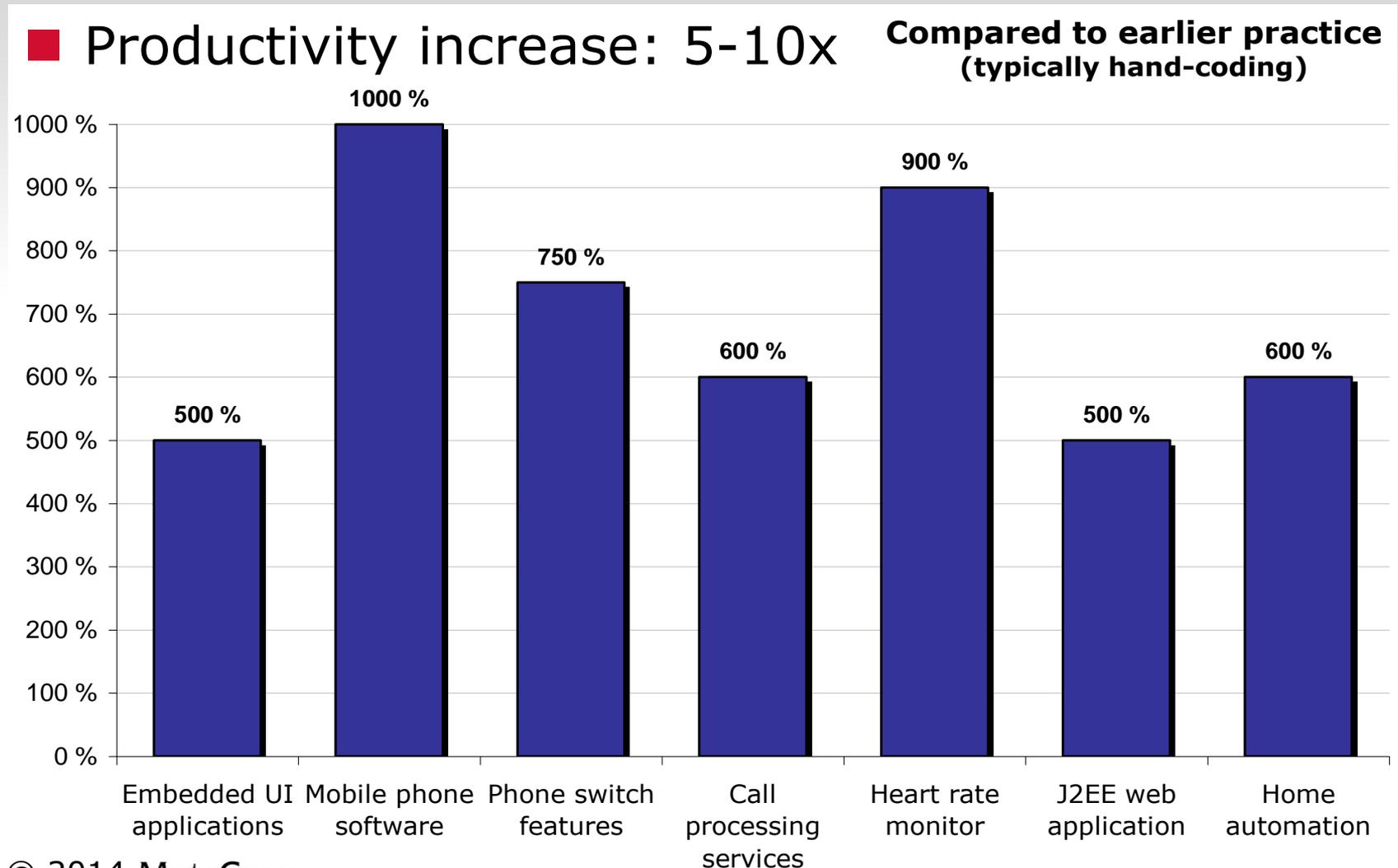
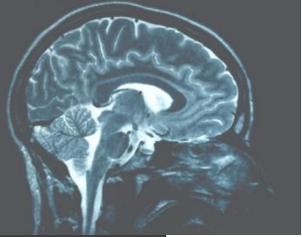


