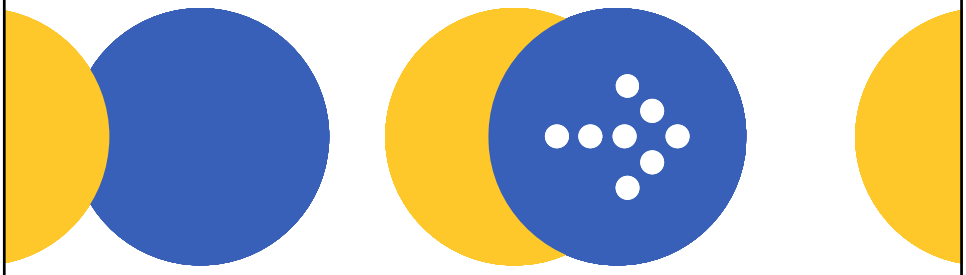
Evolving the Notion of Software Architecture: Three Dimensions

Jan Bosch, Intuit

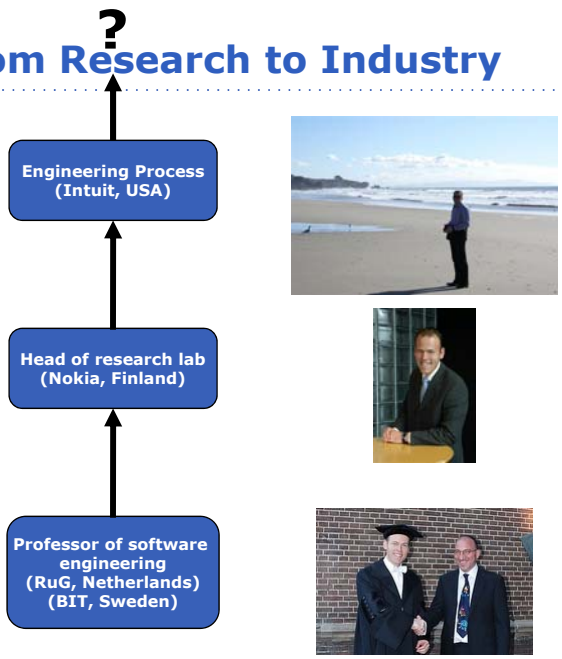
UC Irvine, ISR Research Forum 2008, June 6



Overview

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- Trends
- Software Product Line Architecture
- Architecture for the Ecosystem
- Dynamic Software Architectures
- Conclusion

From Research to Industry



Industrial development

Industrial research

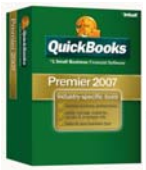
Academia (+ consulting)

intuit

Intuit Company Information

Who We Are…

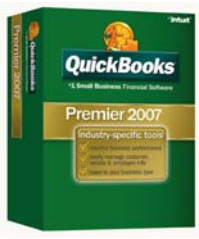
A leading provider of business and financial management solutions

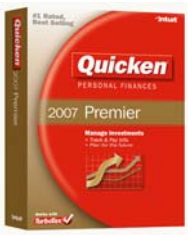


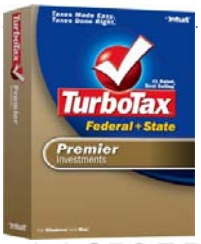
- Founded in 1983
- FY 2007 revenue of $2.67 billion
- Intuit is traded on the NASDAQ: INTU
- Employs more than 8,000 people
- Major offices across the U.S. and in Canada and the United Kingdom

