軟體研發的思維與工程

倪文君

GSS Galaxy Software Services 叡揚資訊

The Prologue

FLEXCUBE – FUNCTIONAL ARCHITECTURE

GSS Galaxy Software Services 叡揚資訊

The flexible functional architecture of FLEXCUBE enables it to be deployed across a variety of scenarios depending on the financial institution's business model.

CUSTOMER DELIVERY CHANNELS


Internet


Teller


ATM


POS


Phone


Mobile Phone


PDA

FLEXCUBE@ Internet Banking Suite

- PORTAL
- RETAIL
- CORPORATE
- INVESTOR SERVICES
- BROKER

3rd Party Interfaces

- CRM Systems
- Dealing Systems
- Other Systems

FUNCTIONAL MODULES

Current and Savings Account	Deposits	Loans	Loan Application Processing	Loan Syndication
Bills and Collections	Letters of Credit	Foreign Exchange	Money Market	Dealer
Investor Services	Cash and Liquidity Management	Asset Management	Derivatives	Securities

COMMON FACILITIES

Signature Verification	Funds Transfer	Standing Instructions	Electronic Messaging System
Workflow Management	Nostro Reconciliation	Fixed Assets Management	Expense Processing

CORE MODULE

- Risk Management
- Clearing
- Management Information System
- Customer Information System
- General Ledger

INSTITUTIONAL DELIVERY

- ACH
- SWIFT
- GIRO
- CENTRAL BANK REPORTING
- ZENGIN
- CLEARSTREAM /CEDEL
- RATE FEEDS

REVELEUS

Technology Solutions

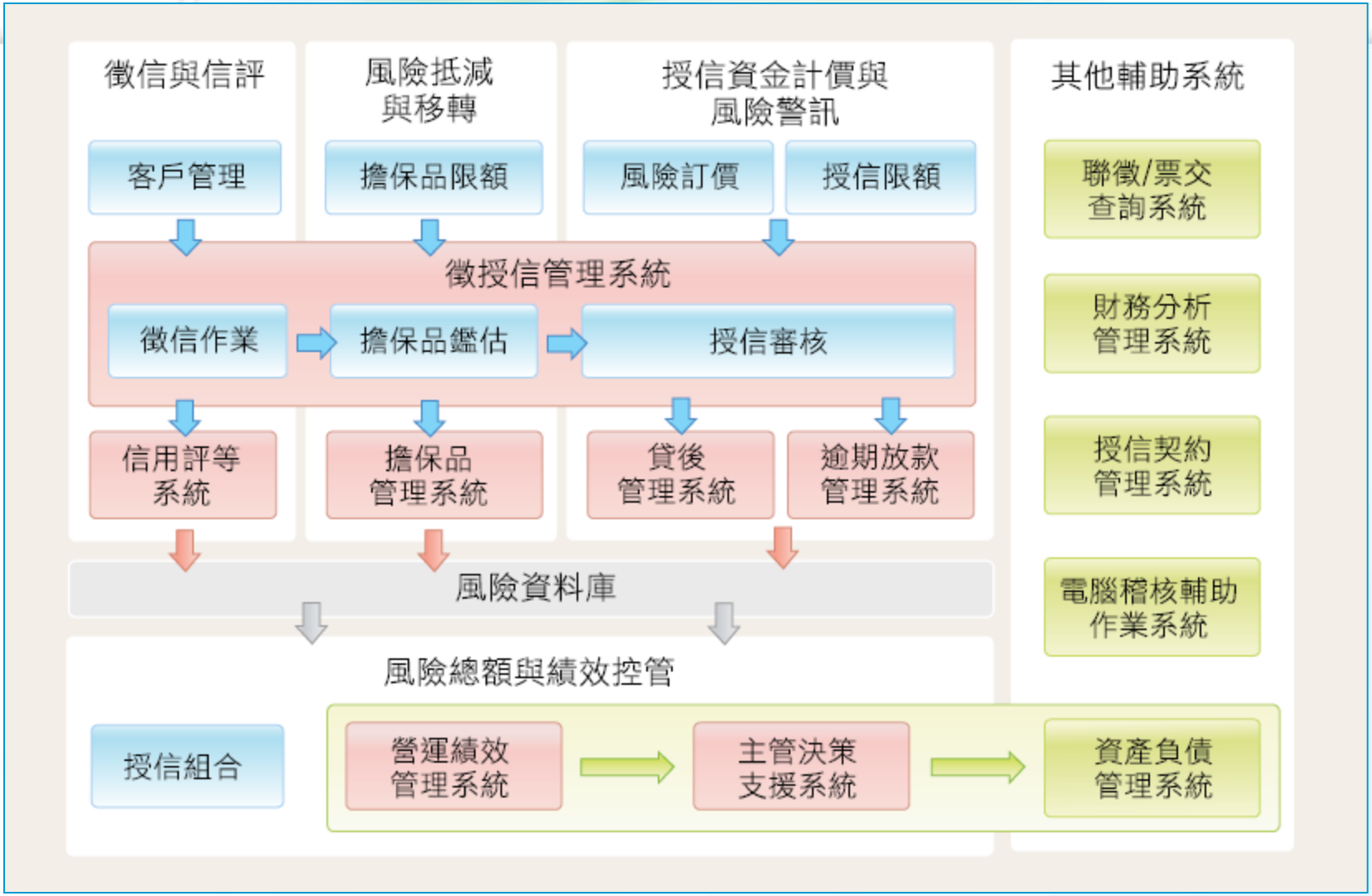
- Reveleus Data Integrator
- Reveleus Mart Builder
- Reveleus Insight
- Reveleus Active Portal

Business Analytics

- Customer Relationship
- Risk Management
- Performance Measurement

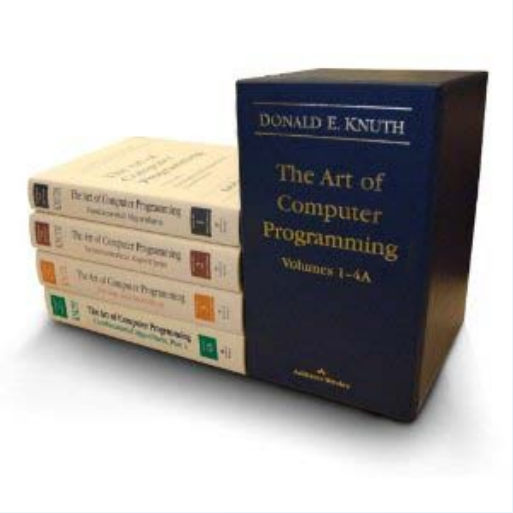
銀行信用風險資訊解決方案

GSS Galaxy Software Services 叡揚資訊



代表性客戶：第一銀行，台北富邦銀行，上海商銀，中國信託，日盛商銀

[...] SOFTWARE IS HARD. From now on, I shall have significantly greater respect for every successful software tool that I encounter. -- Donald Knuth

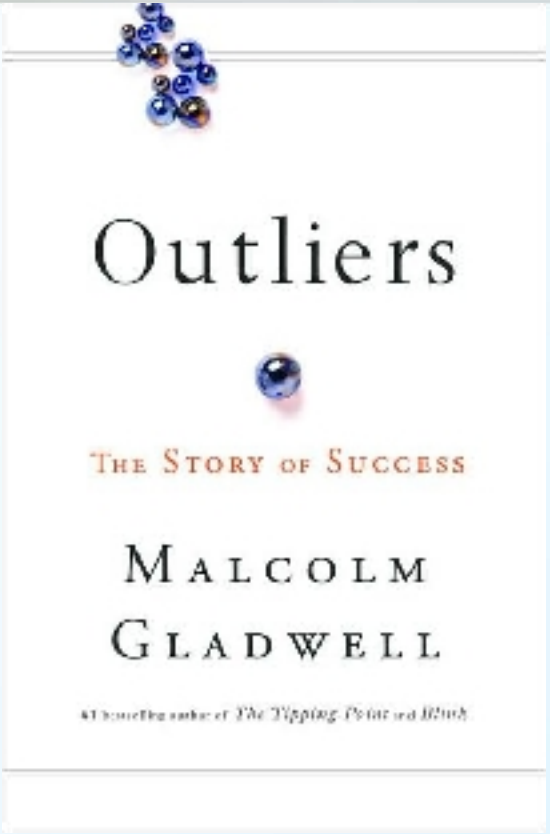


Software development is ***not*** hard, as long as you don't have to change code.