Graphical DSLs
aka
Domain-Specific Modelling

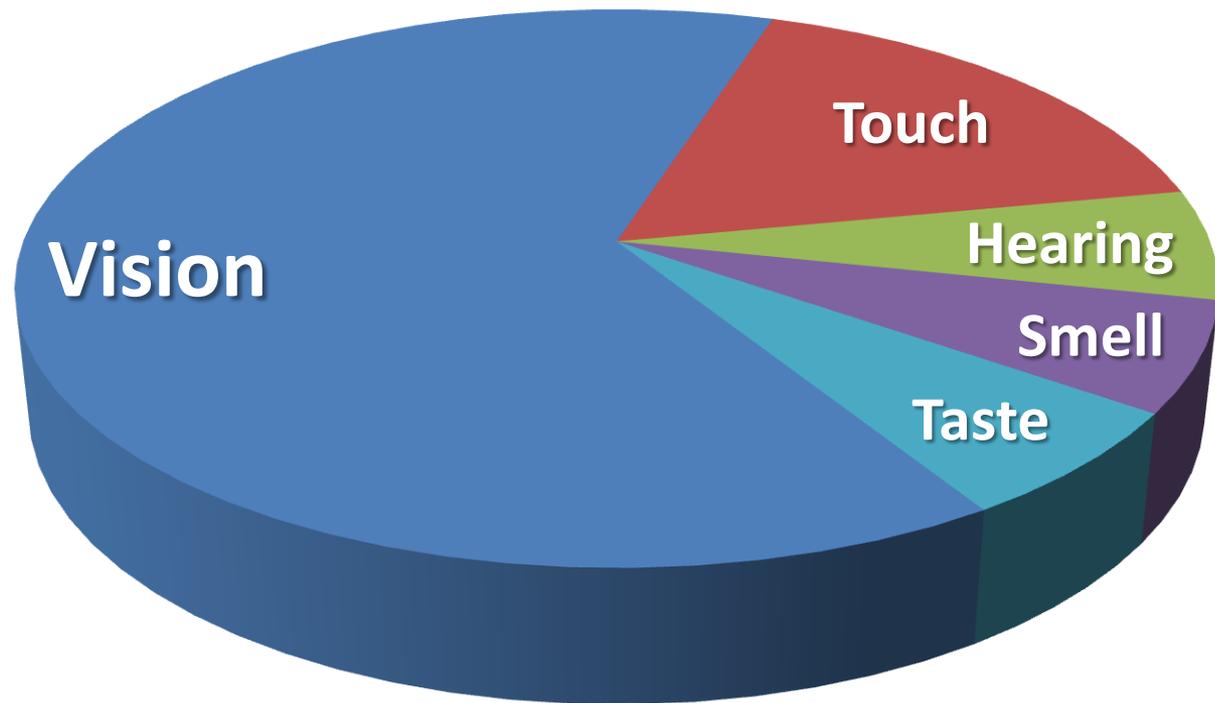
Domain-Specific Modelling: DSM

<http://metacase.com/blogs/stevek/blogView?entry=3446309083>





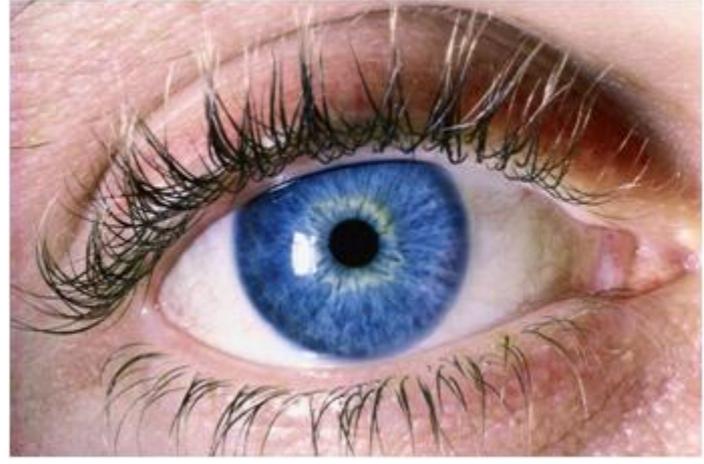
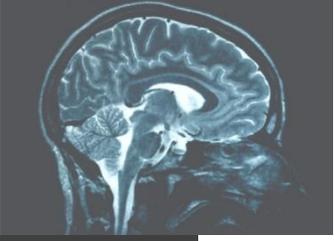
We have **visual** brains