intuit

Great Brands and Great Products







QuickBooks Enterprise Solutions Business Management Software

Quicken Rental Property Manager

Quicken Medical Expense Manager

LACERTE THE FIRST NAME IN TAX SOFTWARE

QuickBooks Simple Start

Credit Card Processing Kit

QuickBooks Customer Manager

ProSeries The Tax Professional's Choice

Quicken Health

TurboTax ItsDeductible

Quicken Bill Pay

QuickBooks Online Edition

QuickBooks Point of Sale

EasyACCT Professional Series

Intuit Payroll Services

Intuit QuickBase

QuickBooks MERCHANT SERVICE

DIGITAL INSIGHT an Intuit company

Intuit MRI Commercial

intuit

Page 5

Fortune top 100 places to work

2008

2007

2006

2005

2004

2003

Page 6

intuit

AMERICA'S MOST ADMIRED COMPANIES 2007 FORTUNE

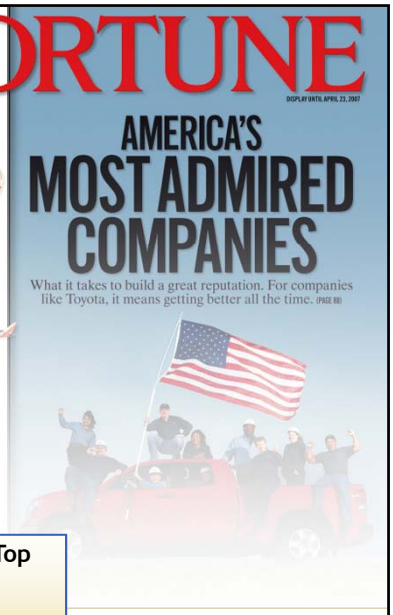
Industry: Computer Software

Most Admired

Rank	Company	Overall score
1	Intuit	7.55
2	Autodesk	7.35
3	Adobe Systems	7.35
4	Microsoft	6.73
5	Electronic Arts	6.65

Contenders

Rank	Company	Overall score
6	Symantec	6.40
7	SAP	6.35
8	Oracle	6.12
9	BMC Software	
10	CA	



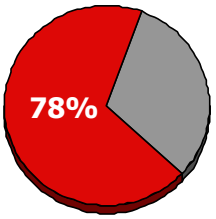
Intuit's Rise to the Top
2007 - No. 1
2006 - No. 1
2005 - No. 1
2004 - No. 3

intuit

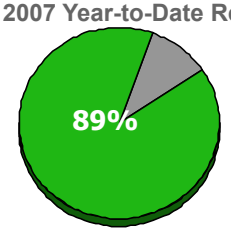
Did You Know?

Intuit has Some of the Strongest Brands in Business.

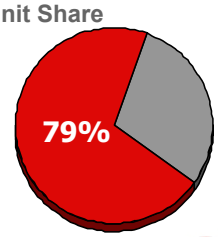
FY 2007 Year-to-Date Retail Unit Share



Quicken



QuickBooks



TurboTax

- 15 million Quicken customers
- Nearly 7 million small businesses are Intuit customers
- More than 14 million federal desktop and Web TurboTax units (Tax Year 2006)

Source: NPD, company estimates

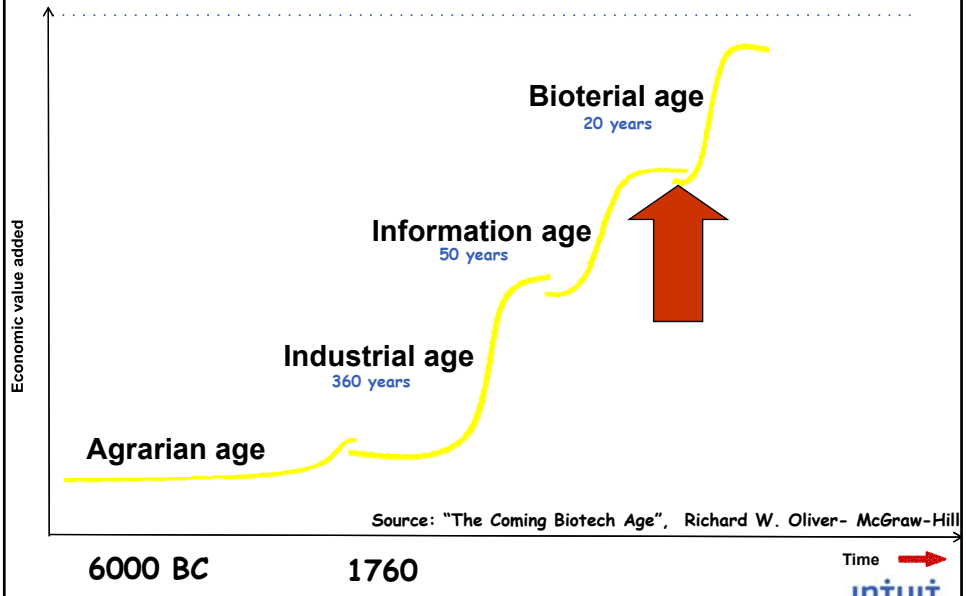
intuit

Overview

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- **Trends**
- Software Product Line Architecture
- Architecture for the Ecosystem
- Dynamic Software Architectures
- Conclusion

