CREDIT SUISSE

Public

15 Years of Service Oriented Architecture at Credit Suisse

Lessons Learned – Remaining Challenges

Keynote SATURN 2013 – Minneapolis, USA

Stephan Murer, Senior Advisor, KG 97

May 1, 2013

Information Technology at Credit Suisse

What we do

- **Information Technology** is committed to fueling Credit Suisse's growth. Our objective in IT is to partner closely with the business to deliver innovative and cost-efficient results. In today's competitive environment, IT drives performance and revenue growth. By directly aligning our IT initiatives with the bank's overall business objectives, Information Technology helps provide Credit Suisse with a distinct competitive advantage.

Who we are

- **Investment Banking and Client Channels Technology IT:** Partners with the Investment Banking Division to deliver integrated and innovative technology solutions
- **Private Banking & Wealth Management and Shared Services IT:** Partners with Private Banking & Wealth Management and all Shared Services divisions to provide exceptional technology to our clients and employees
- **Technology Infrastructure Services:** Provides our business partners global, standardized information technology infrastructure and systems

Information Technology

Facts and Figures[1]

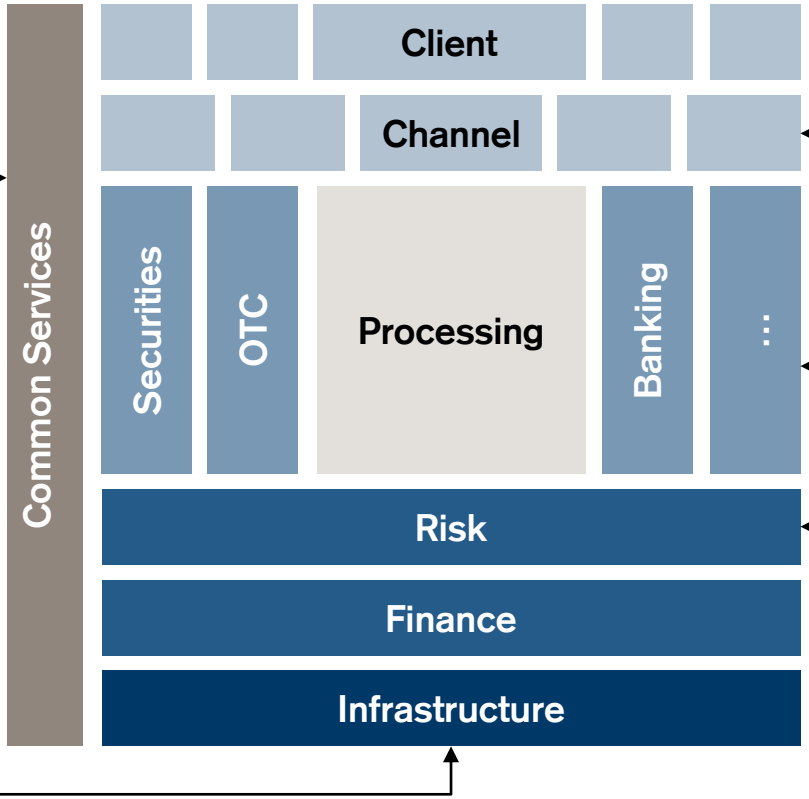
- 66,400 supported users in 550 locations
- 4 main hub Production – Disaster Recovery pairs of data centers consuming 14.1 MW[2] of power
- Hardware
  - 85,500 workstations/laptops
  - 21,710 physical servers with more than 43 petabytes of storage
  - 4 host/mainframe CPUs with 83,200 MIPS[3] provided
- Software
  - ~6,021 applications
- Email
  - 67,541 email accounts
  - 5.5 m emails/day
- Helpdesk
  - 416,760 tickets (2012)
- Print Center
  - 198 million pages (2012)

[1] As of March 2013
[2] MW = Mega Watt
[3] MIPS = Million Instructions per Second

Convergence to More Agile and Standardized Operating Model to Improve Our Capabilities

Shared components
- Single sources of reference data
- Common Service Oriented Architecture (SOA)
- Common domain and business services model

Common infrastructure
- Common infrastructure based on consistent architectural standards that support our systems in a scalable, agile, and cost-efficient manner