Brain Power

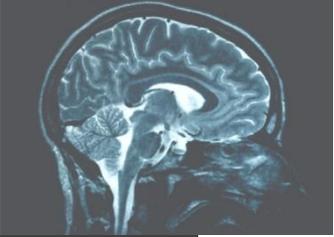
Sensory neurons

Brain Power



s
e
r
i
a
l

parallel
parallel
parallel
parallel
parallel



Brain Power

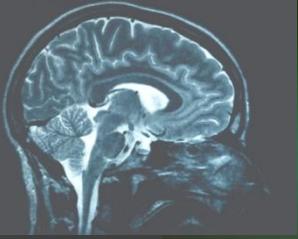
Text

s
e
r
i
a
l

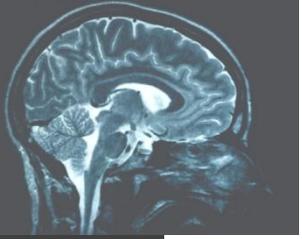
Visual

parallel
parallel
parallel
parallel
parallel
parallel

Brain Power



**Perceptual
vs.
Cognitive**

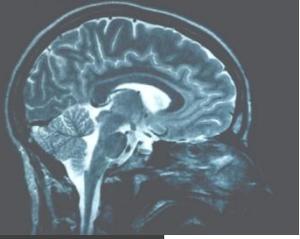


Moody TSE09

Brain Power

Perceptual Popout

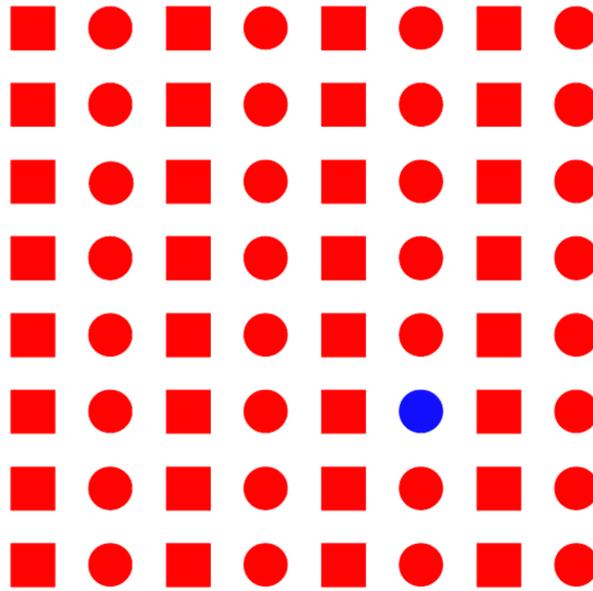
**Spot the
odd one out!**



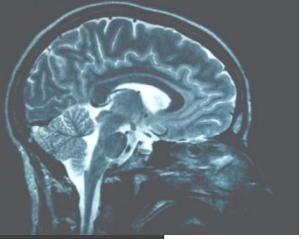
Moody TSE09

Brain Power

Perceptual Popout



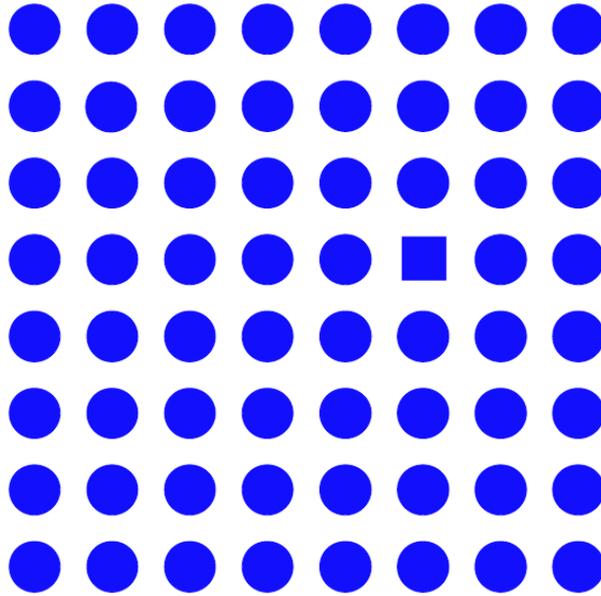
**Spot the
odd one out!**



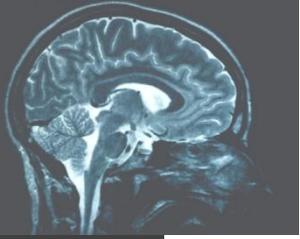
Moody TSE09

Brain Power

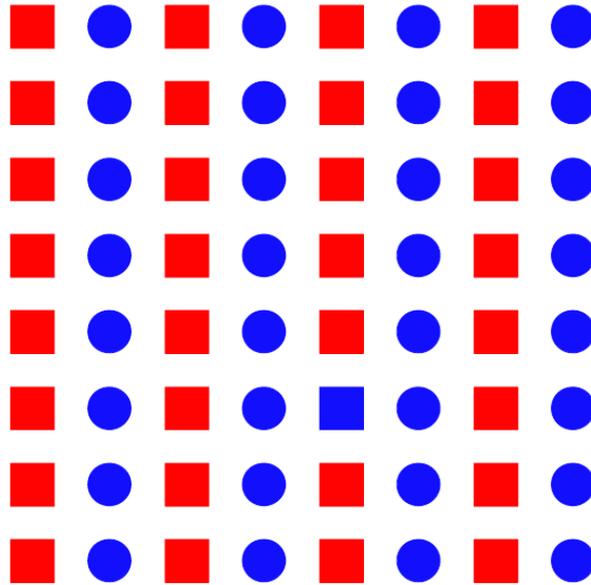
Perceptual Popout



**Spot the
odd one out!**



Perceptual Popout



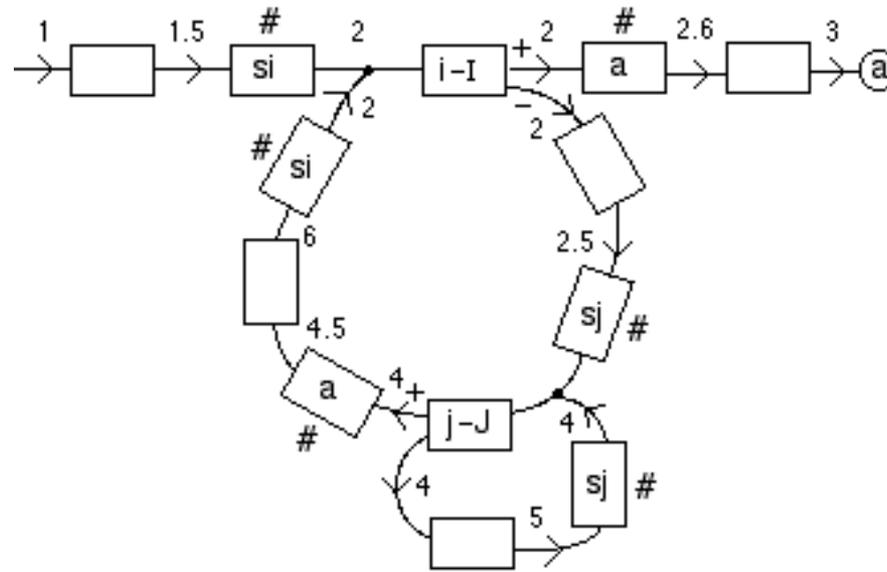
**Spot the
odd one out!**

Graphical vs. Textual



.txt

Software flow diagram



1940s

LISP
FORTRAN
ALGOL
COBOL
1950s

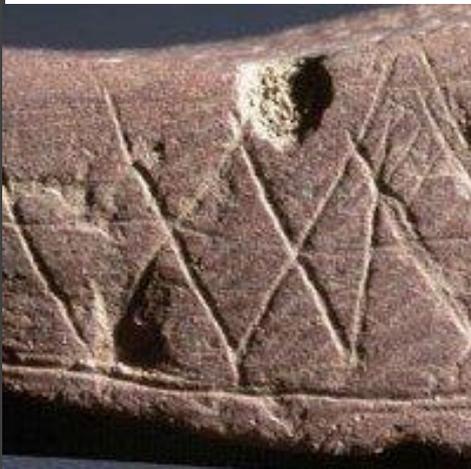
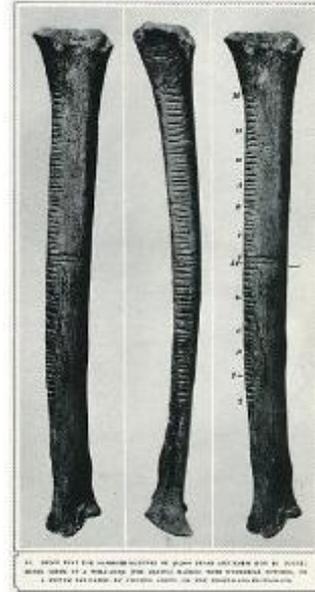
Graphical + Textual

.txt

Visual languages

25 000 years

before writing



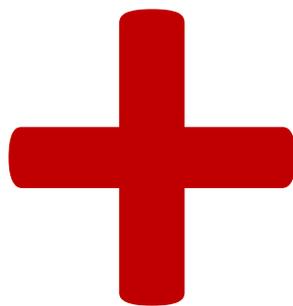
Graphical + Textual

.txt

Dual Coding



GPU



CPU

Graphical + Textual

Notation, Notation, Notation

$$\frac{\partial}{\partial \theta} \mathbf{M}T(\xi) = \frac{\partial}{\partial \theta} \int_{\mathbb{R}^n} T(x) f(x, \theta) dx = \int_{\mathbb{R}^n} \frac{\partial}{\partial \theta} T(x) f(x, \theta) dx$$

$$\frac{\partial}{\partial a} \ln f_{a, \sigma^2}(\xi_1) = \frac{(\xi_1 - a)}{\sigma^2} f_{a, \sigma^2}(\xi_1) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(\xi_1 - a)^2}{2\sigma^2}\right\}$$