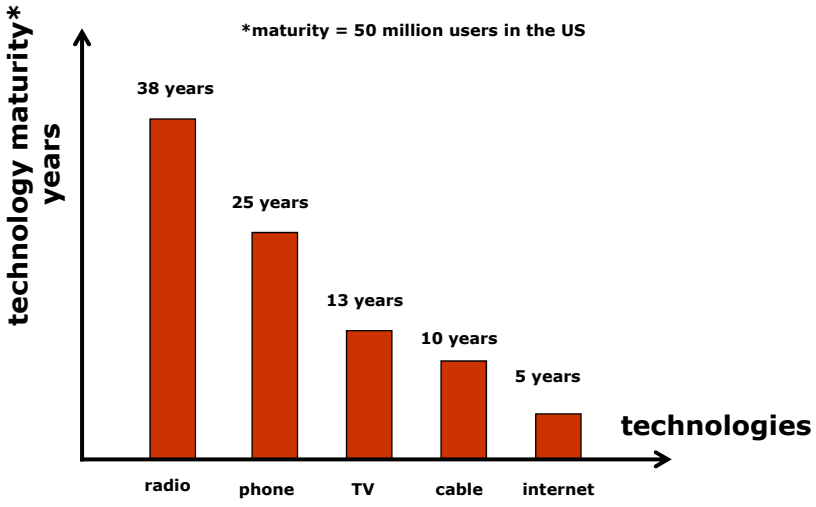
intuit

Where are we going? How fast?



intuit

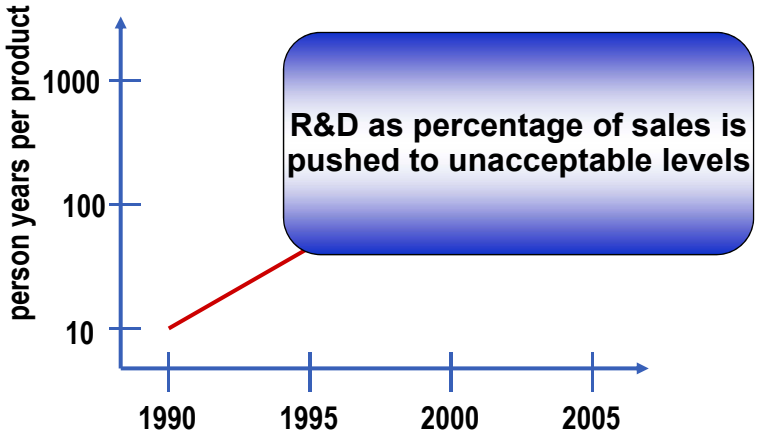
Innovation Adoption



intuit

Trend: software size

- software needs in products constantly increasing



intuit

Trend: systems of systems

systems increasingly need to be integrated with other systems

- Information systems
  - from manual data exchange to behavioural integration
- embedded/technical systems
  - from stand-alone to signal exchange to behavioural integration

Unilateral control of system functionality is diminishing

intuit

Trend: Variability

Variability needs in software are constantly increasing because

- variability moves from mechanics and hardware to software
- design decisions are delayed as long as economically feasible

# of variation points is increasing
time of binding is constantly delayed
party performing the binding is changing

intuit

Later Binding

- trend is towards later binding and increased automation

intuit

Software Platforms – The Rules are Changing!

- Scope of platform te
  - S40 – 100% of pro
  - S60 – 60% of produ
  - MAEMO – 30% of p
- Requirements mana
  - S40 – complete con
  - S60 – control, but shared with Symbian, licensees and 3rd party developers
  - MAEMO – only UI is controlled (but not completely), rest is influence-based
- Architectural control
  - S40 – complete control
  - S60 – major controller, but significant influences elsewhere
  - MAEMO – largely through influencing and collaborating with Open-Source community

From "The Cathedral" to "The Bazaar"