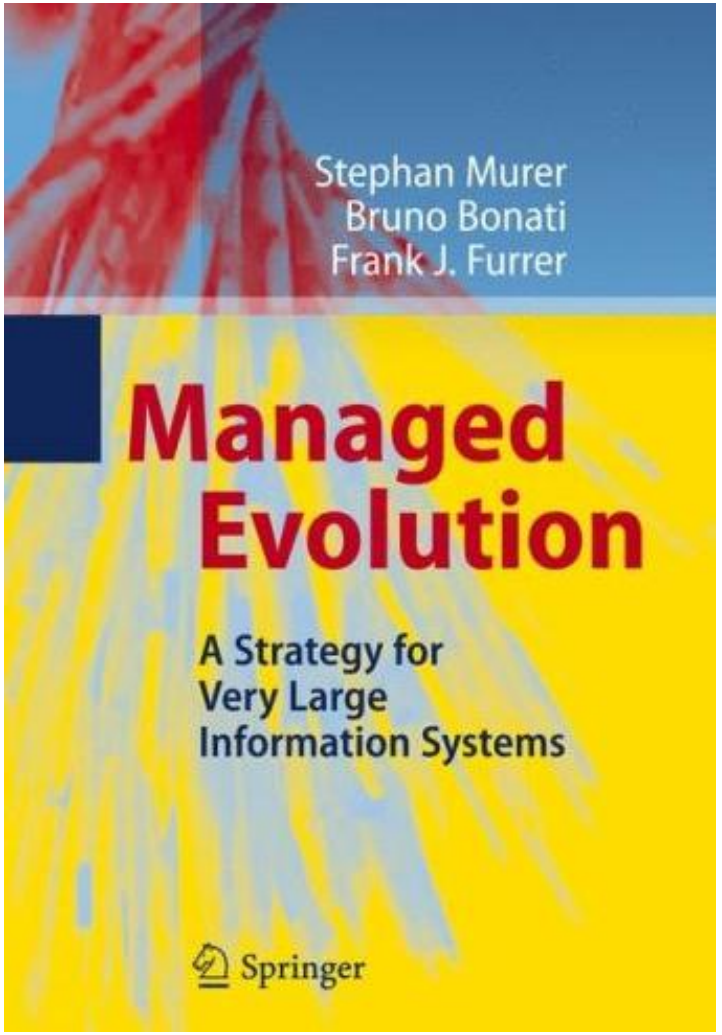
Consistent client experience
- Single sign-on
- One portal, multiple solutions

Product processing
- Shared, legal entity-agnostic global processing platforms

Risk management and financial control
- Unified risk and finance platforms, based on agreed front-office data

Managed Evolution of Very Large Systems Requires Stable Interfaces

- **Hard to replace a very large information system as a whole**
  - High cost (> CHF 1 bn, estimate for Swiss platform)
  - Development time too high (> 5 years)
  - High risk, as both technical and business prerequisites shift over time
- **Managed evolution is the only feasible approach**
  - Stepwise transformation of landscape, renewing component after component
  - Multi-year effort which gradually implements the target architecture
- **Well encapsulated components exposing managed interfaces are a prerequisite for managed evolution**
  - Technically renew components, without affecting clients
  - Consolidate redundant data and functionality behind common interfaces and simplify
  - Seamlessly operate a technically heterogeneous system resulting from the evolutionary approach



Service Oriented Architecture (SOA)