$$\int_{\mathbb{R}^n} T(x) \cdot \frac{\partial}{\partial \theta} f(x, \theta) dx = \mathbf{M}\left(T(\xi) \cdot \frac{\partial}{\partial \theta} \ln L(\xi, \theta)\right) = \int_{\mathbb{R}^n} \frac{\partial}{\partial \theta} T(x) f(x, \theta) dx$$

$$\int_{\mathbb{R}^n} T(x) \cdot \left(\frac{\partial}{\partial \theta} \ln L(x, \theta)\right) \cdot f(x, \theta) dx = \int_{\mathbb{R}^n} T(x) \cdot \left(\frac{\partial}{\partial \theta} \frac{f(x, \theta)}{f(x, \theta)}\right) \cdot f(x, \theta) dx$$

$$\frac{\partial}{\partial \theta} \mathbf{M}T(\xi) = \frac{\partial}{\partial \theta} \int_{\mathbb{R}^n} T(x) f(x, \theta) dx = \int_{\mathbb{R}^n} \frac{\partial}{\partial \theta} T(x) f(x, \theta) dx = \int_{\mathbb{R}^n} \frac{\partial}{\partial \theta} T(x) f(x, \theta) dx$$

$$\frac{\partial}{\partial \theta} \ln f_{\mu, \sigma^2}(\xi) = \frac{\partial}{\partial \theta} \left[-\frac{1}{2\sigma^2} (\xi - \mu)^2 - \frac{1}{2\sigma^2} \ln \sigma^2 \right]$$
$$\frac{\partial}{\partial \theta} \ln f_{\mu, \sigma^2}(\xi) = \frac{(\xi - \mu)}{\sigma^2} \frac{\partial \mu}{\partial \theta} - \frac{1}{\sigma^2} \frac{\partial \sigma^2}{\partial \theta}$$
$$\int T(x) \cdot \frac{\partial}{\partial \theta} f(x, \theta) dx = M \left(T(\xi) \frac{\partial \xi}{\partial \theta} \right)$$
$$\int T(x) \left(\frac{\partial}{\partial \theta} \ln f(x, \theta) \right) \cdot f(x, \theta) dx = \int T(x) \left(\frac{\partial}{\partial \theta} \ln f(x, \theta) \right) \cdot f(x, \theta) dx$$
$$\frac{\partial}{\partial \theta} M T(\xi) = \frac{\partial}{\partial \theta} \int T(x) f(x, \theta) dx = \int T(x) \frac{\partial}{\partial \theta} f(x, \theta) dx$$

Notation, ...