How to become good at it

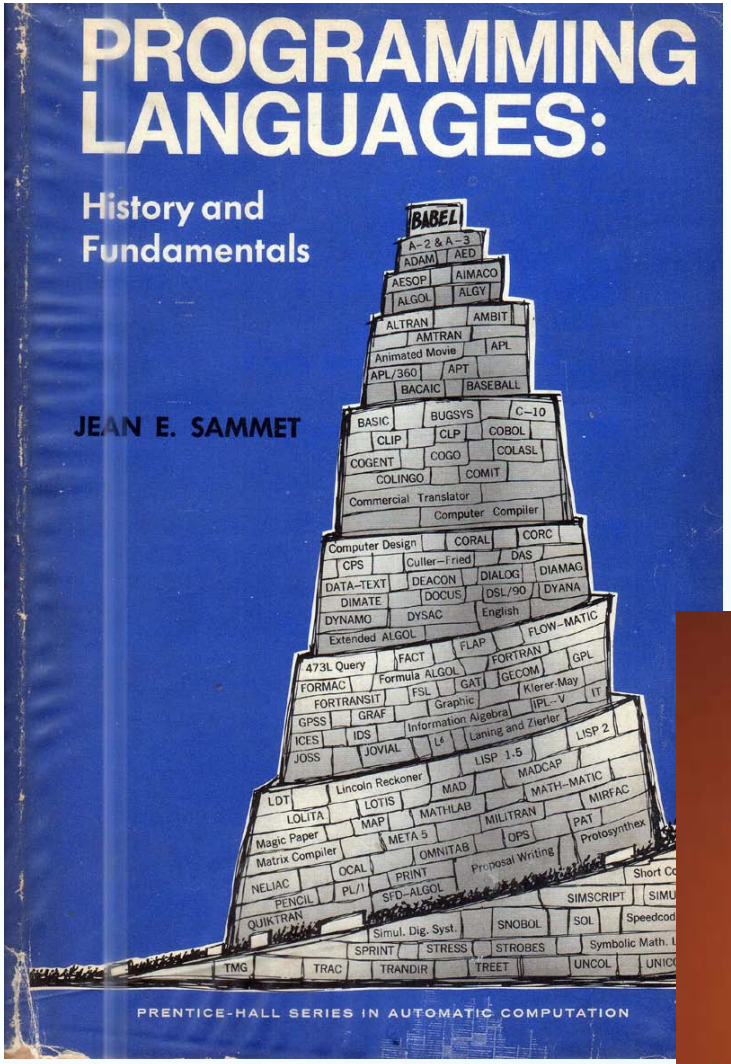
The Efforts

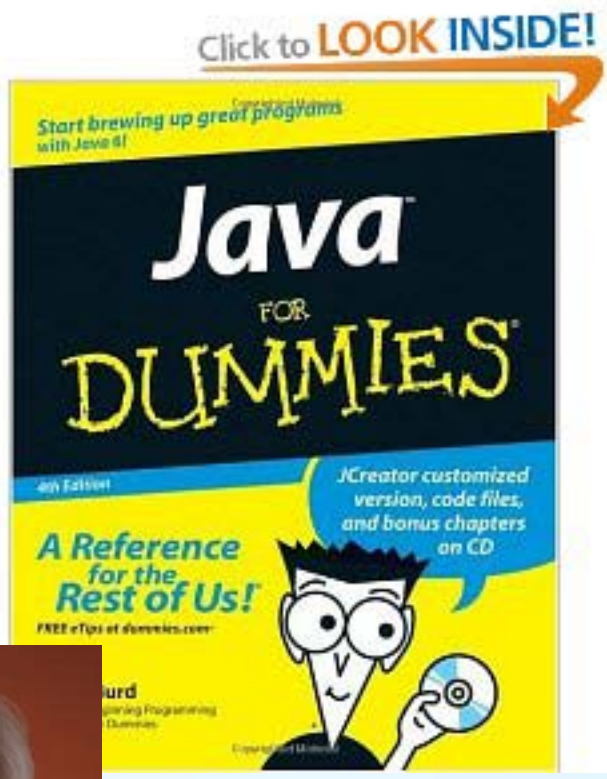
It takes **10** years or **10,000** hours



Mozart, The Beatles, and Bill Joy

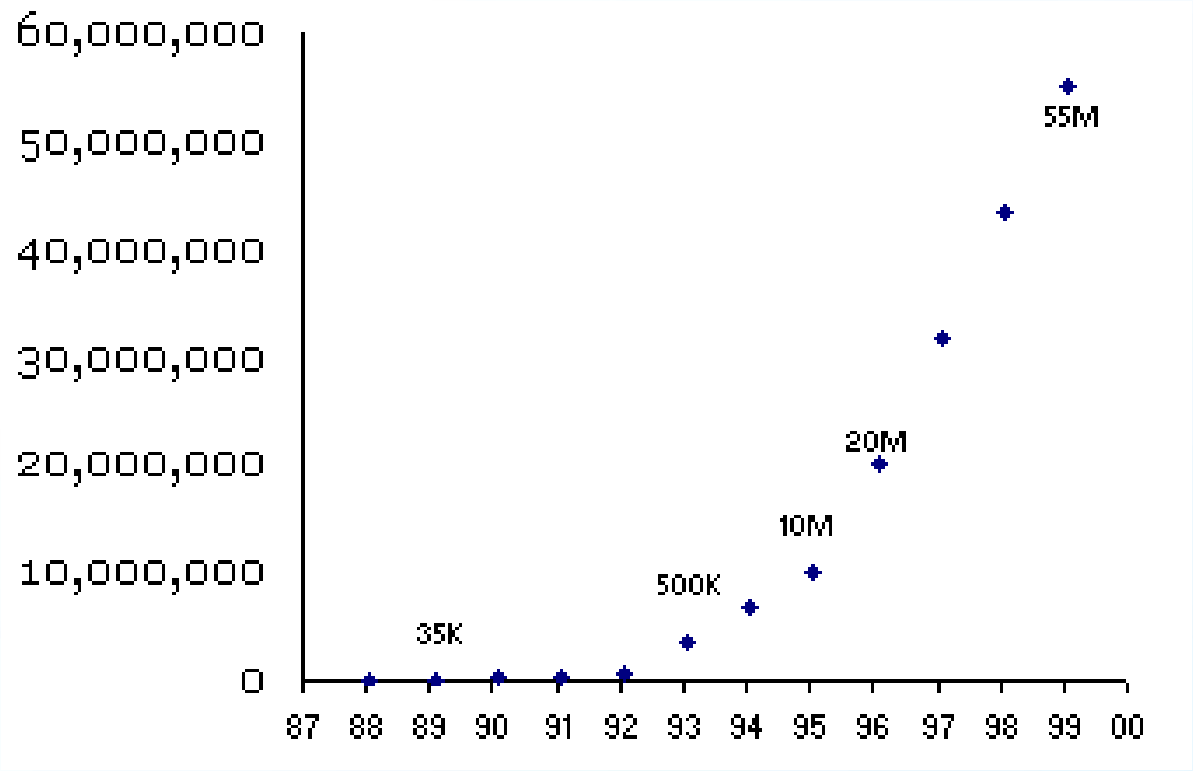
It takes **10** years





Peter Norvig

It takes **10** years



When Lotus Notes finally shipped, it has been developed for 5 years (1984-1989).

How we develop people

The Team

Team Spirit

- Give the team a name (ours is Delta)
- Feel and perform like an elite team
- Responsive to changes in thinking
- Always fearless in attacking problems

Team Size

GSS 叡揚資訊 Galaxy Software Services

- Amazon Rule: two pizzas can feed the team (2 to 8)
- Communication links: $N(N-1)/2$
- Communication is the most important one
- Large projects: partition the team
- Risk: Hard to replace leaving members

Qualification

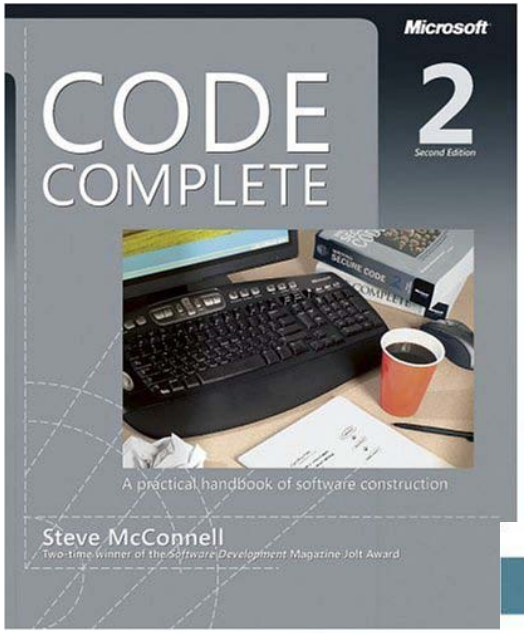
GSS Galaxy Software Services 叡揚資訊

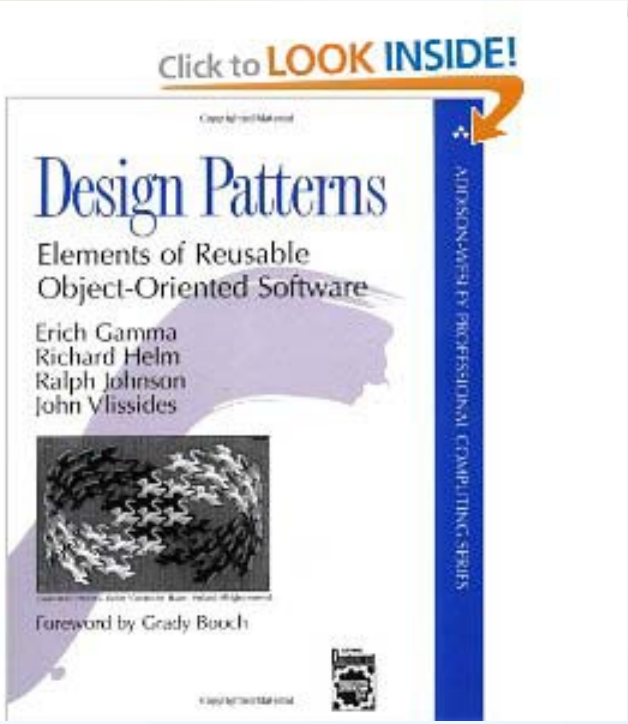
- Learning speed and curiosity
- Team Player
  - Willing to communicate
  - Willing to take responsibility
- English
  - Good reading a must
  - Should improve writing within one year
- Math: better than average
- Graduate degree **not** required