Principles and Benefits

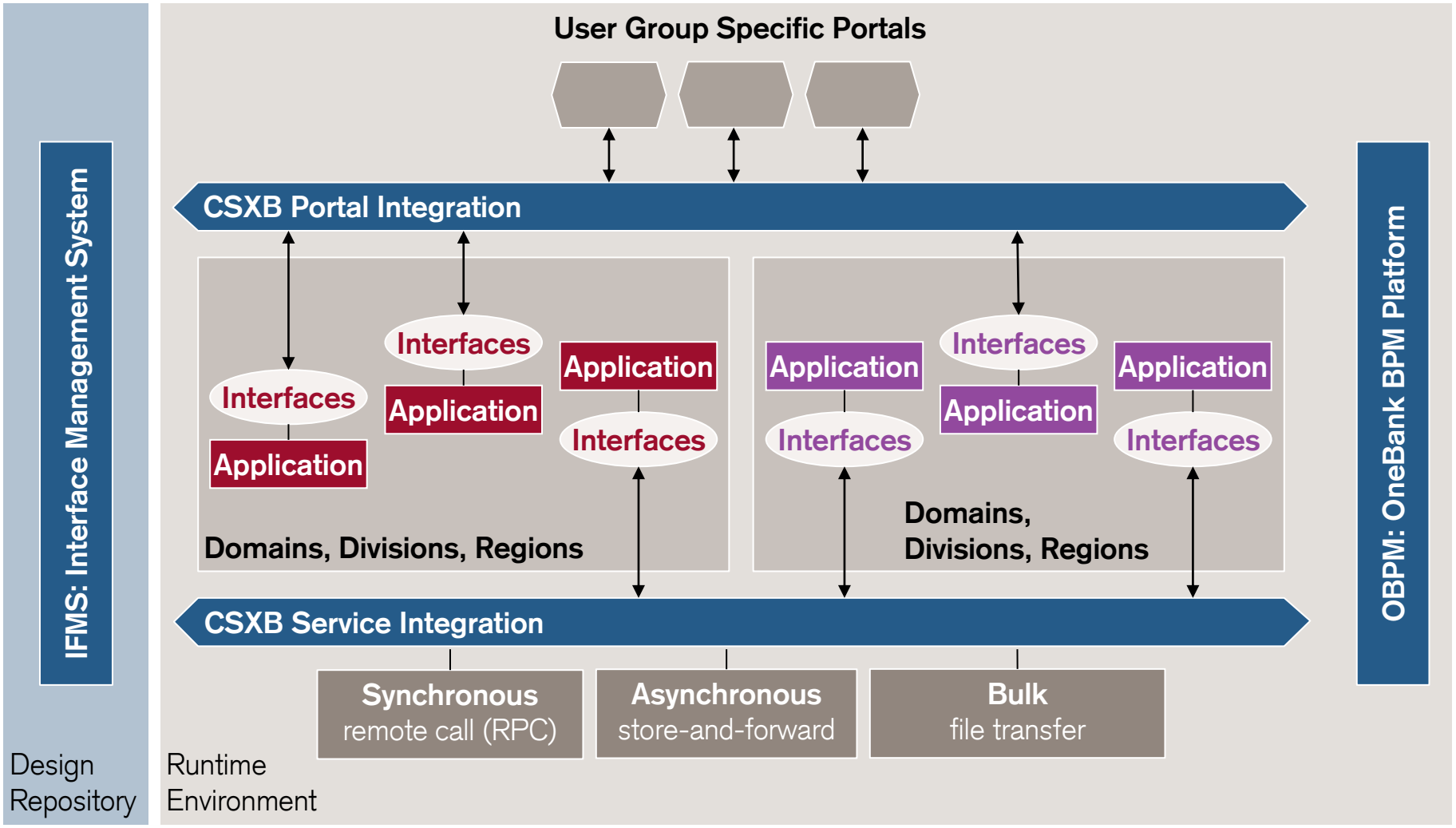
Principles

- Service interfaces and contracts
- Loose coupling
- Service abstraction
- Reusability
- Discoverability
- Interoperability

Benefits

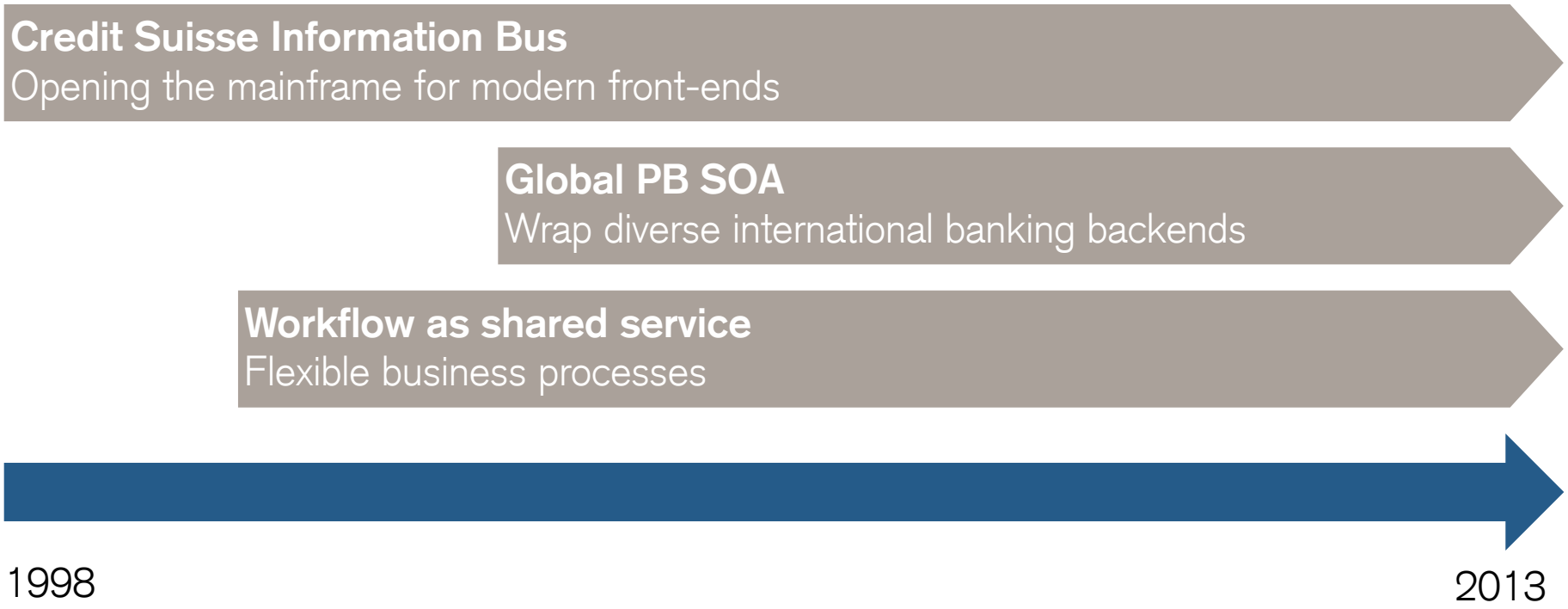
- Reduced integration expense
- Increased asset reuse
- Increased business agility
- Reduction of business risk