Domain Users

care deeply
about notation!

“UI” for the language

Daniel Moody

The “Physics” of Notations:

Towards a Scientific Basis for Constructing
Visual Notations in Software Engineering,
IEEE Transactions on Software Engineering,
Vol. 35, No. 5, November-December 2009



[1] Alexander, C.W., *Notes On The Synthesis Of Form*. 1970, Boston, US: Harvard University Press.

[2] Avison, D.E. and G. Fitzgerald, *Information Systems Development: Methodologies, Techniques and Tools (3rd edition)*. 2003, Oxford, United Kingdom: Blackwell Scientific.

.....

[150] Zhang, J., *The Nature of External Representations in Problem Solving*. *Cognitive Science*, 1997. 21(2): p. 179-217.

[151] Zhang, J. and D.A. Norman, *Representations in Distributed Cognitive Tasks*. *Cognitive Science*, 1994.



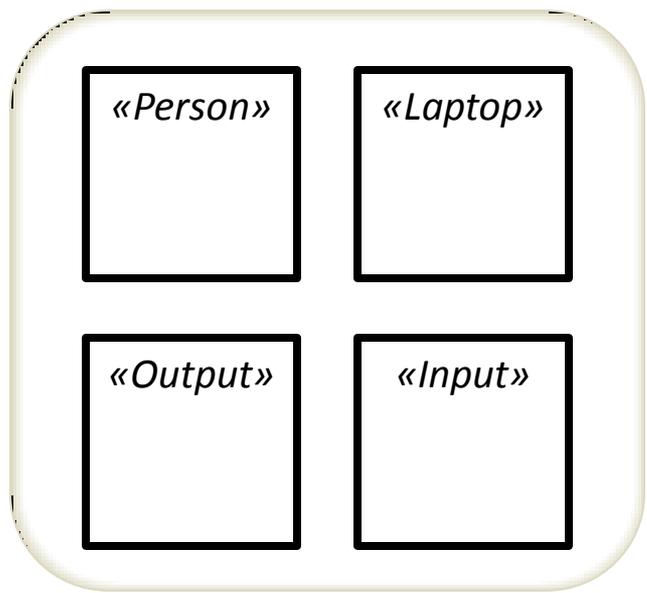
Form \geq Content

“ research in diagrammatic reasoning shows that the form of representations has an equal, if not greater, influence on cognitive effectiveness as their content [68, 122, 151]. ”

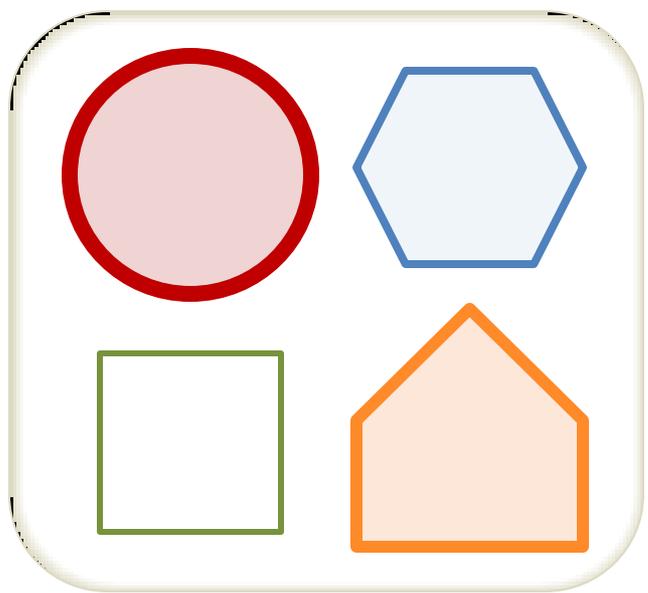


Concrete syntax **>** **Abstract syntax**

“ apparently minor changes in visual appearance can have dramatic impacts on understanding and problem solving performance [19, 68, 103, 122]... especially by novices [53, 56, 57, 79, 93, 106, 107]. ”



pictogram > geometric  **photo**



Notation, ...



Paradigm 1:

String matching in files

- Strings are 1-dimensional character arrays
- Look for same sequence, “E”, “m”, “p” etc.
 - Or UUID, unique identifier in XML
- Inefficient, hard to see, fragile
 - but familiar!

c	l	a	s	s		E	m	p	l	o	y	e	e
	.	.	.			c	l	a	s	s		M	a
n	a	g	e	r		e	x	t	e	n	d	s	
E	m	p	l	o	y	e	e		.	.	.		
D	e	v	e	l	o	p	e	r		e	x	t	e
n	d	s				E	m	p	l	o	y	e	e

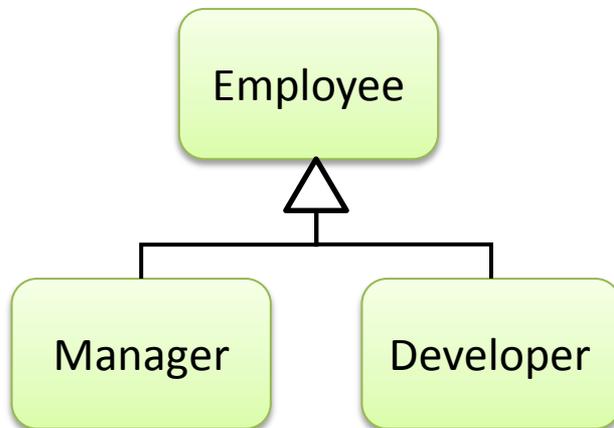




Paradigm 2:

Direct reference in repository

- Works like objects in memory
- Efficient: Direct pointer
- Visible: See referrers
- Robust: Change once
 - But less familiar!