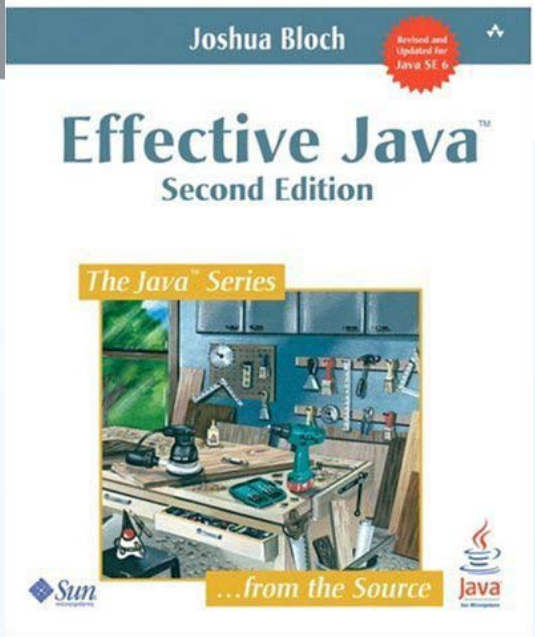
Team Learning

- Reviews
  - Analysis
  - Design
  - Code
- Basic Readings
- Advanced Readings
- *What's New* Sessions

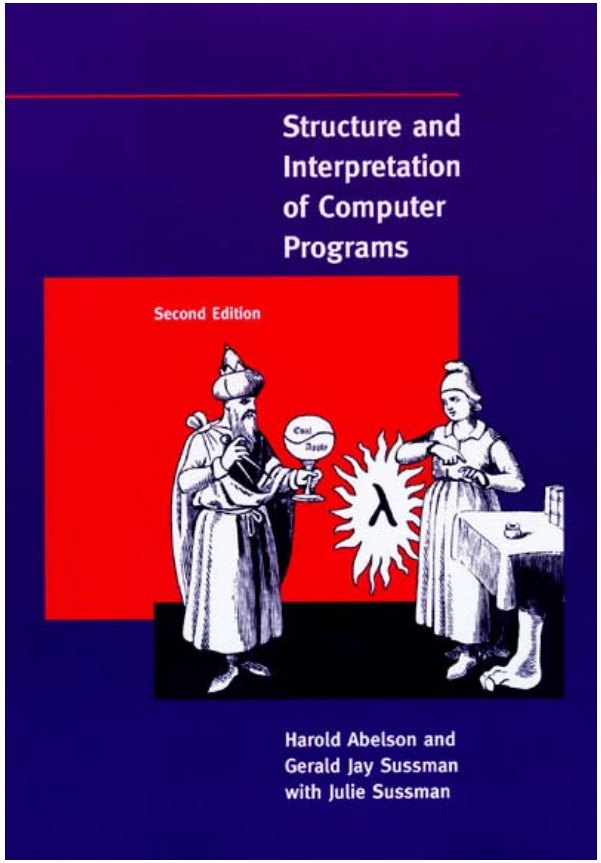
Basic Readings

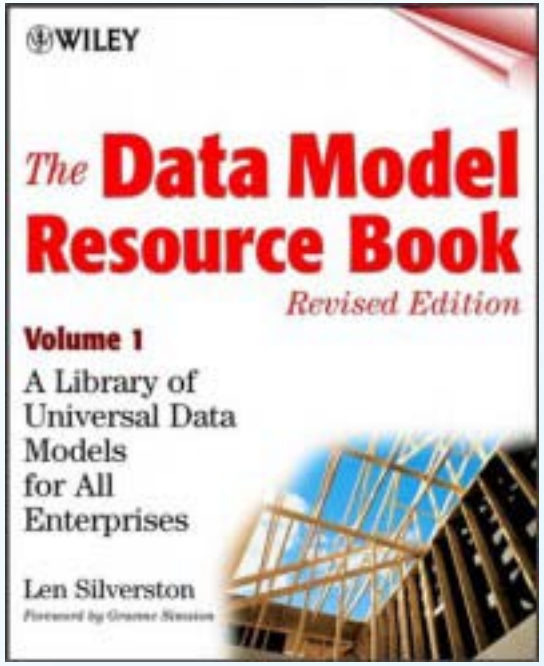






Advanced Readings





How we build the system

The Process

Tools

- Code repository
  - subversion now
  - May migrate to git
- Auto-build server: Jenkins
  - Static analysis of code quality
  - Unit tests
- JIRA for issues
- teamKube for planning and tasking

Development Activities

GSS 叡揚資訊 Galaxy Software Services

- Devs are required to write daily work notes
- Fear not checking in code
- Devs are responsible for unit tests
- Comprehensive plans are made before development
- Only essential documents are produced
- Dedicated QA engineers for UI and integration
- Dedicated UI designer

Releases

- Major releases every 3 months
- Single current version
  - Branches are for experiments
  - Hard to merge in subversion
- Patching automagically
  - Patching code is the easy part
  - Patching environment and data is hard

What's wicked

The ~~Problem~~ Complexity

When we had no computers, we had no programming problem either.

When we had a few computers, we had a mild programming problem.

Confronted with machines a million times as powerful, we are faced with a gigantic programming problem.

Sources of Complexity

GSS Galaxy Software Services 叡揚資訊

- Requirement
- System Analysis and Design
- Software
- Validation and Verification
- Operations

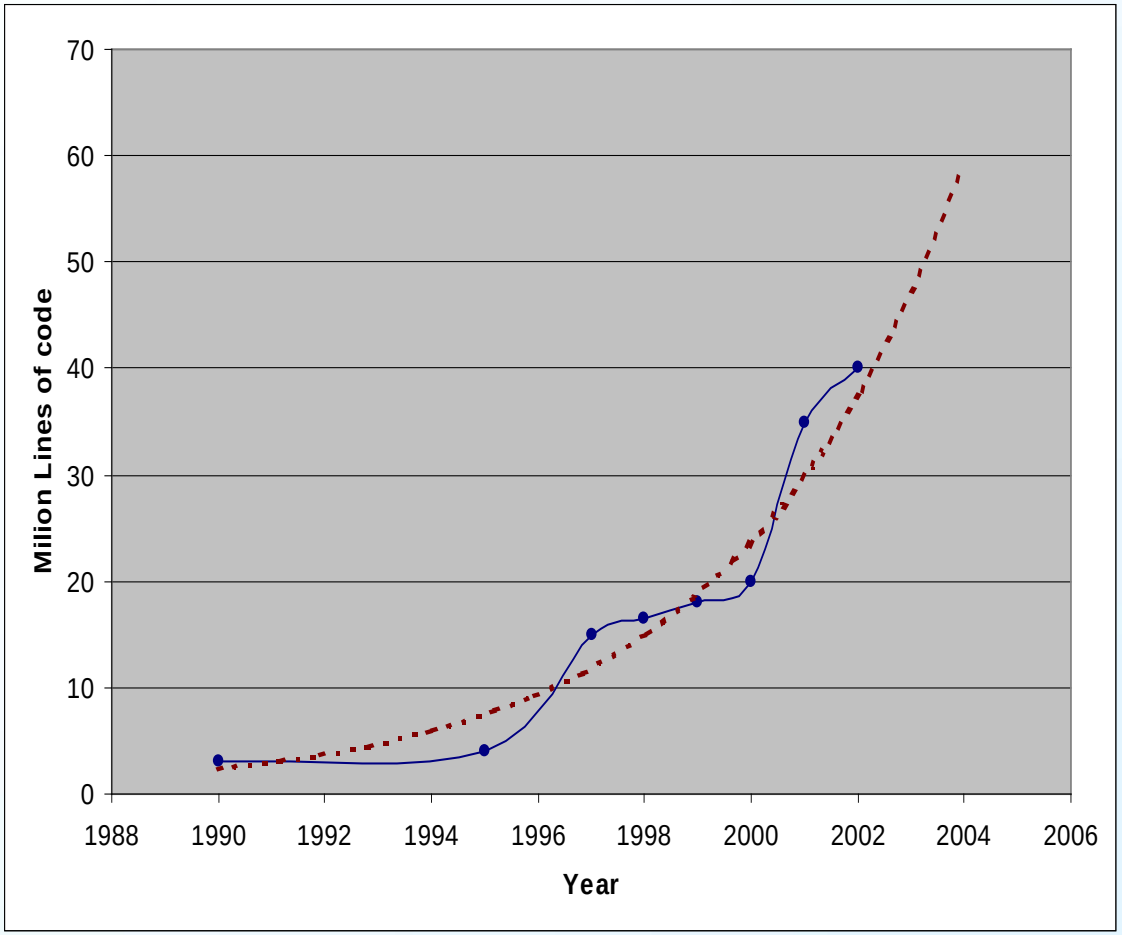
Windows Operating Systems

GSS 叡揚資訊 Galaxy Software Services

Year	Operating System	SLOC (Million)
1993	Windows NT 3.1	4-5
1994	Windows NT 3.5	7-8
1996	Windows NT 4.0	11-12
2000	Windows 2000	more than 29
2001	Windows XP	45
2003	Windows Server 2003	50

