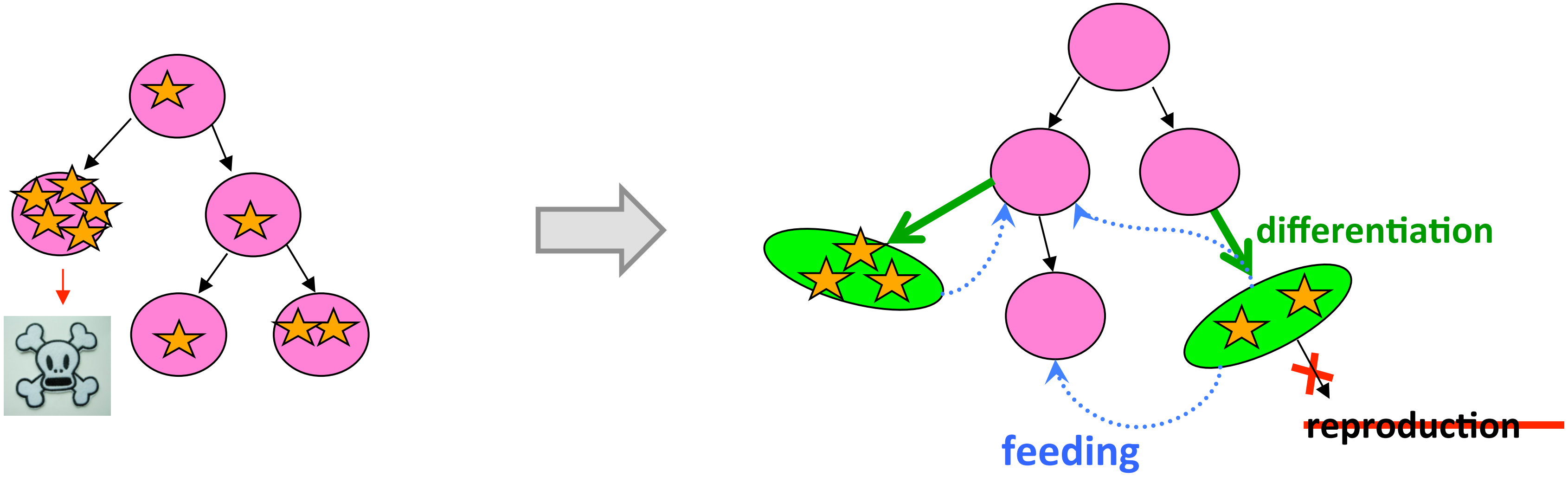
Employment

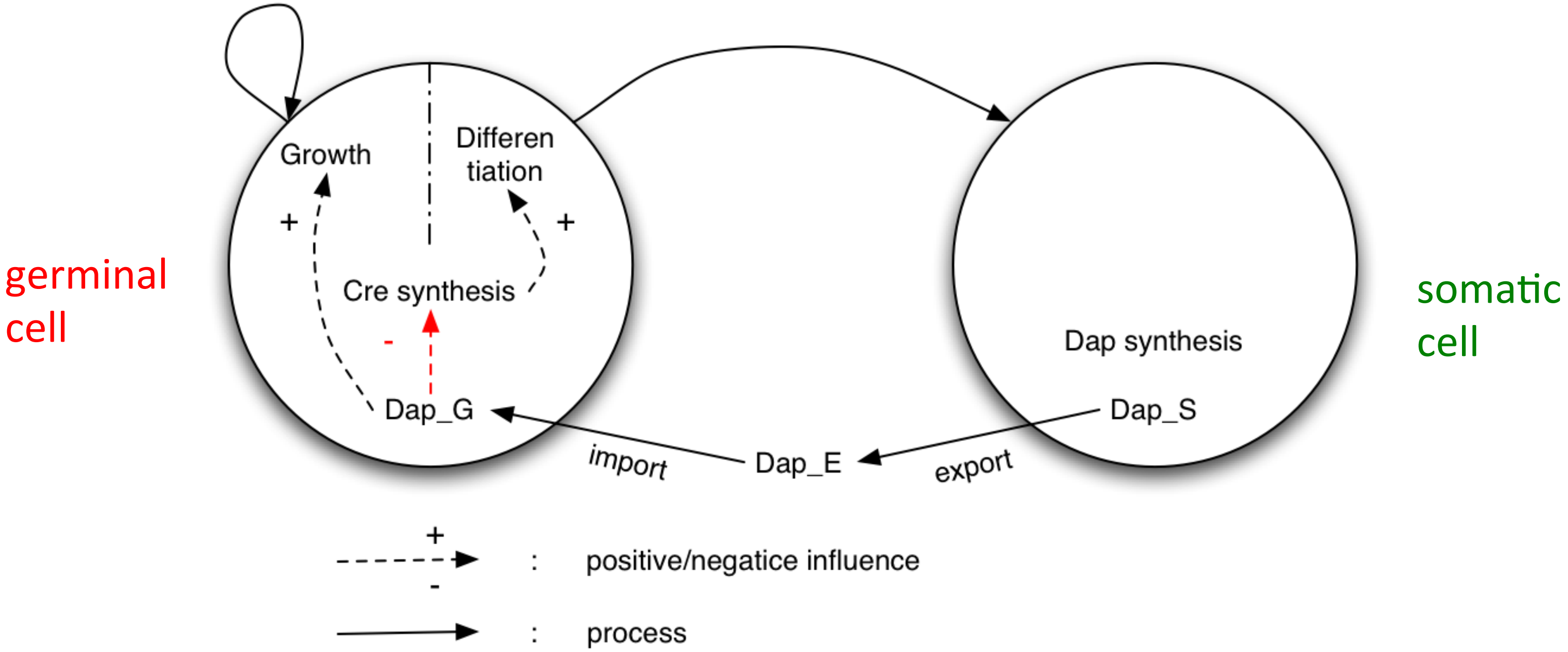
The Registry is looking for full-time Technical Assistants and Web Programmers. Please contact Staffing Services at MIT for details: Technical Assistant, Web Programmer.