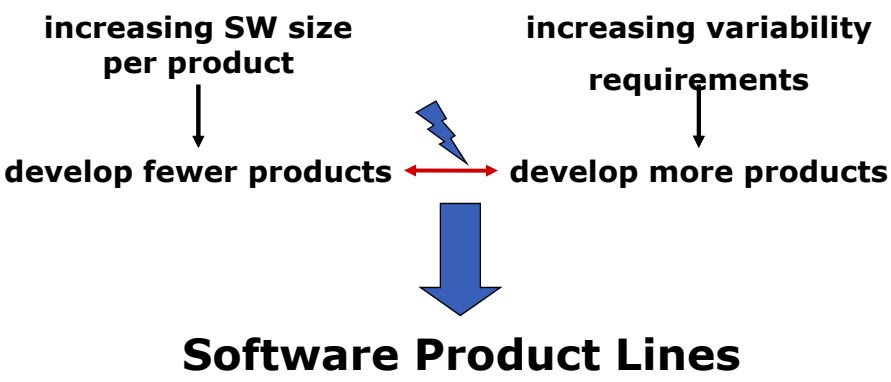
intuit

Overview

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- Trends
- Software Product Line Architecture
- Architecture for the Ecosystem
- Dynamic Software Architectures
- Conclusion

intuit

Software Challenge



intuit

Defining Software Product lines

- Product consists of:
  - Product specific components
  - Shared components configured for the product
  - Shared components used as is
  - Externally developed components
- Software Product Line consists of:
  - Software product line architecture
  - Shared components
  - Commonality and variability model

intuit

A Brief History of Software Reuse

Technology orientation
- Module
- Object & class

Technology and process
- Component
- Object-oriented frameworks

Complete intra-organizational perspective
- Software product lines

Inter-organizational (or eco-system) perspective
- Open software platforms
- Open-source software communities

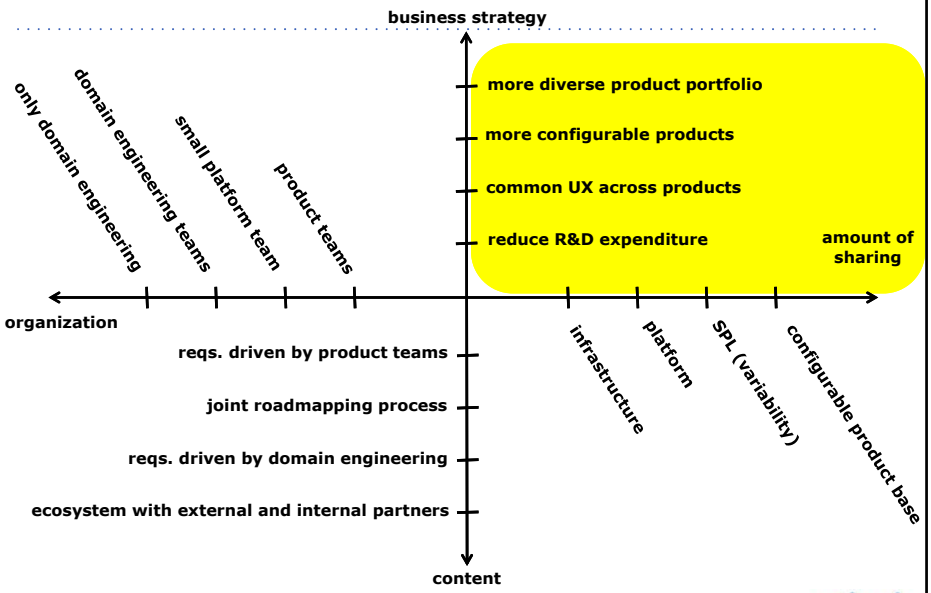
intuit

What Success Looks Like

- Business perspective: SPL technology forms the basis for "the next S-curve of growth" for the company
- R&D perspective: Order of magnitude richer product portfolio against stable or slightly increased R&D investment
- Ways to achieve success:
  - Product portfolio diversity
  - Common user experience for products in the portfolio
  - Much more customizable customer products
  - Higher quality products due to reliable shared core
- Often ignored advantage
  - Low opportunity cost of new product experiments
  - Improvements become available for all products at once
  - Improved productivity due to specialization of teams