A complexity measure of Windows family

1990	Win 3.1
1995	Win NT
1997	Win 95
1998	NT 4.0
1999	Win98
2000	NT 5.0
2001	Win2k
2002	XP



Non-Windows Operating Systems

GSS Galaxy Software Services 叡揚資訊

Operating System	SLOC (Million)
Debian 2.2	55-59
Debian 3.0	104
Debian 3.1	215
Debian 4.0	283
Debian 5.0	324
OpenSolaris	9.7
FreeBSD	8.8
Mac OS X 10.4	86
Linux kernel 2.6.0	5.2
Linux kernel 2.6.29	11.0
Linux kernel 2.6.32	12.6
Linux kernel 2.6.35	13.5

Size Comparisons of Embedded Software

GSS 叡揚資訊 Galaxy Software Services

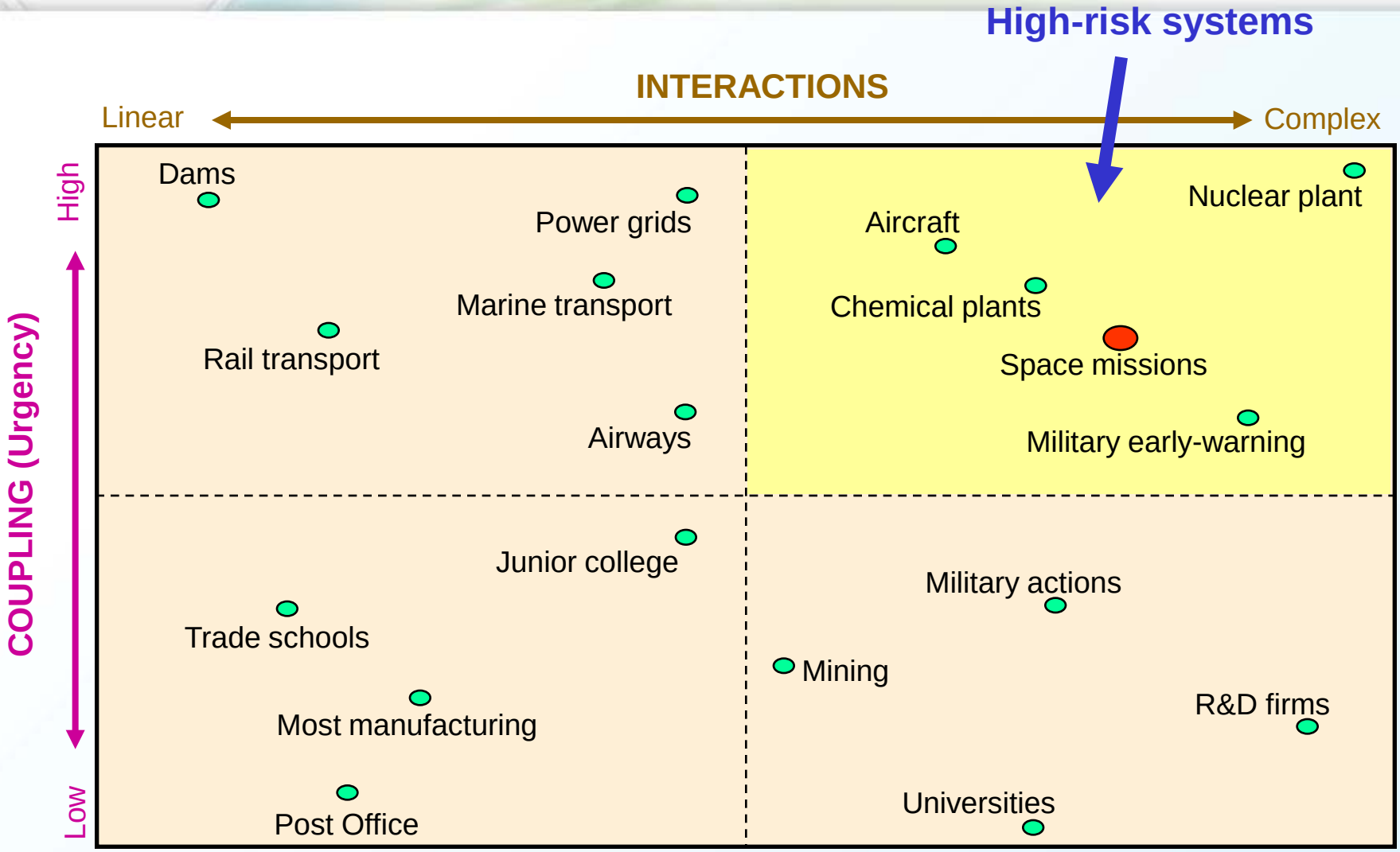
System	Lines of Code	Language
Mars Reconnaissance Orbiter	545K	C
F-22 Raptor	2.5M	Ada (90%)
Seawolf Submarine Combat System AN/BSY-2	3.6M	Ada
Boeing 777	4M	Ada
Boeing 787	7M	Ada (largely)
F-35 Joint Strike Fighter	19M	C and C++
Typical GM car in 2010	100M	MISRA-C for critical systems

The Complexity of teamKube

GSS Galaxy Software Services 叡揚資訊

files	language	blank	comment	code
3600	Java	76,108	71,633	**289,953**
1018	Javascript	11,633	11,919	**67,743**
744	HTML	5,839	1,750	**34,869**
364	XML	5,514	6,033	**30,146**
104	CSS	3,888	1,285	**18,597**
25	SQL	1,562	220	6,341
85	JSP	681	588	6,226
73	Ruby	630	389	2,394
8	XSLT	157	98	1,613
28	ActionScript	228	151	1,199
11	MXML	96	65	770
72	Bourne Shell	179	524	730
6	PHP	81	86	566
59	DOS Batch	129	207	477
3	XSD	23	34	343
3	DTD	22	40	201
1	Bourne Again	10	18	62
	SUM	106,870	95,040	**462,230**

Complex interactions and high coupling raise risk of design defects and operational errors



Source: Charles Perrow, “Normal Accidents: Living with High-Risk Technologies”, 1984.

How we set design constraints

The Architecture

Architecture is the set of *early* decisions that are extremely costly to change later.

Some Design Guidelines

Changing data is easier than changing the code

Always establish abstraction barrier first

- Interface-first in Java/C#

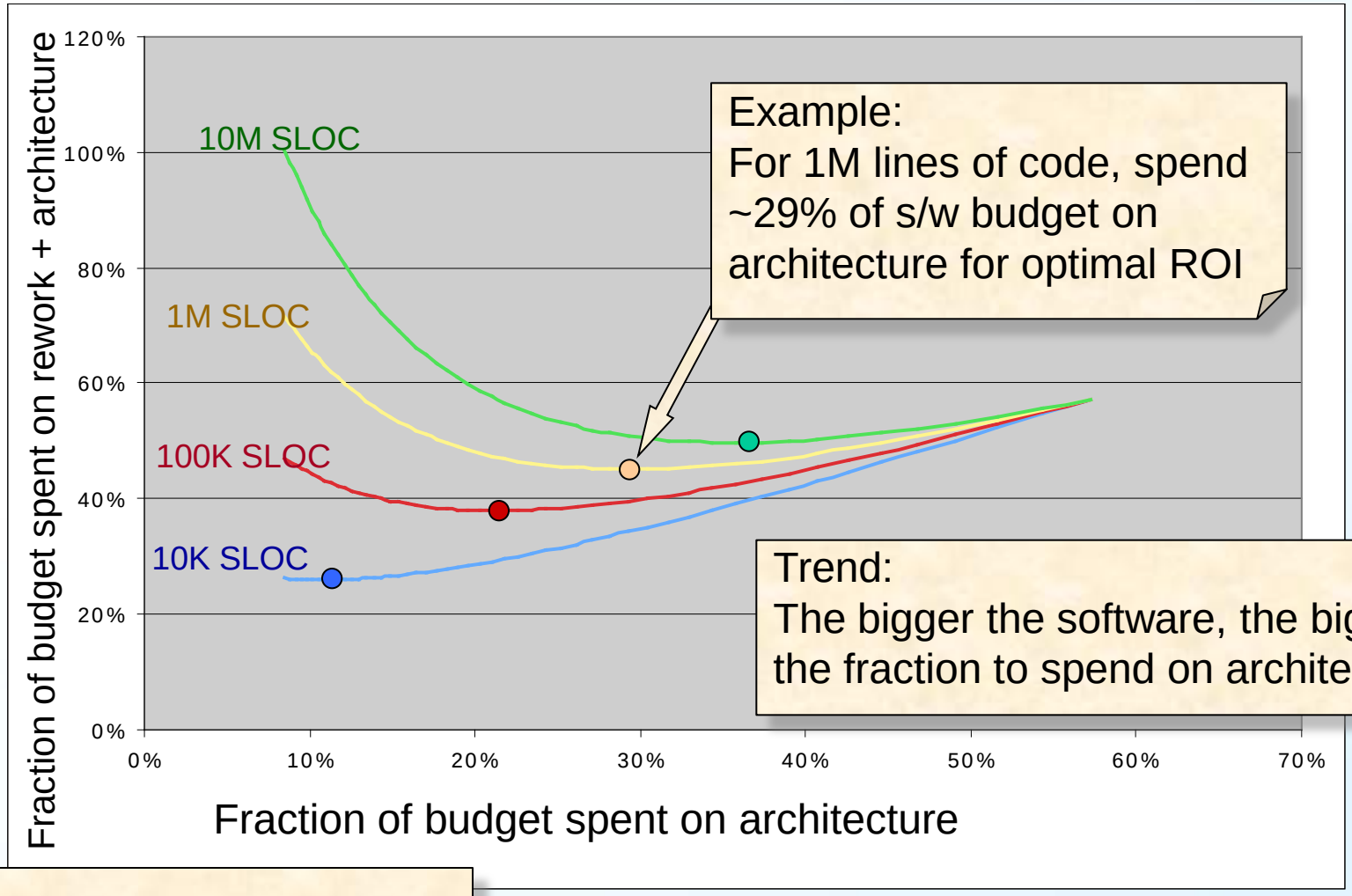
Try to accommodate possible future changes