Production at rosalind - 4.4.05

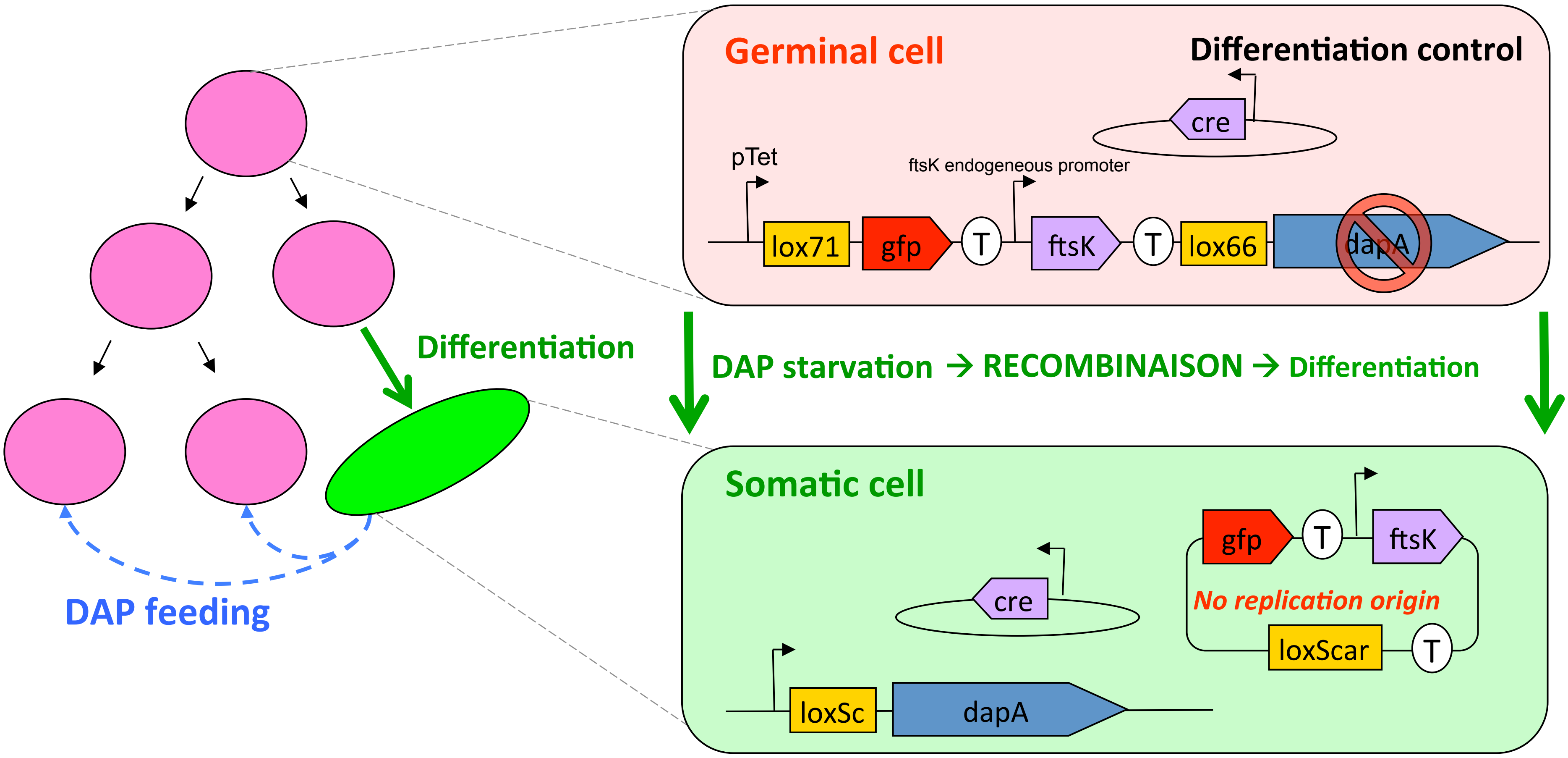
David Bikard, Thomas Landrain, David Puyraimond, Eimad Shotar, Gilles Vieira, Aurélien Rizk, David Guegan, Nicolas Chiaruttini, Thomas Clozel, Thomas Landrain

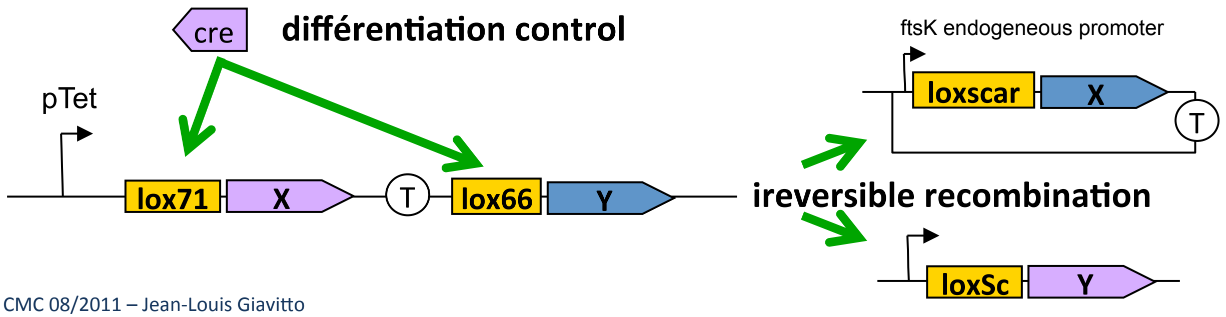
The Paris iGEM project: a « multicellular bacteria » to decouple growth and transgene expression