Credit Suisse eXchange Bus (CSXB)



SOA Experience @ Credit Suisse

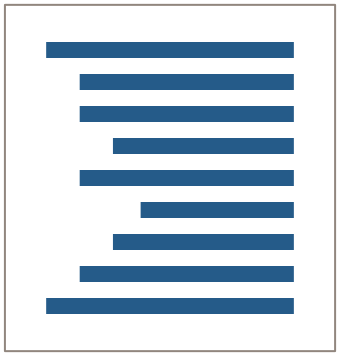
Three Case Studies

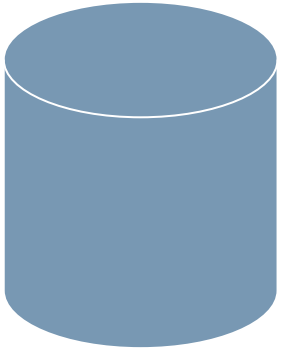


Case 1: Credit Suisse Information Bus (CSIB)

Facts	■ Started in 1998. More than 1,200 services built up to now. ■ All applications on the Swiss Platform offer and/or consume services today
Objectives	■ Enable Managed Evolution of Mainframe Platform ■ Component architecture for the Swiss Platform ■ Reuse of core data&functionality (mainly) residing on the mainframe for modern front-ends
Technology	■ Orbix – CORBA for synchronous services, migrating to WebServices ■ WebSphere MessageBroker/MQ for messaging ■ Ftps for files (“Bulk Services”), “File Broker” for flow control, transformation, etc.
Footprint	■ About 1,400 public services, 70 message publishers ■ 400 mn CORBA calls and 120 mn messages delivered per month