- One man's constant is another man's variable

Revisit and improve aggressively

Architecture Investment “Sweet Spot”

Predictions from COCOMO II model for software cost estimation



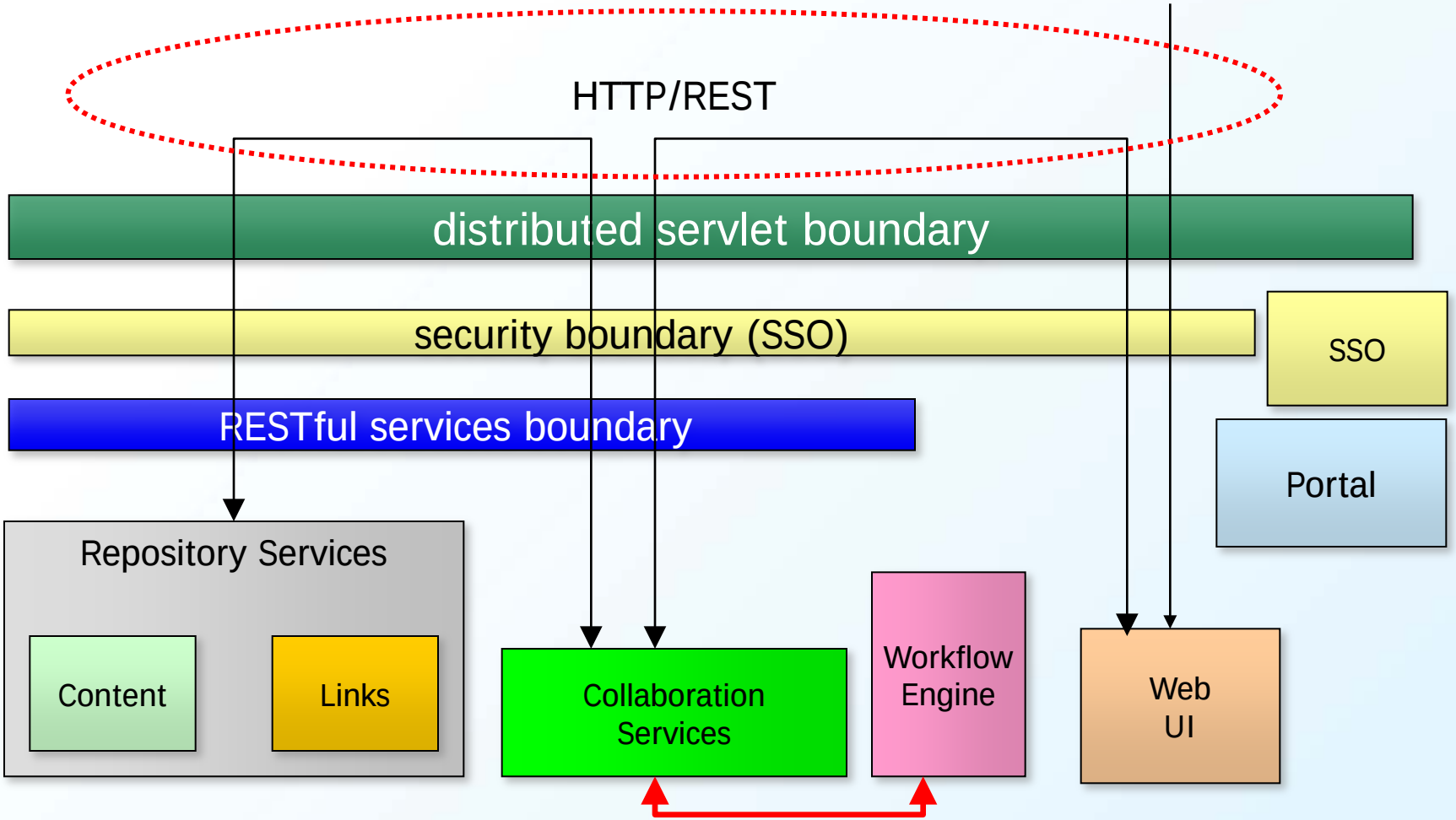
Example:
For 1M lines of code, spend ~29% of s/w budget on architecture for optimal ROI

Trend:
The bigger the software, the bigger the fraction to spend on architecture

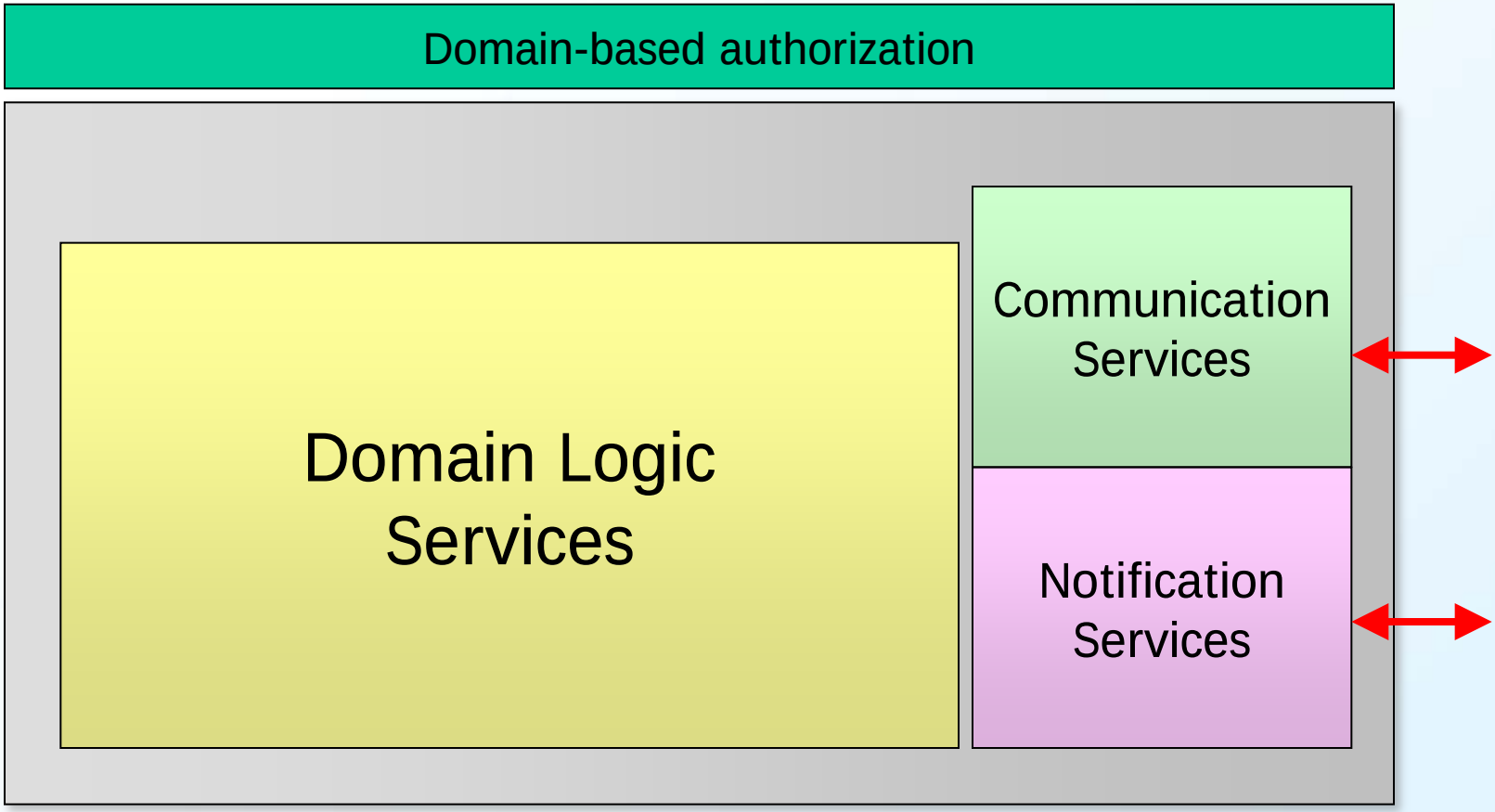
Note:
Prior investment in a reference architecture pays dividends

