CREDIT SUISSE

Muddy Rain: Seeding Clouds with the “Big Ball of Mud”

Eric Newcomer
Chief Architect, Investment Banking Division

Credit Suisse Group today – key facts

- Global bank headquartered in Zurich, serving clients in private banking, investment banking and asset management. Among top 5 or 10 global banks, depending on metric (revenue, assets)
- Registered shares of Credit Suisse Group AG (CSGN) are listed in Switzerland (SIX) and as American Depositary Shares (CS) in New York (NYSE).
- Total number of employees: 49,000.
- IT employees about third of the total – IT spend ~$4B annually

CREDIT SUISSE

Complexity: Our top IT challenge

Today‘s IT systems have the following characteristics:

- Very-large-scale: more than 6000 applications with more than 100M Lines of Code
- Interdependent: large number of tightly coupled, networked components
- Aging: parts of the system are becoming obsolete and must be replaced (obsolete technology, end-of-life applications)
- High rate of change: continuous flow of new business requirements which must be implemented (Several 1000 application changes per week)
- Demanding operational quality: systems must have high reliability, good availability, sufficient security etc.

Legacy IT Environments: "Unmanaged Evolution"

- Lack of enterprise architecture

  "*A **Big Ball of Mud** is a haphazardly structured, sprawling, sloppy, duct-tape-and-baling-wire, spaghetti-code jungle. These systems show unmistakable signs of unregulated growth, and repeated, expedient repair.* *

- This is no one's fault – it is a natural part of IT evolution
  - "Foote and Yoder do not universally condemn "big ball of mud" programming, pointing out that this pattern is most prevalent because it works — at least at the moment it is developed. However, programs of this pattern become maintenance nightmares."

* http://en.wikipedia.org/wiki/Big_ball_of_mud The term was popularized in Brian Foote and Joseph Yoder's 1999 paper of the same name, from which this definition is taken.

“Shanty Town” (from Big Ball of Mud)

- It’s not really fair to call our IT a shanty town
  - Many of our applications are well designed
  - For the most part, the applications do their job
- But the analogy holds
  - Lack of common infrastructure
  - Lack of common design standards
  - Hard to fix, maintain, and improve

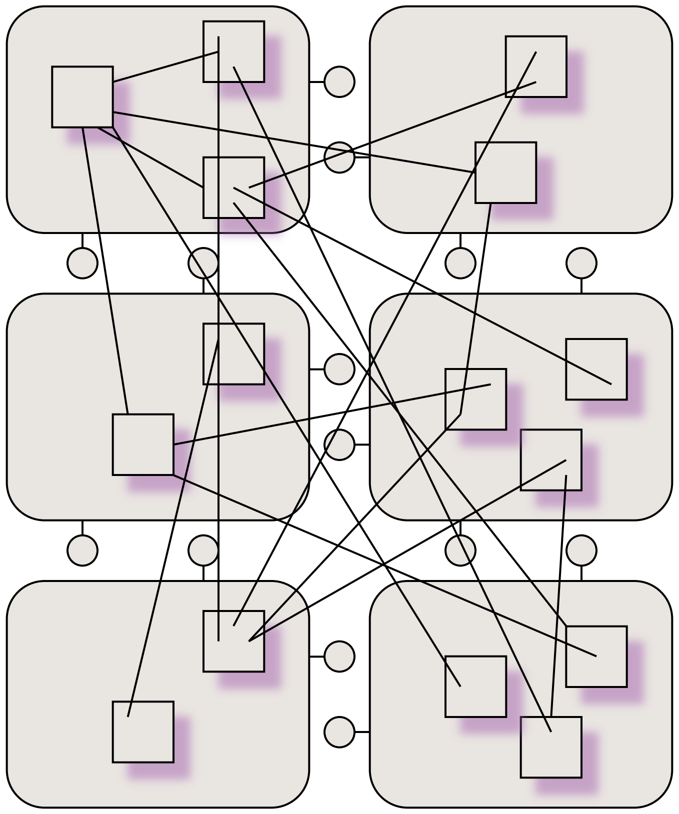
CREDIT SUISSE

Managing Evolution via SOA-Designed Components: reduce complexity and increase effeciency

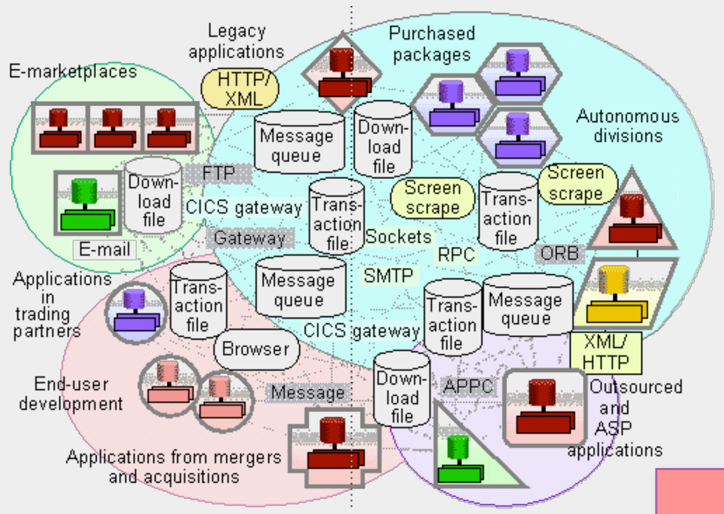
1. Monolithic landscape with uncontrolled dependencies

2. Grouping data and functions into components

3. Decoupling components through services



Credit Suisse



Almost all of the business applications of the enterprise were not written using consistent architecture. Instead they are byproducts of the evolution of IT:

-Mainframe transactions

-C++ Client/Server Apps

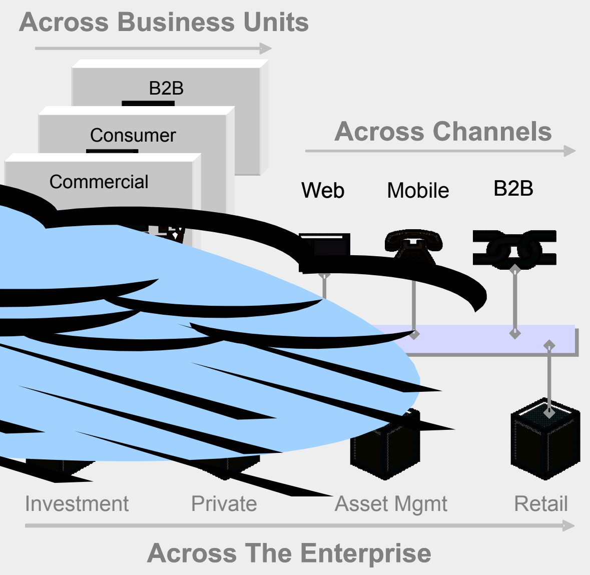
-Middleware Islands

-Home Grown / Dark Matter

-Java / .NET mixture

Solution: Expose and modularize existing enterprise systems as software services

- Plug-i
  and pro
- Encapsulate the
- Deploy the services on today’s mod
  platforms



Building our own private cloud

- Because we' re a bank, and a Swiss bank at that
  - Huge security concerns, especially Swiss client data
- What cloud model are we following:
  - Amazon EC2
  - No cloud models with specific APIs or language limitations
- Building on top of virtualization
  - Our initial offering is customized VMware, basically
  - Next offering (already available) is "EC2" like
- Some non-confidential services may go to EC2
  - Some prototypes have been done
  - Public data applications may go
- As security get better, more apps will go
  - We would like the capabilities to span internal/external boundaries
- Use RESTful and/or WS* implementations of SOA?