Tool support for direct reference

- Concrete syntax: view
- UI: edit
- Cross-model references: link
- Disk representation: load

	view	edit	link	load
Xtext				
EMF/GMF	✓	✓		
MPS		✓	✓	
MetaEdit+	✓	✓	✓	✓



modeling != coding

Diff + merge:

Text easy, graphs hard

Multi-user editing:

Text hard, graphs easy

Building together

Building Together



TRAIL-SEEKER.COM



It's easy to build a simple modeling tool

- 6 ways to get the tools we need for DSM
 1. Write own modeling tool from scratch
 2. Write own modeling tool based on frameworks
 3. Metamodel, generate modeling tool skeleton, add code
 4. Metamodel, generate full modeling tool over a framework
 5. Metamodel, output configuration for generic modeling tool
 6. Integrated modeling and metamodeling environment

- 1-3 = Coding
 - Language expressed throughout the code
- 4-6 = Language Workbench
 - Language expressed as data

- 4: Language data is transformed into code
- 5: Language data is copied
- 6: Language data is live

Current Active Graphical LWBs

■ **MetaEdit+**

- average 100 commits/month over last year

■ **Sirius (/Obeo)**

- average 85 commits/month over last ½ year

■ **GMF**

- average 18 commits/month over last year

■ **Graphiti - incubation**

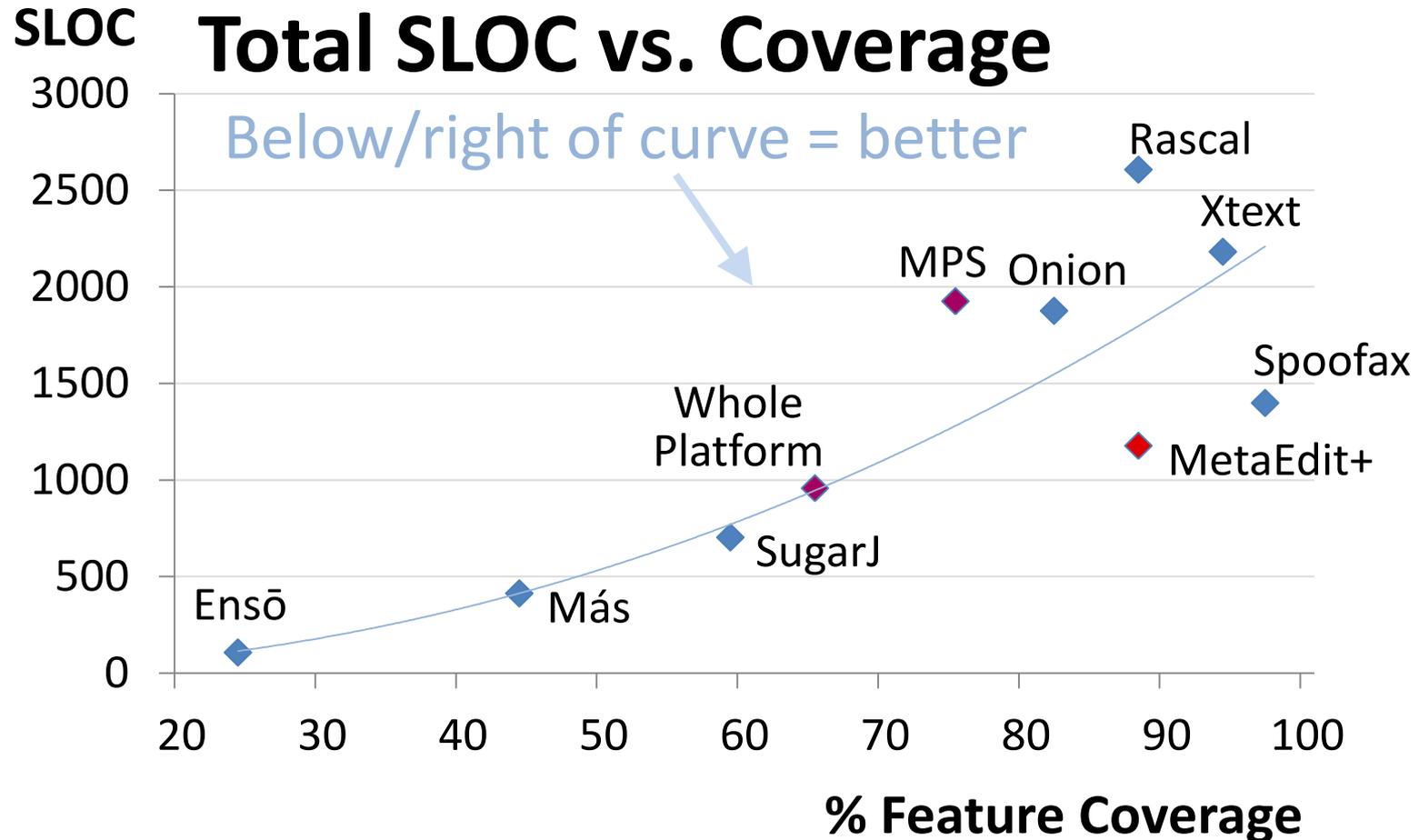
- average 8 commits/month over last year