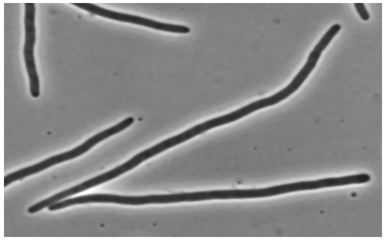




Design principle







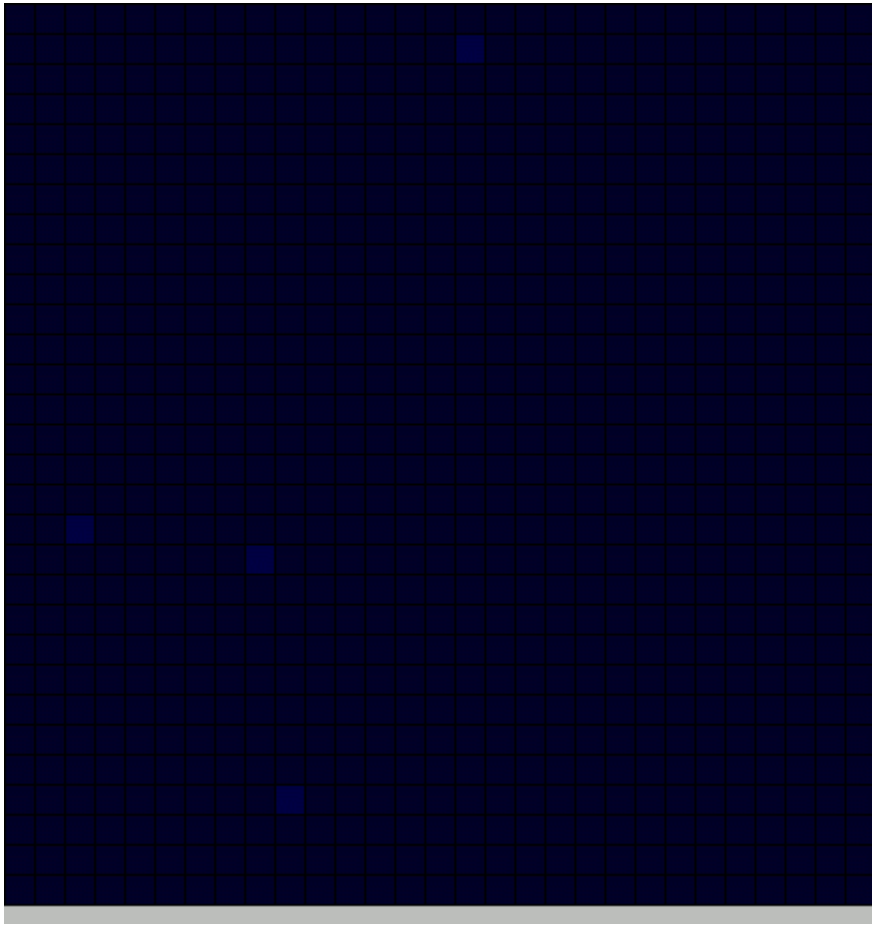
ftsK needed for cellular division

Anime

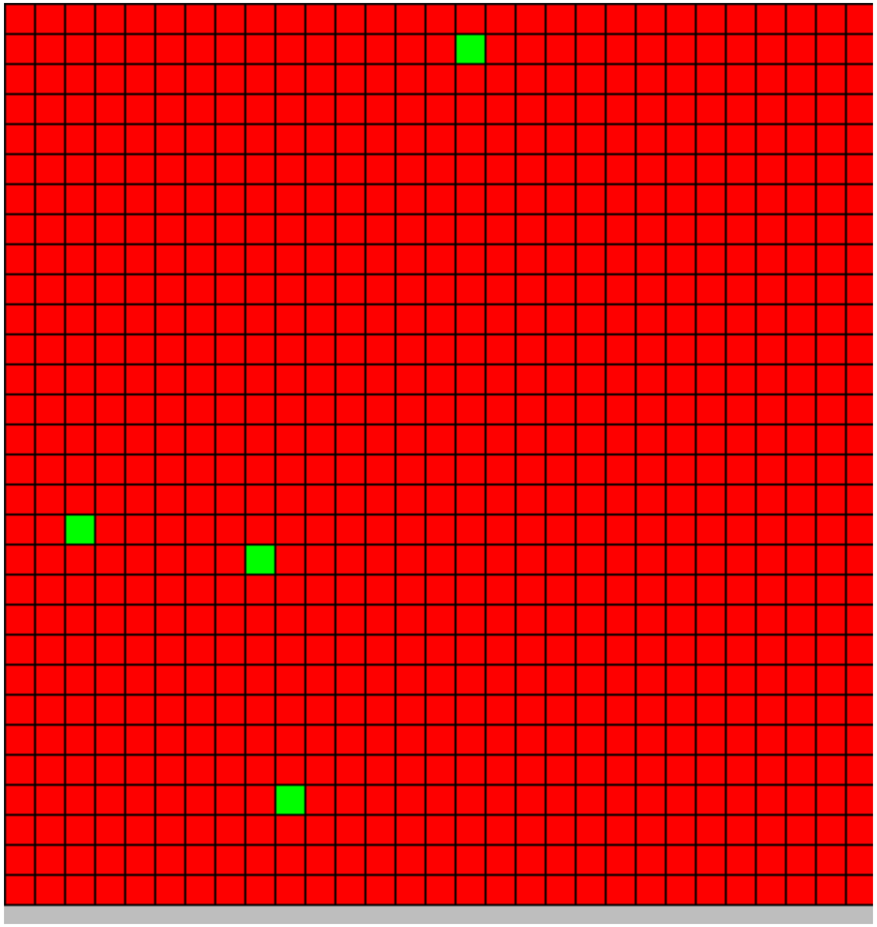
© David Bikar
& the 2007 Paris iGEM team