Different Life Cycles in Different Application Layers





GUI, presentation,
channel specific

2+ years

Application logic,
product specific

10+ years

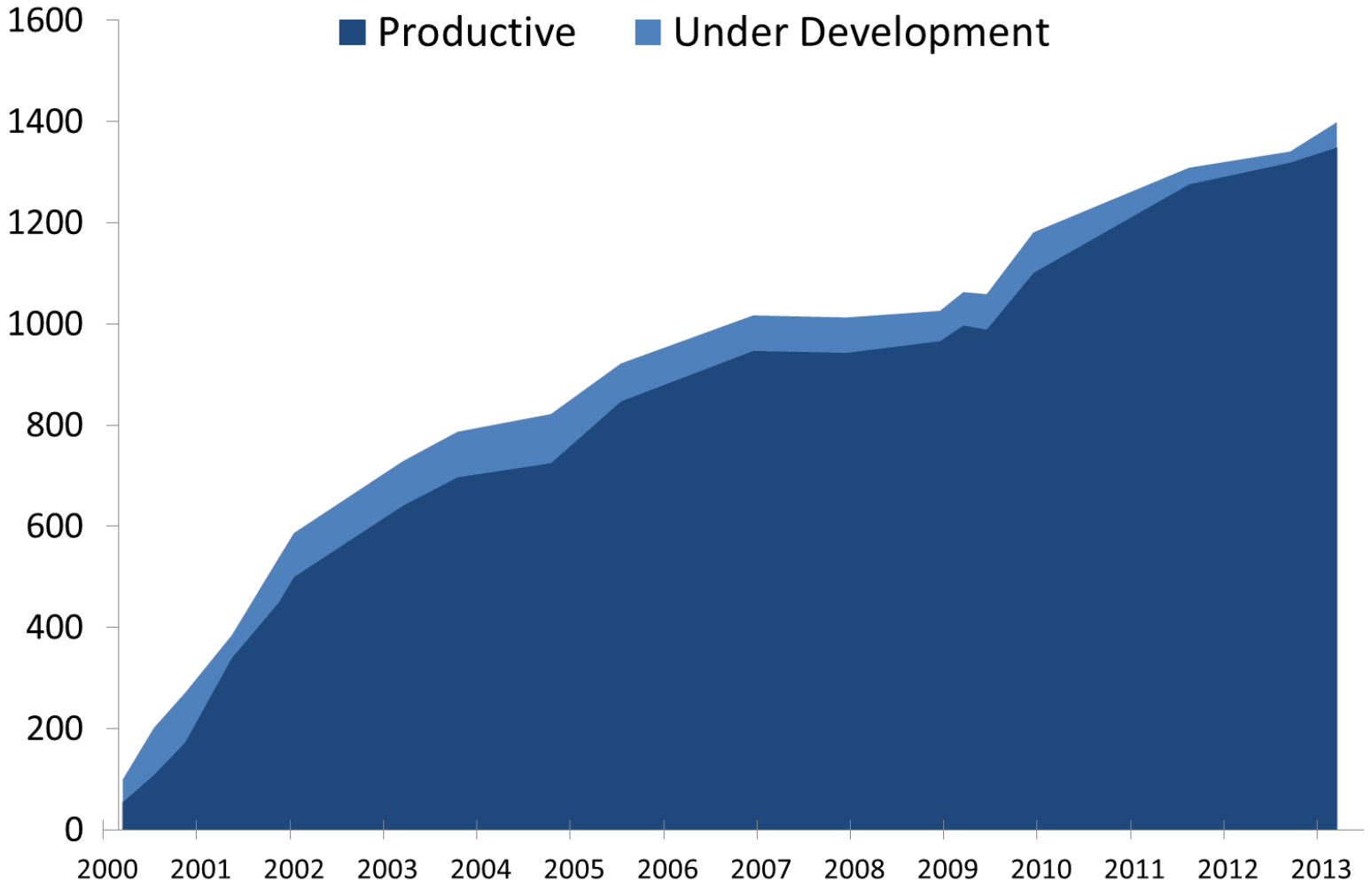
Core system,
close to operational data,
generic for the bank