Source: Kirk Reinholtz, JPL

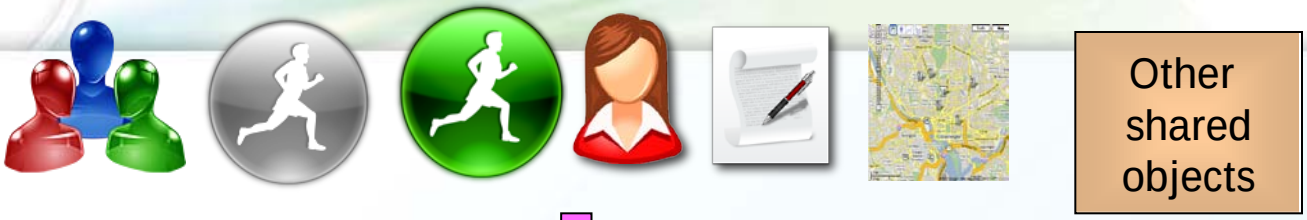
teamKube's Architecture



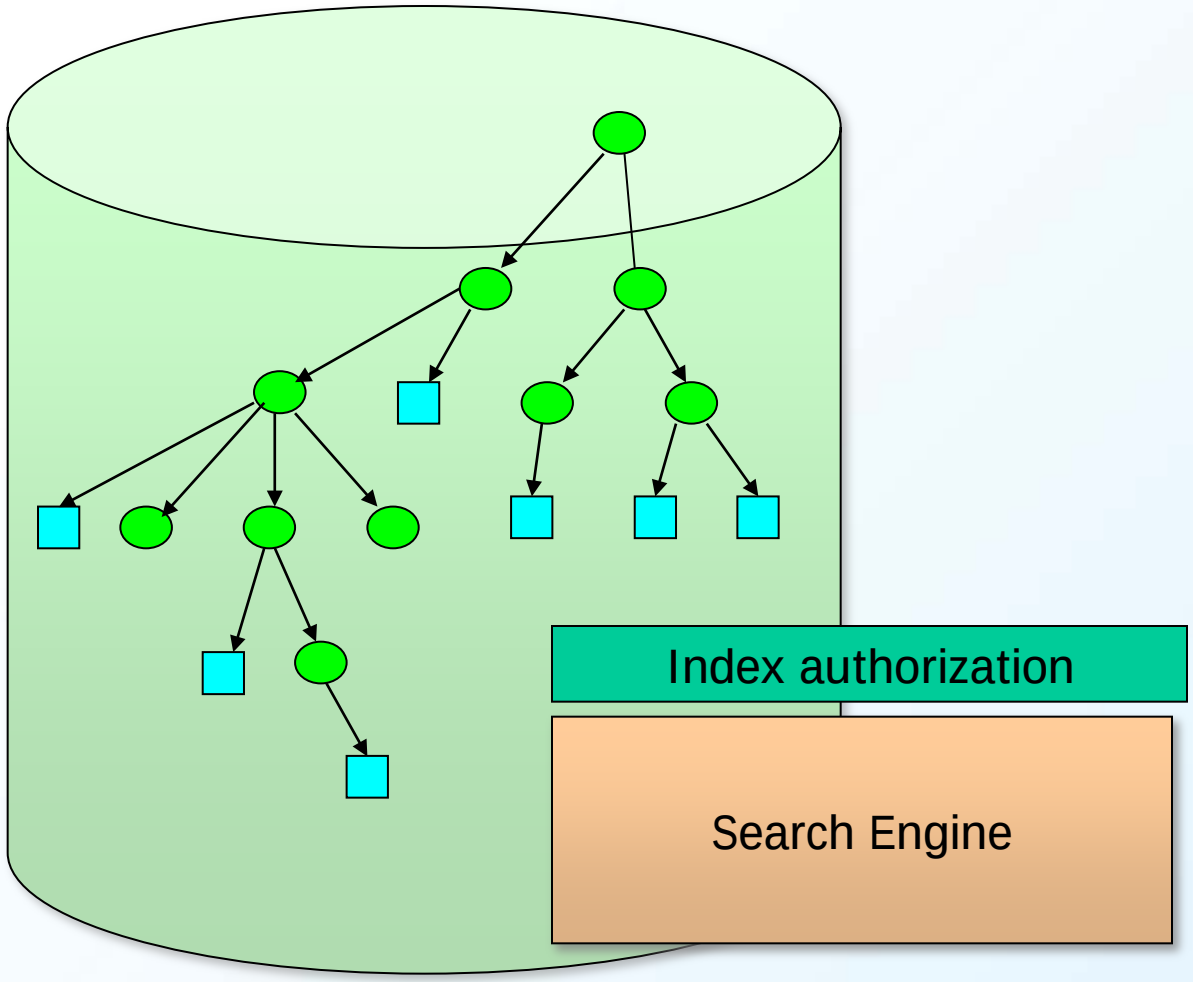
Collaboration Services



Repository Services



Content



What to expect next

The Data

1 TB	260,000 songs
20 TB	Photos uploaded to facebook/month
120 TB	All data/image collected by Hubble Telescope
260 TB	Data collected by Large Hadron Collider (LHC) each week
460 TB	All the digital data from national climate center
530 TB	All the video on YouTube
600 TB	All US census data from 1790 to 2000
1000 TB	Data processed by Google per 72 minutes

1 Petabytes = 1024 Terabytes

GSS Galaxy Software Services 叡揚資訊

1 million HD movies = 20 years playing

100 million = 2,000 years of MP3 songs
(10 megabytes or 128 kbits/sec)