Proof of Concept: Simulation to answer 4 questions

- **How does differentiation induces feeding?** (proof of concept)
  cellular automaton (in MGS)



diffusion of DAP



somatic and germ cell

Proof of Concept: Simulation to answer 4 questions

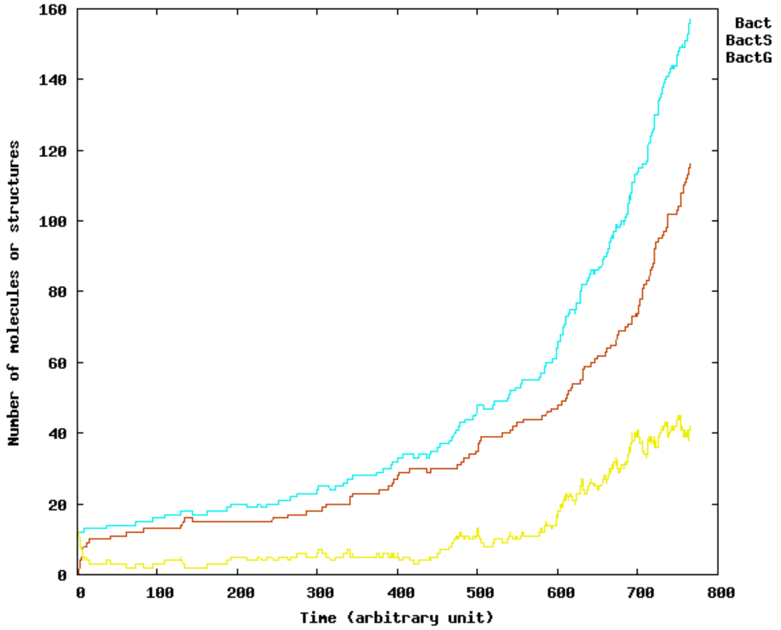
- **How does differentiation induces feeding?** (proof of concept)
  cellular automaton (in MGS)
- **How do spatial organization and distribution evolve?**
  agents based system (in MGS)