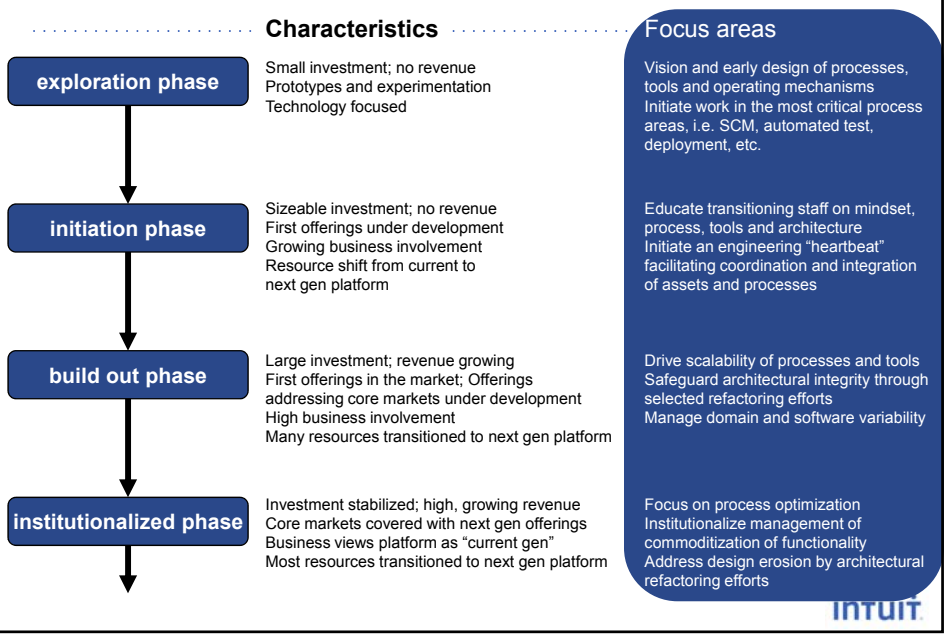
intuit

Dimensions of SPLs



intuit

Generic Next Gen Adoption Phases



intuit

Risks inherent to Reuse Strategies

- **Complexity** – the "gravity" of software engineering: Reuse can add complexity by creating dependencies between previously autonomous organizational units.
- **Web of dependencies**: Can lead to a "lockstep" evolution model in which everyone has to evolve synchronously.
- **Coordination cost**: dependencies require significant synchronization and alignment, diminishing the benefits of strategic reuse.
- **Offering integration cost**: often the cost of offering integration is higher than expected due to the complexity of configuring and integrating the selected shared assets.
- **Process & tool divergence**: teams with diverging "external" interfaces, e.g. different release cycles and mechanisms, "creative" interface management, immature requirements management, lacking quality management, etc. cause significantly higher offering creation cost and jeopardize the product line effort.

Strategic reuse creates competitive advantage as long as we manage to these risks

intuit

Mitigation Strategies

- **Decoupling**: replace coordination mechanisms with a uniform set of architecture and team responsibilities
- **Independent deployment**: maximize the ability of components to evolve independently through upward and downward interface compatibility requirements
- **Knowledge management**: design for integration, interface management, documentation and production plans help consumers be productive against minimal investment
- "**Step function**" change: as we are at an "inflection point", radical change can be imposed, i.e. harmonizing processes and tools
- **Specialization**: Separate development from process and tools evolution

Proven techniques must be proactively applied to ensure success

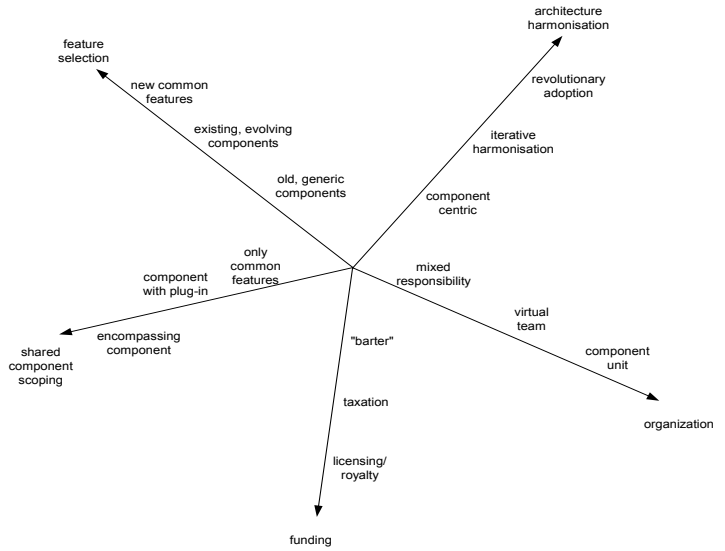
intuit

Issues While Transitioning

- key issue:
  - Adopting SPLs requires **changes to all aspects of the business** – this is often ignored
- potential issues:
  - mismatch between shared components and product needs
  - design erosion of shared components
  - complex interface
  - high degree of "organizational noise"
  - inefficient knowledge management
  - evolution causes ripple effects through the R&D organization