30+ years

Measure Progress

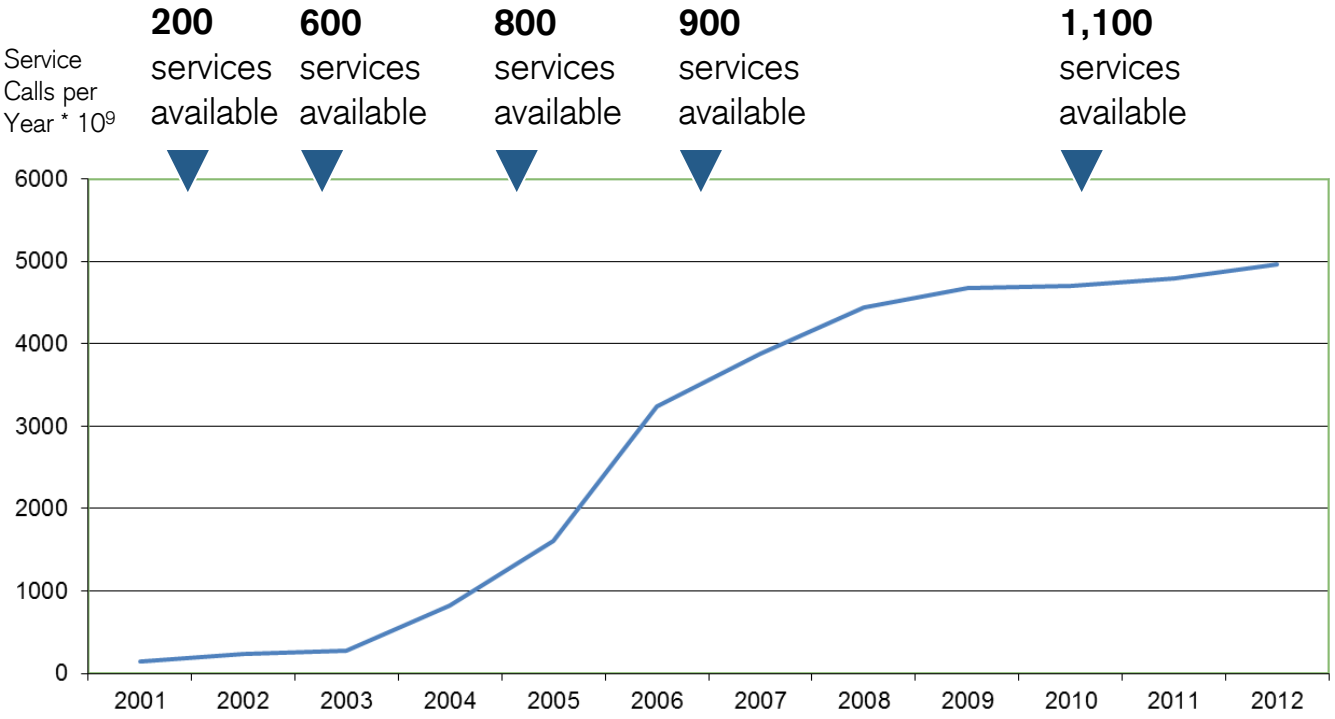
Services Built as Needed

- Build services as they are needed
- Gradually decouple domains
- Use existing functionalities



Measure Progress

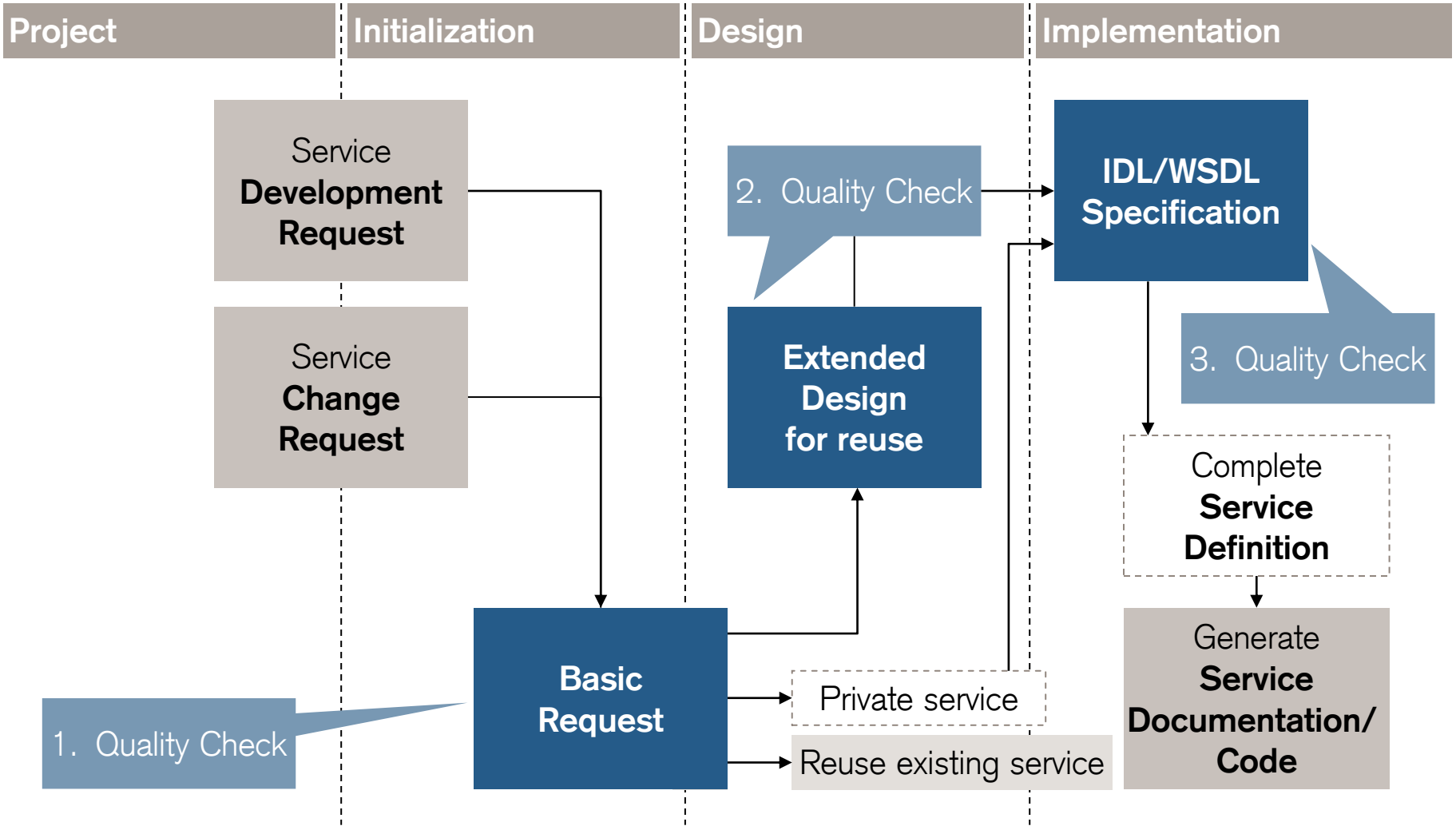
Use of Services Follows Availability



- Wide use of services follows a critical mass of available services
- Today ~5 billion service calls a year in Switzerland
- Fully decoupled platform some years ahead
- Core banking system on mainframe completely decoupled from remainder of platform