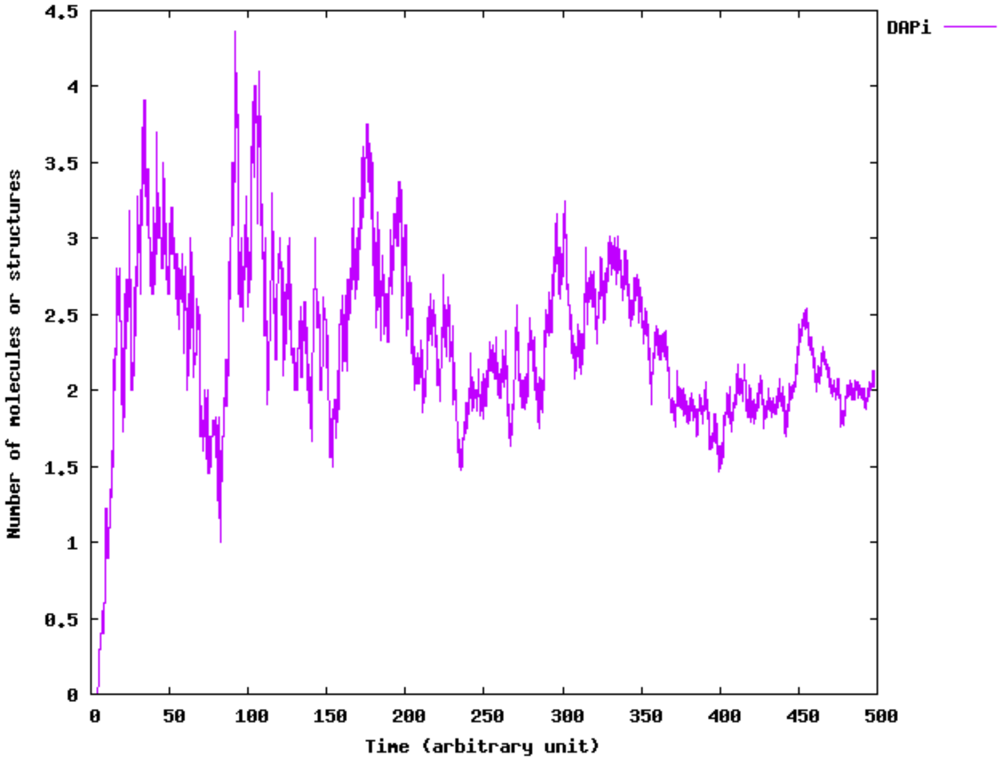
Proof of Concept: Simulation to answer 4 questions

- **How does differentiation induces feeding?** (proof of concept)
  cellular automaton (in MGS)
- **How do spatial organization and distribution evolve?**
  agents based system (in MGS)
- **How robust and tunable is the model?**
  ODE kinetics (matlab)

Proof of Concept: Simulation to answer 4 questions

- **How does differentiation induces feeding?** (proof of concept)
  cellular automaton (in MGS)
- **How do spatial organization and distribution evolve?**
  agents based system (in MGS)
- **How robust and tunable is the model?**
  ODE kinetics
- **How sensitive is the system to noise?**
  Gillespie based simulation (in MGS)

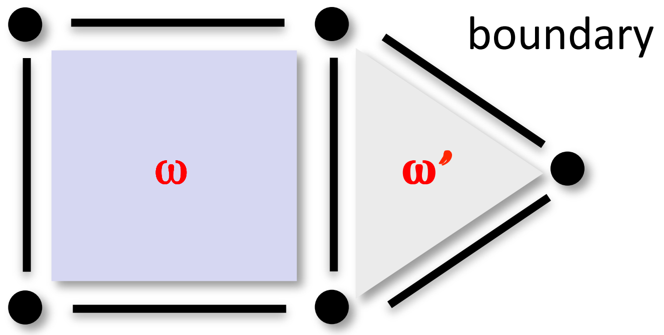




Blurring the discrete/continuous barrier

Cochains as (a special kind of) Transformations

- The Boundary Operator ∂
  - Starting point of the elaboration of a discrete diff. calculus
  - *Transport* of data *from cells to their faces*



  - Cochains notation

    The boundary operator is a cochain

    cochain = chain ⟶ chain

$$\partial = \sum_{\sigma\in\mathcal{K}} \partial_\sigma.\sigma \quad \text{with} \quad \forall\sigma\in\mathcal{K},\ \partial_\sigma(g) = \sum_{\tau<\sigma} o_{\sigma\tau}(g).\tau$$

  - MGS notation

```
trans Boundary      = {
  x => CofacesFold(fun y acc -> $o_{\hat{}y\hat{}x}(y)$ $+_G$ acc,$0_G$,^x) }
```