intuit

Decision Dimensions during Adoption



intuit

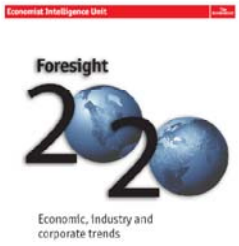
Overview

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- Trends
- Software Product Line Architecture
- Architecture for the Ecosystem
- Dynamic Software Architectures
- Conclusion

intuit

Building Web 3.0 at Intuit?

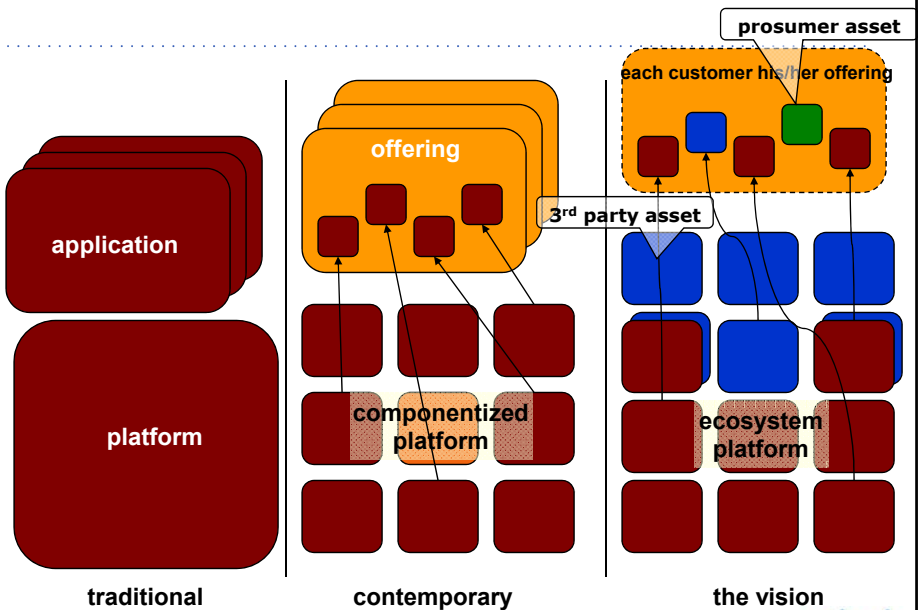
3 Atomisation. Globalisation and networking technologies will enable firms to use the world as their supply base for talent and materials. Processes, firms, customers and supply chains will fragment as companies expand overseas, as work flows to where it is best done and as information digitises. As a result, effective collaboration will become more important. The boundaries between different functions, organisations and even industries will blur. Data formats and technologies will standardise.



"My prediction would be that Web 3.0 will ultimately been seen as applications which are pieced together. There are a number of characteristics: the applications are relatively small, the data is in the cloud, the applications can run on any device, PC or mobile phone, the applications are very fast and they're very customizable. Furthermore, the applications are distributed virally: literally by social networks, by email. You won't go to the store and purchase them... That's a very different application model than we've ever seen in computing." —Eric Schmidt