It takes 12 days to restore data with 1 GB/s

My Life

GSS Galaxy Software Services 叡揚資訊

Human Input	Per hour	lifetime
read	100 KB	25 GB
hear	40 MB	10 TB
see	2 GB	8 PB

Canon 5D Mark II – HD Video

- 1920 x 1680 (30 fps)
- 333 MB /minute
- 20 GB/hr
- 80 PB/lifetime

All of Our Lives

GSS 叡揚資訊 Galaxy Software Services

Zetta

Everything recorded

Exa

All books multimedia

Peta

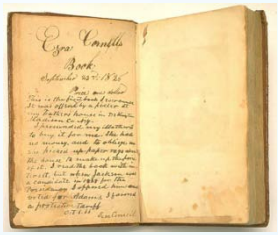
Tera

The LIBRARY of CONGRESS

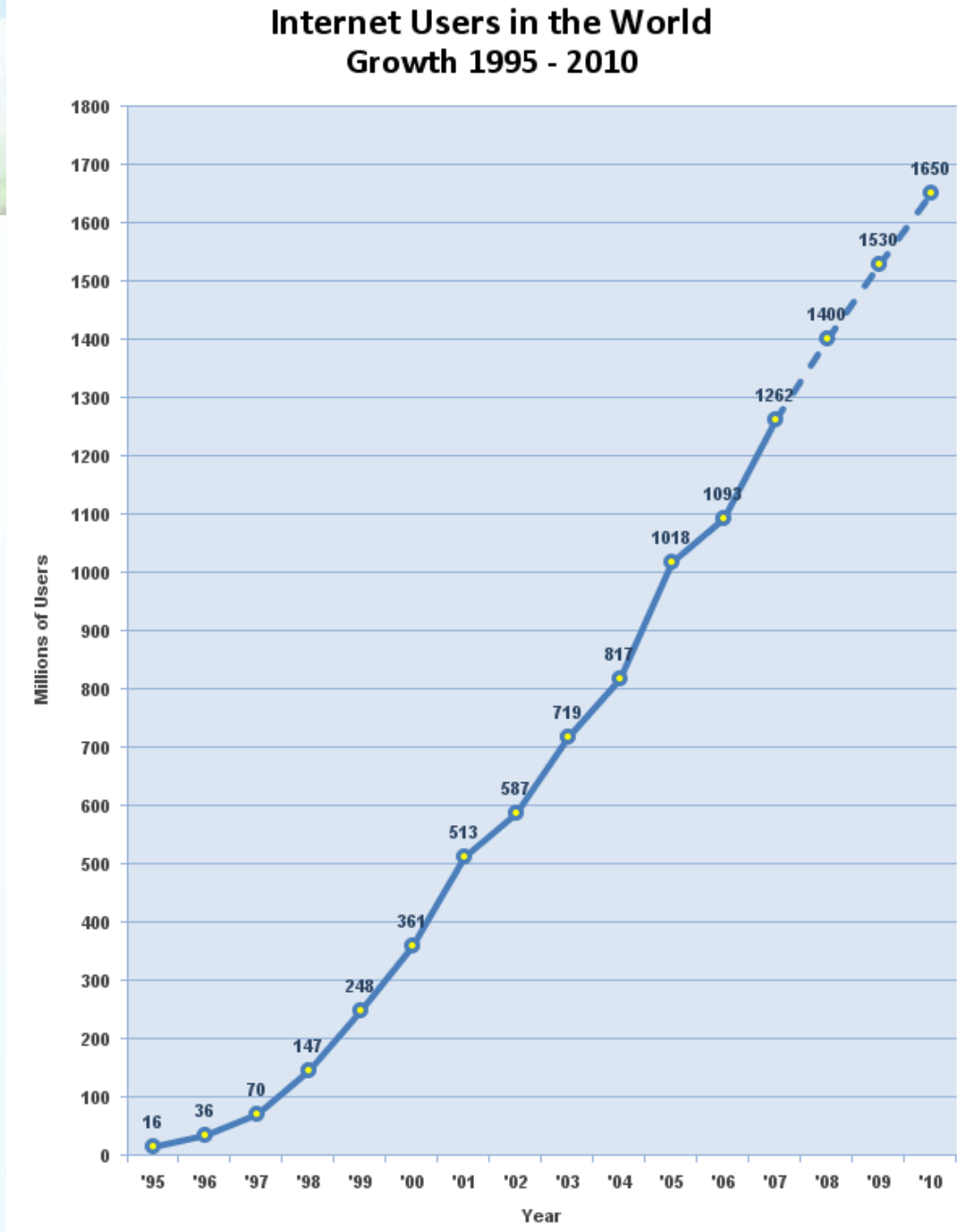
Giga

Mega

Kilo



We're drowning in *users*



Source: www.internetworldstats.com - January, 2008

Facebook in 20 minutes (2010)

GSS Galaxy Software Services 叡揚資訊

- Shared links: 1,000,000
- Tagged photos: 1,323,000
- Event invites sent out: 1,484,000
- Wall Posts: 1,587,000
- Status updates: 1,851,000
- Friend requests accepted: 1,972,000
- Photos uploaded: 2,716,000
- Comments: 10,208,000
- Message: 4,632,000


IN 60 SECONDS...
1 NEW DEFINITION IS ADDED ON urban
1,600+ READS ON Scribd.
13,000+ HOURS MUSIC STREAMING ON PANDORA
12,000+ NEW ADS POSTED ON craigslist
370,000+ MINUTES VOICE CALLS ON skype
98,000+ TWEETS
20,000+ NEW POSTS ON tumblr.
THE LARGEST SOCIAL READING PUBLISHING COMPANY!!
320+ NEW twitter ACCOUNTS
13,000+ iPhone APPLICATIONS DOWNLOADED
100+ NEW Linked in ACCOUNTS
1 associatedcontent NEW ARTICLE IS PUBLISHED
THE WORLD'S LARGEST COMMUNITY CREATED CONTENT!!
QUESTIONS ASKED ON THE INTERNET...
100+ Answers.com
40+ YAHOO! ANSWERS
6,600+ NEW PICTURES ARE UPLOADED ON flickr
600+ NEW VIDEOS
50+ WordPress DOWNLOADS
25+ HOURS TOTAL DURATION
70+ DOMAINS REGISTERED
695,000+ facebook STATUS UPDATES
125+ PLUGIN DOWNLOADS
60+ NEW BLOGS
168 MILLION EMAILS ARE SENT
694,445 SEARCH QUERIES
1,700+ Firefox DOWNLOADS
79,364 WALL POSTS
1,500+ BLOG POSTS
510,040 COMMENTS
Google
Google Search
GO-Globe.com

Complex Data

GSS Galaxy Software Services 叡揚資訊

- Unstructured: Microsoft Office files
- Semi-structured: XML, HL7
- Complex structured: Hierarchical XML (ACORD)