■ **Spray - labs**

- no release since June '13, last code change Nov '13
- average 0.5 commits/month over last year

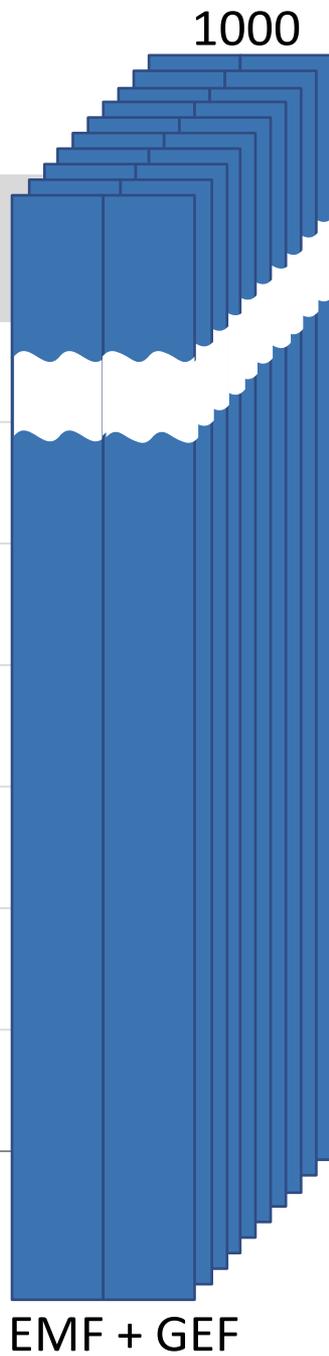
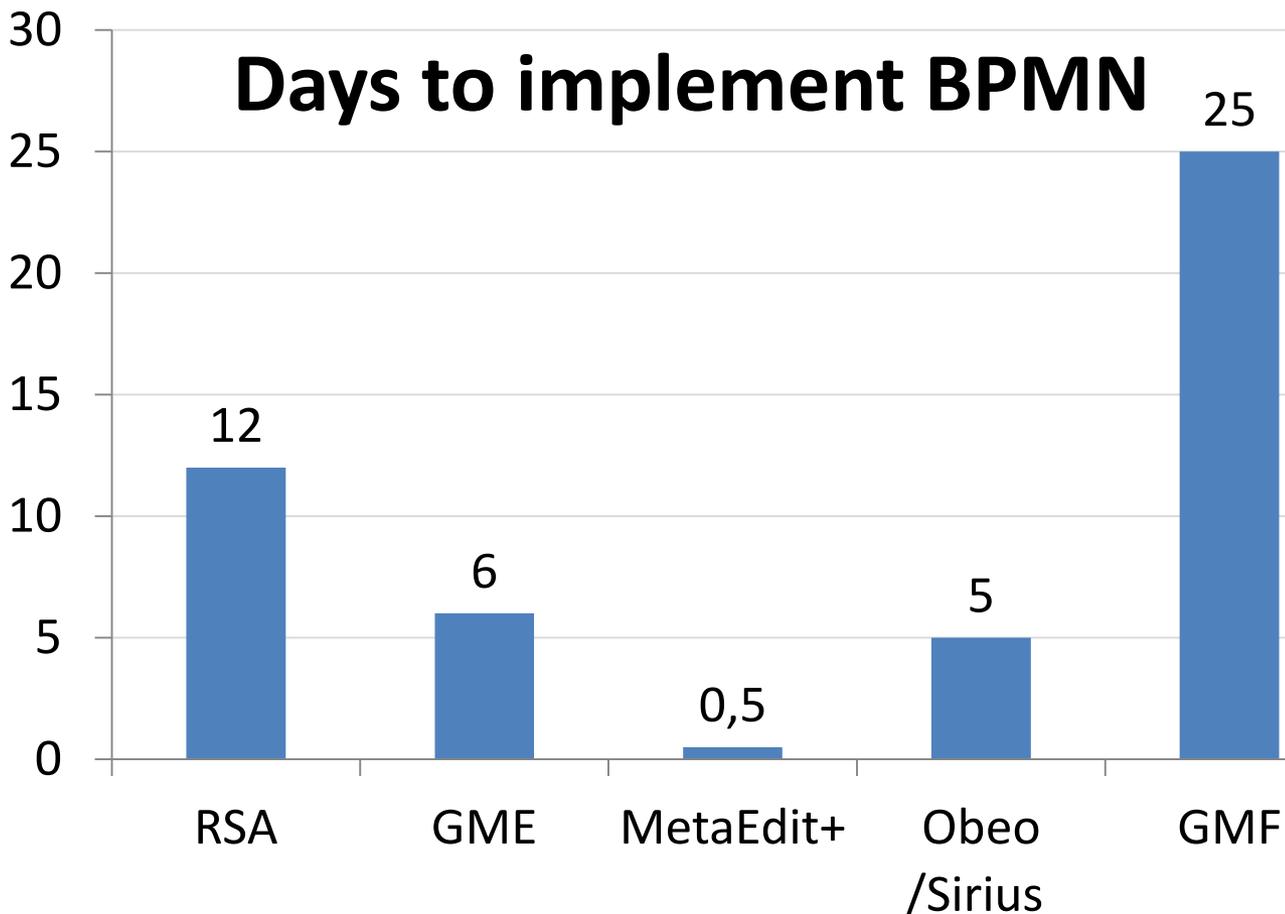
LWB comparison: LWC 2013