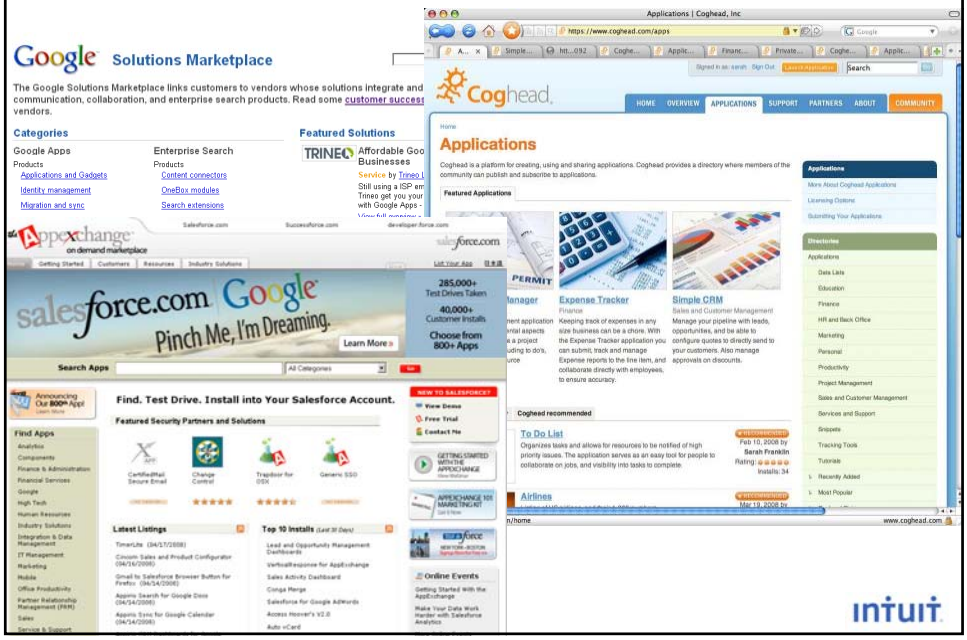
intuit

From Pre-Packaged Offerings to Customer-Assembled



intuit

Platform-as-a-Service



Overview

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- Trends
- Software Product Line Architecture
- Architecture for the Ecosystem
- **Dynamic Software Architectures**
- Conclusion

intuit

Assumptions about Architecture

- software architecture is hard to change
- consequently, design architectures carefully
  - architecture assessment
  - architecture design
- software architecture is static, the stable part of the system

inflexible architecture is good/fact of life!

intuit

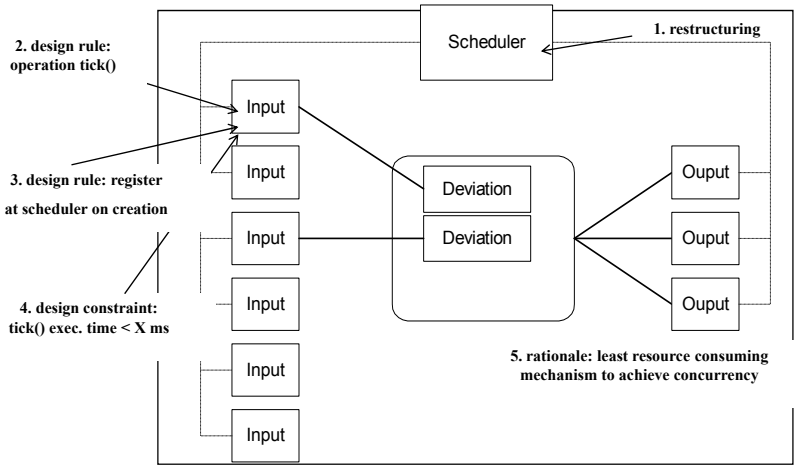
Flexible Architectures?

- why is SW architecture hard to change?
- ignored aspect of the problem
- loss of design knowledge – vaporizes during
  - architecture design
  - component development
  - system evolution

⇒ architecture design decisions are lost

intuit

Example – Fire Alarm System



intuit

Architecture Design Decisions

architecture design decisions consists of

- restructuring effect
- design rules
- design constraints
- rationale - new principles, guidelines, etc.

and are taken in response to

- functional requirements
- quality requirements

intuit

Dynamic Software Architectures

- Later binding trend also applies to software architecture
- Design decisions are taken in response to functional and quality requirements
- For dynamic systems, the desired behaviour of the system is required to change post-deployment
- This requires design decisions to be reversed and/or replaced
- First class representation of design decisions, at run-time, would facilitate this behaviour

Alternative perspectives

- Variability requirements in software are constantly increasing
- Achieving superior user experience increasingly requires user specific intelligent behaviour
- Context changes in mobile systems

intuit

Conclusion

- Vem är jag? Wie ben ik? Who am I?
- Introducing Intuit
- Trends
- Software Product Line Architecture
- Architecture for the Ecosystem
- Dynamic Software Architectures

Software architecture is more important than ever

The notion of architecture needs to evolve with the needs of the software industry

intuit