Cochains as Transformations

- Derivative Operator $\mathbf{d}$
  - Defined w.r.t. the discrete Stockes' Theorem

$$[\mathbf{d}T, c] = [T, \partial c]$$

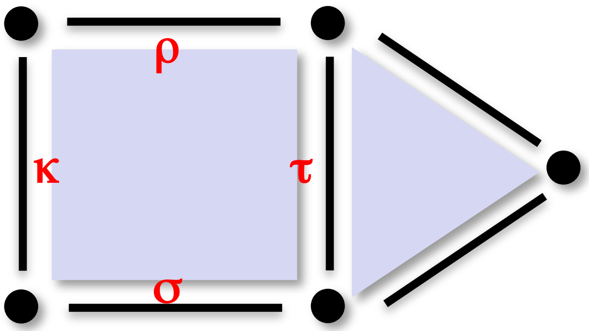
  - Cochains Notation

    One can show that the derivative verifies

$$\mathbf{d} = \sum_{\tau \in \mathcal{K}} \mathbf{d}_\tau.\tau \quad \text{with} \quad \forall \tau \in \mathcal{K},\ \mathbf{d}_\tau(f) = \sum_{\tau < \sigma} (f \circ o_{\sigma\tau}).\sigma$$

  - MGS Notation

    We directly use the Stockes' Theorem



```
let Derivative  T = fun c -> T (Boundary c)
```

Cochains as Transformations

- Illustrative example : the Laplacian Operator Δ

  – The Laplacian in terms of ★ and **d** [Desbrun *et al*., 2006]

$$\Delta \;=\; \delta\mathbf{d} + \mathbf{d}\delta \qquad \text{where} \quad \delta = (-1)^{n(k-1)+1} \star \mathbf{d} \star$$

  – MGS notation

    Big assumption: here the Hodge star ★ is the co-derivative **d**^{co} assuming uniform geometry

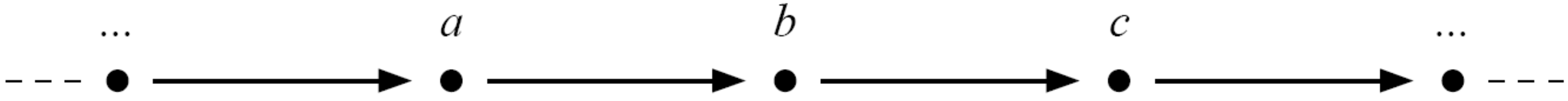
```
let Laplacian T =
  let Sg T' c' =
    T'(trans { x => -1**((dim c')*((dim ^x)-1)+1)*x }(c'))
  in
    fun c ->    Derivative(Sg(Derivative^co(T)))(c)
              + Sg(Derivative^co(Derivative(T)))(c)
```

Cochains as Transformation

- Illustrative example : the Laplacian Operator Δ
  - Corresponding Data Transport (case of dimension 1)

$$\Delta = \mathbf{d}^{co}\mathbf{d}$$

    - Dimension 1: $\partial \circ \partial^{co}$
    - Stockes' Theorem: équivalence with