teamKube's Elements

Places

Documents

Activities

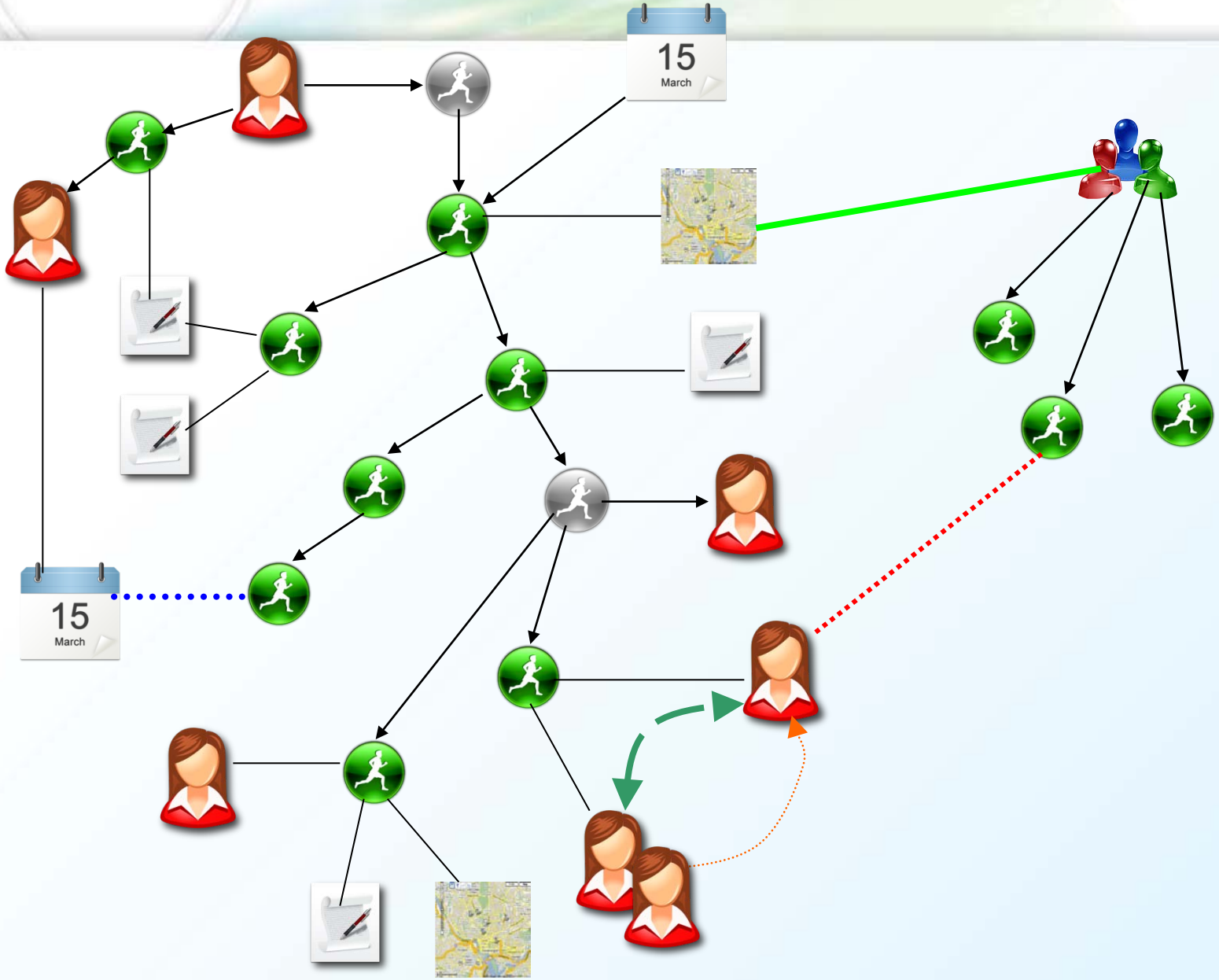
Plans/To dos

Groups

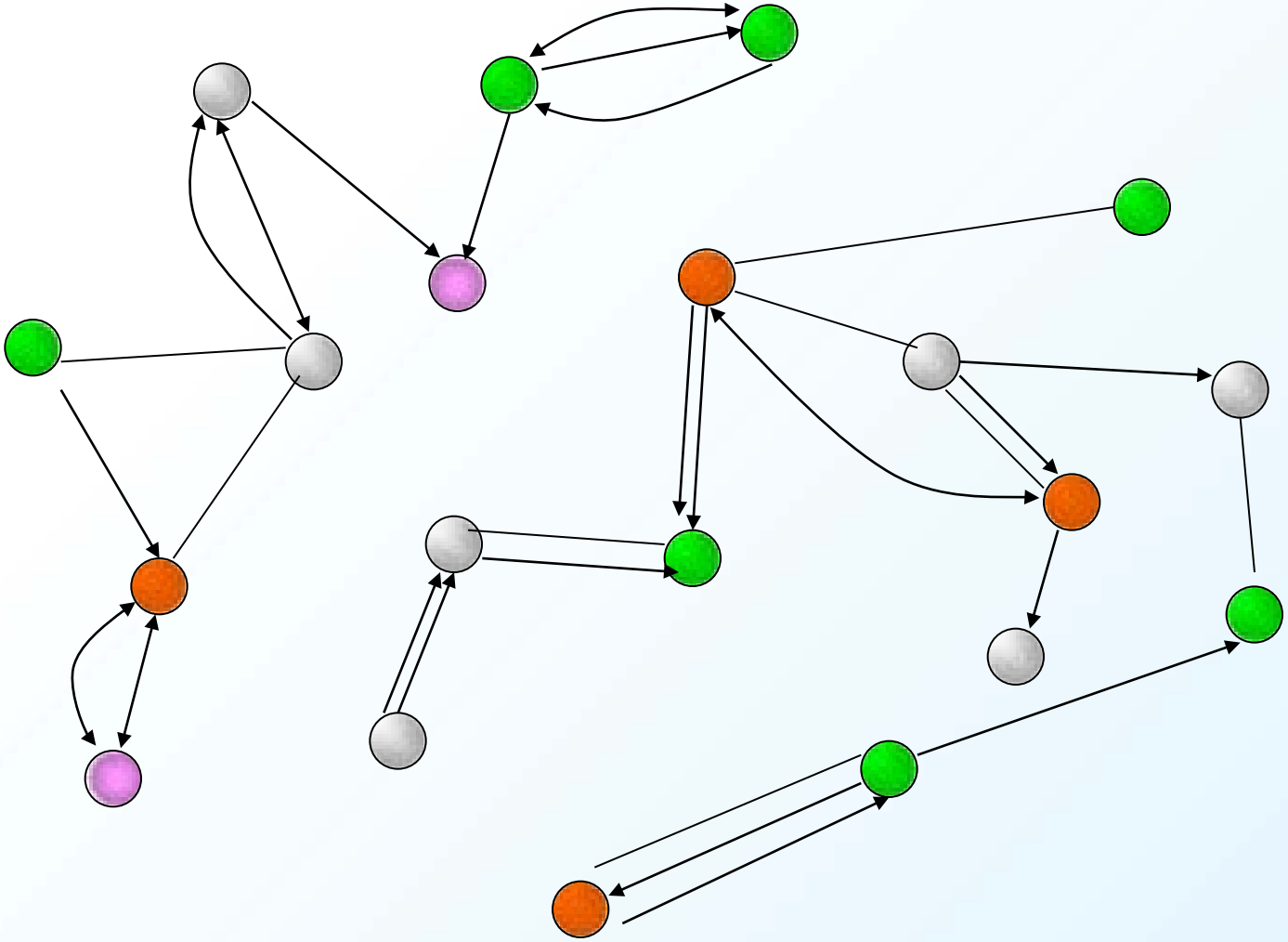
Person

Time

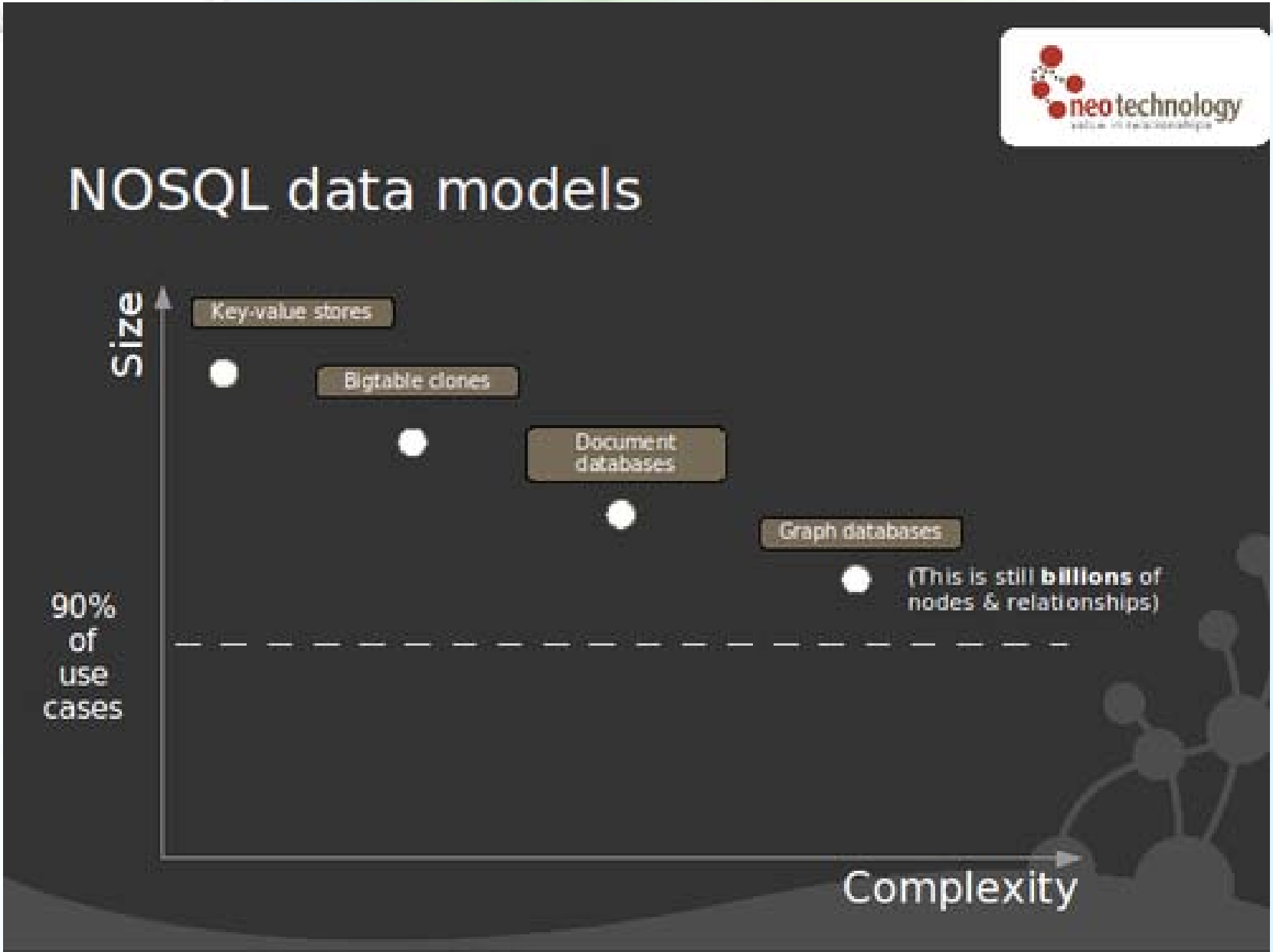
Data in teamKube



Abstract form: multigraph



The Next Frontier of teamKube



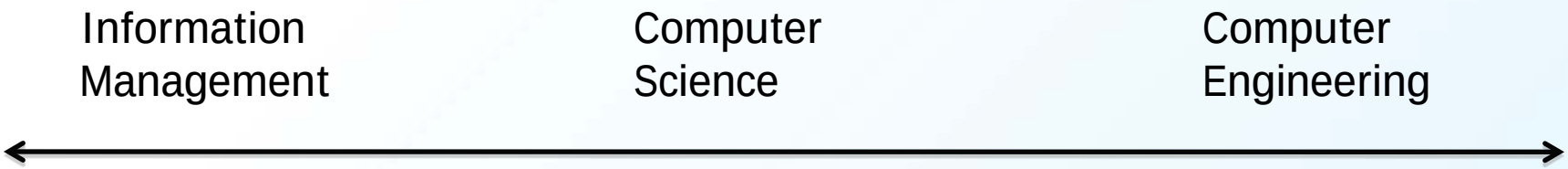
Does Computer Science Matter?

Regarding Computer Science

Computer science is no more about computers than astronomy is about telescopes.
– Edsger Dijkstra

Computer Science is embarrassed by the computer.
-- Alan Perlis (Epigram 112)

The Spectrum



Information Management

- Databases and applications
- Business Applications
  - CRM (Customer Relationship Management)
  - ERP (Enterprise Resource Planning)
  - HCM (Human Capital Management)
  - Financial Software
  - Office Automation

The Core of Computer Science

- Algorithms
- Discrete Math
- Theory of Computation
- Programming Languages
- Software Design Principles
- System Software
  - Compiler
  - Operating Systems
  - Concurrent and Distributed Computing

Computer Engineering

- Computer Architecture and Organization
- VLSI
- Networks
- Sensor technologies
- Computer and Communication

Computer Science and teamKube

- Predicate logic and set theory
- Graph theory and algorithms
- Automata theory
- Compiler – parser
- Concurrency and asynchrony
- Software design
  - Object-oriented programming
  - Functional programming

Feedback to Educators

GSS 叡揚資訊 Galaxy Software Services

- Programming languages exposure:
  - Different programming paradigms
  - Lisp is useful
- Heavier and bigger programming projects
- Deep fundamental knowledge will pay back

Q & A