<http://erdweg.org/publications/language-workbench-state.pdf>



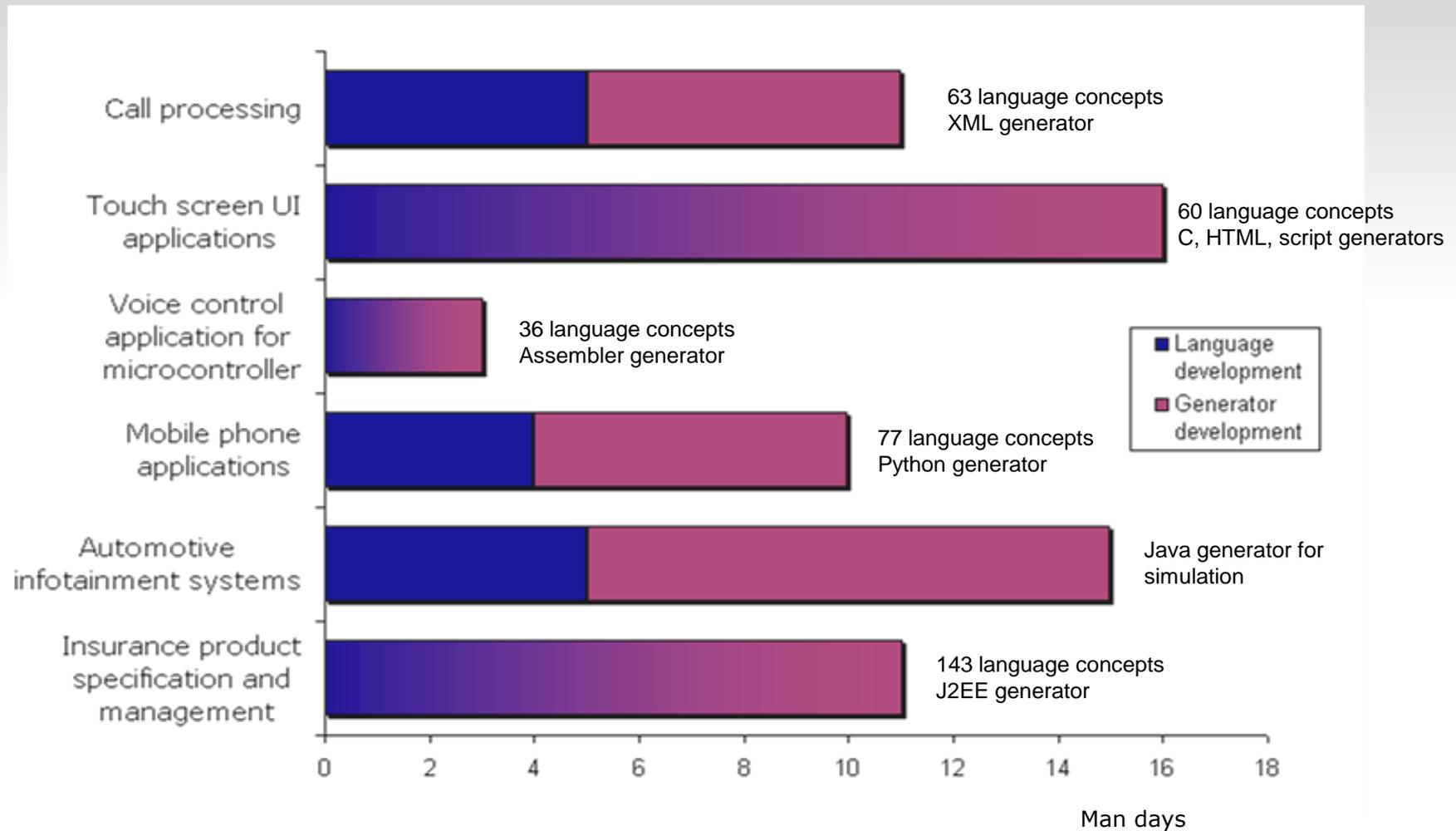
Graphical LWB comparison: El Kouhen et al. 2012

tinyurl.com/gerard12



DSM Solution Implementation Time

<http://metacase.com/blogs/stevek/blogView?entry=3446309083>





Text DSL != Graphical DSL

Language user:

Graphical easier

Language developer:

GMF/Obeo: graphical harder

MetaEdit+: graphical easier

Language wb developer:

Graphical harder

Building together



Text DSL != Graphical DSL

Alan Kay:

"The computer revolution
hasn't happened yet"

The **modeling** revolution
hasn't happened yet

The **text LWB** revolution
hasn't happened yet?