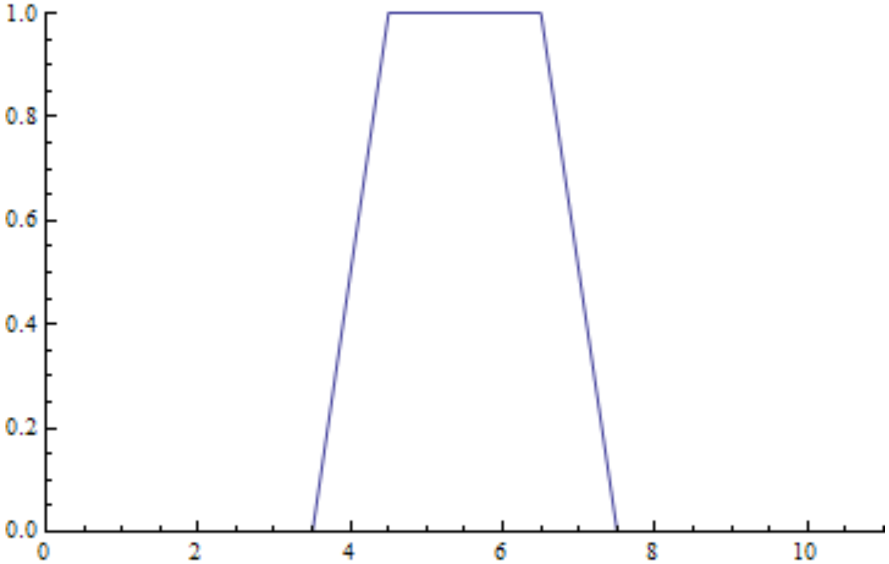


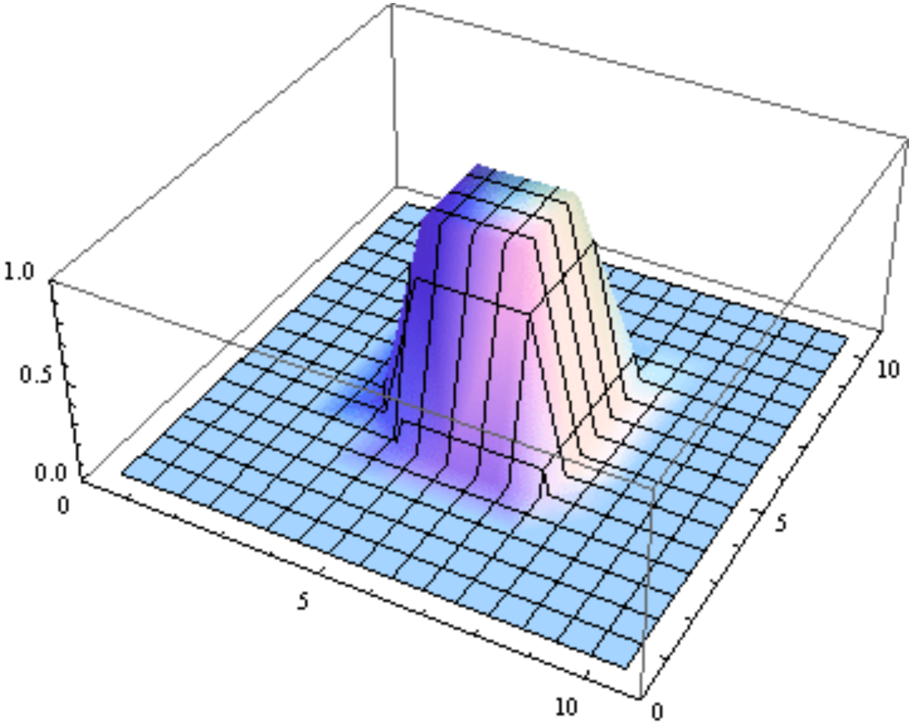
Cochains as Transformation

- Illustrative example : the Laplacian Operator Δ
  - Simulation of diffusion

$$\frac{\partial u}{\partial t} = D\Delta u$$

```
fun diffusion[D,orient](u) =
  u + D*Laplacian[orient=orient](Id)(u) ;;
```





Home message

1. **New kind of computational machines with “dynamic structure”**
   strange loop between structure and processes
   not new (program = data)
   but not understood (*e.g.,* type discipline to avoid that)

2. **Space matters**
   compartmentalization and beyond

3. **Killer app.**
   systems & synthetic biology
   the nano-world: form=function

4. **Usual tools of computer science are relevant**
   but the focus, the questions and the answers are new
   e.g., termination in rewriting

5. **Versatility of the MGS approach**
   time evolution = rewriting strategy
   kind of space = kind of objects to be rewritten

Acknowledgements

MGS: Antoine Spicher, Olivier Michel, Julien Cohen

S&S Bio : Hanna Klaudel, Franck Delaplace, Hugues Berry, Przemek Prusinkiewicz, Annick Lesne…

Spatial Computing: Jacob Beal, Fréderic Gruau, René Doursat…

Examples: Pierre Barbier de Reuille, Christophe Godin, Samuel Bottani, the Paris iGEM'07 team

Some figures are borowed from Olivier Michel, Antoine Spicher, Pierre Barbier de Reuille, Franck Delaplace, Hugues Berry (INRIA), the iGEM Paris 2007…