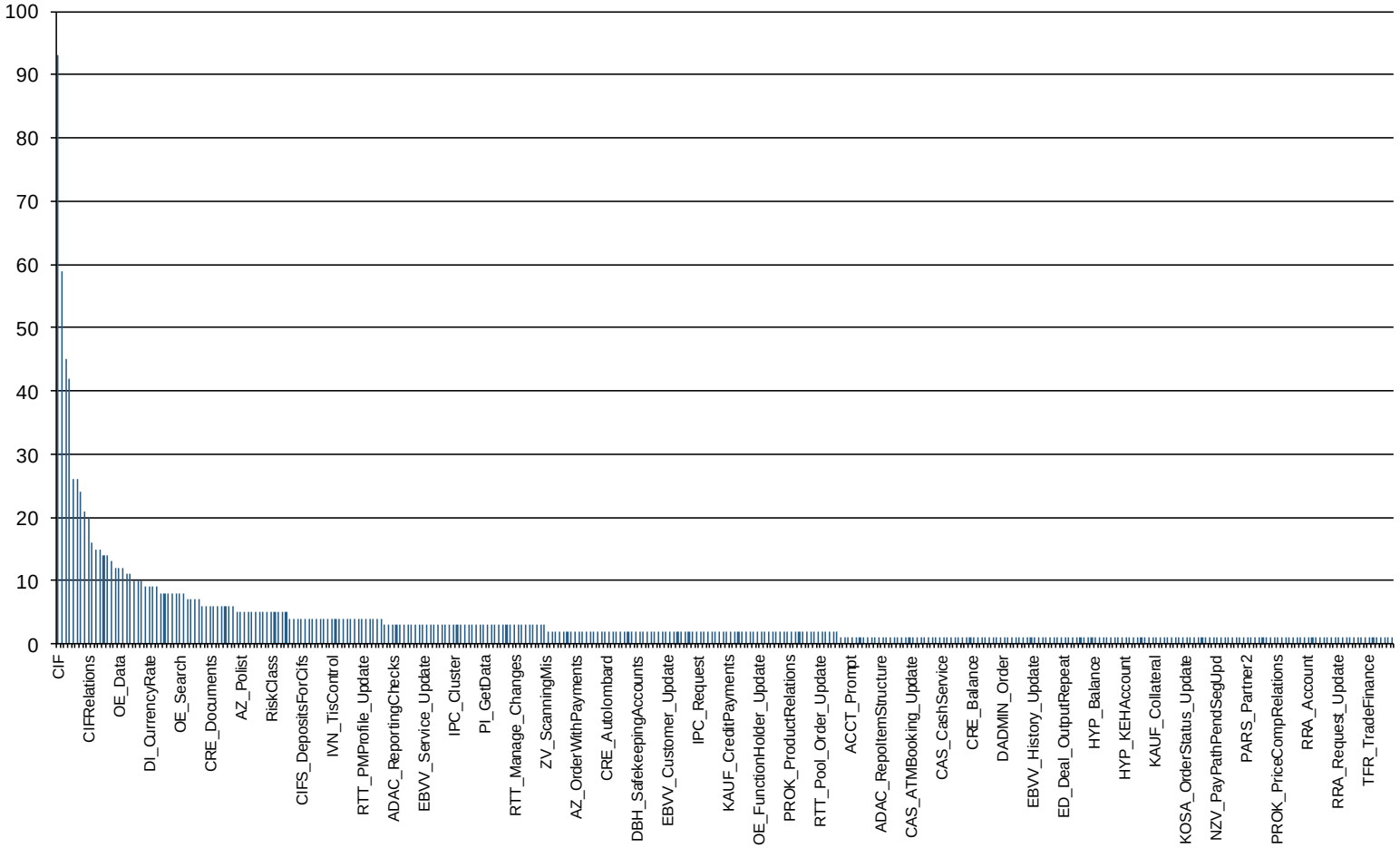
Governance

Bottom-up Requests, Top Down QA



Measure Re-use

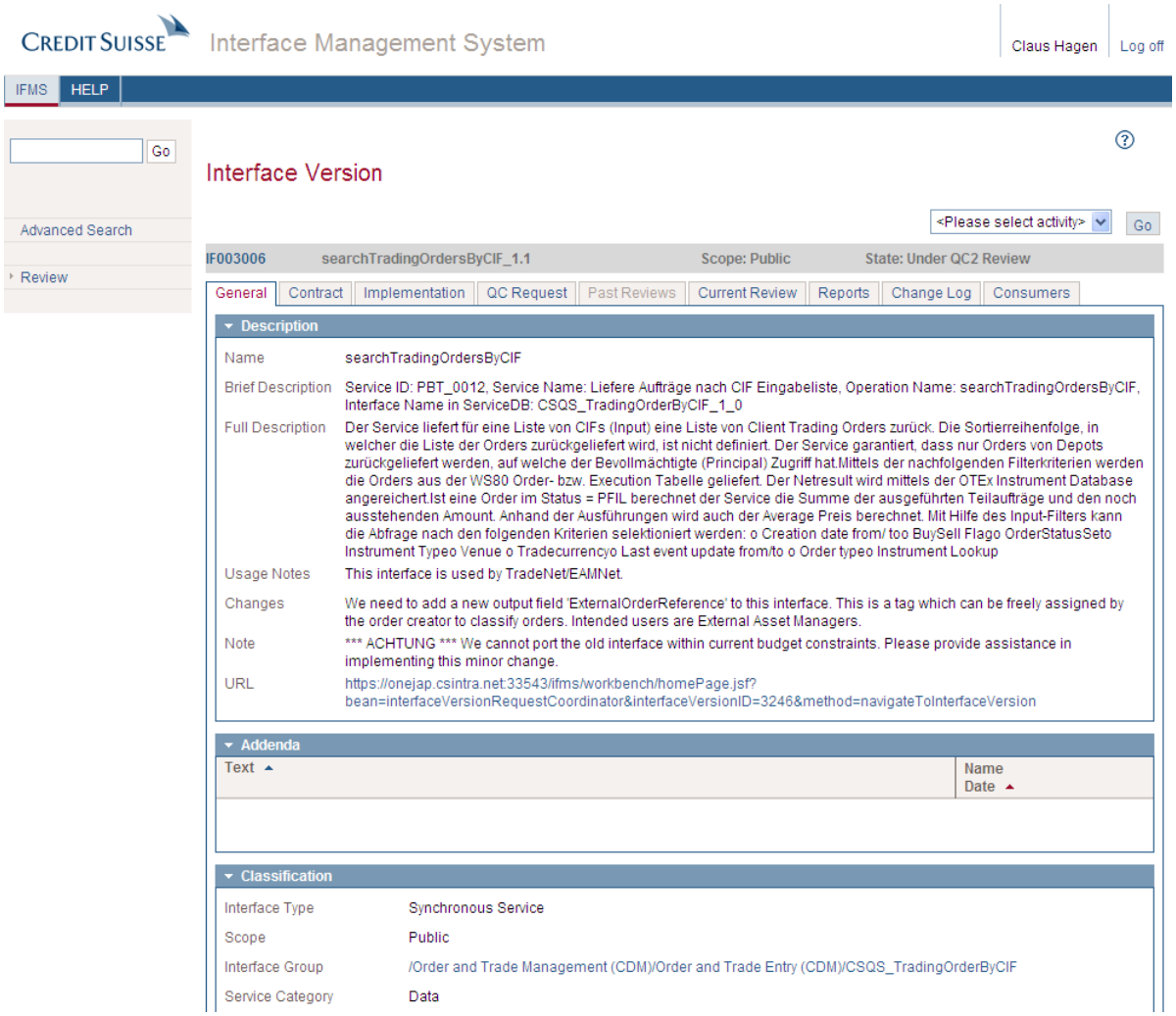
Re-use Is Very Uneven



- Re-use of services varies based on type of service. Reference Data Services have the highest re-use.
- About half of the services are reused
- Average re-use degree is 4–4 different client applications using a service

Governance

Interface Management System



- Searchable Service Catalog
- Data type repository with UM/ WSDL integration
- Governance workflows
- Lifecycle management
- Extensible code generator (MDA)
- Integration with other tools, like application repository or accounting

Case 1: Credit Suisse Information Bus

Experiences

Strengths

- Well-accepted, high proliferation, good re-use of services
- Solid middleware foundation
- Governance: Combination of bottom-up approach and top-down QA works

Challenges

- Management becomes difficult because of size (> 1,000 services)
- Business object model (BOM) required as a framework in which to manage the service landscape (currently in development)
- Long term migration towards Web Services, without compromising performance

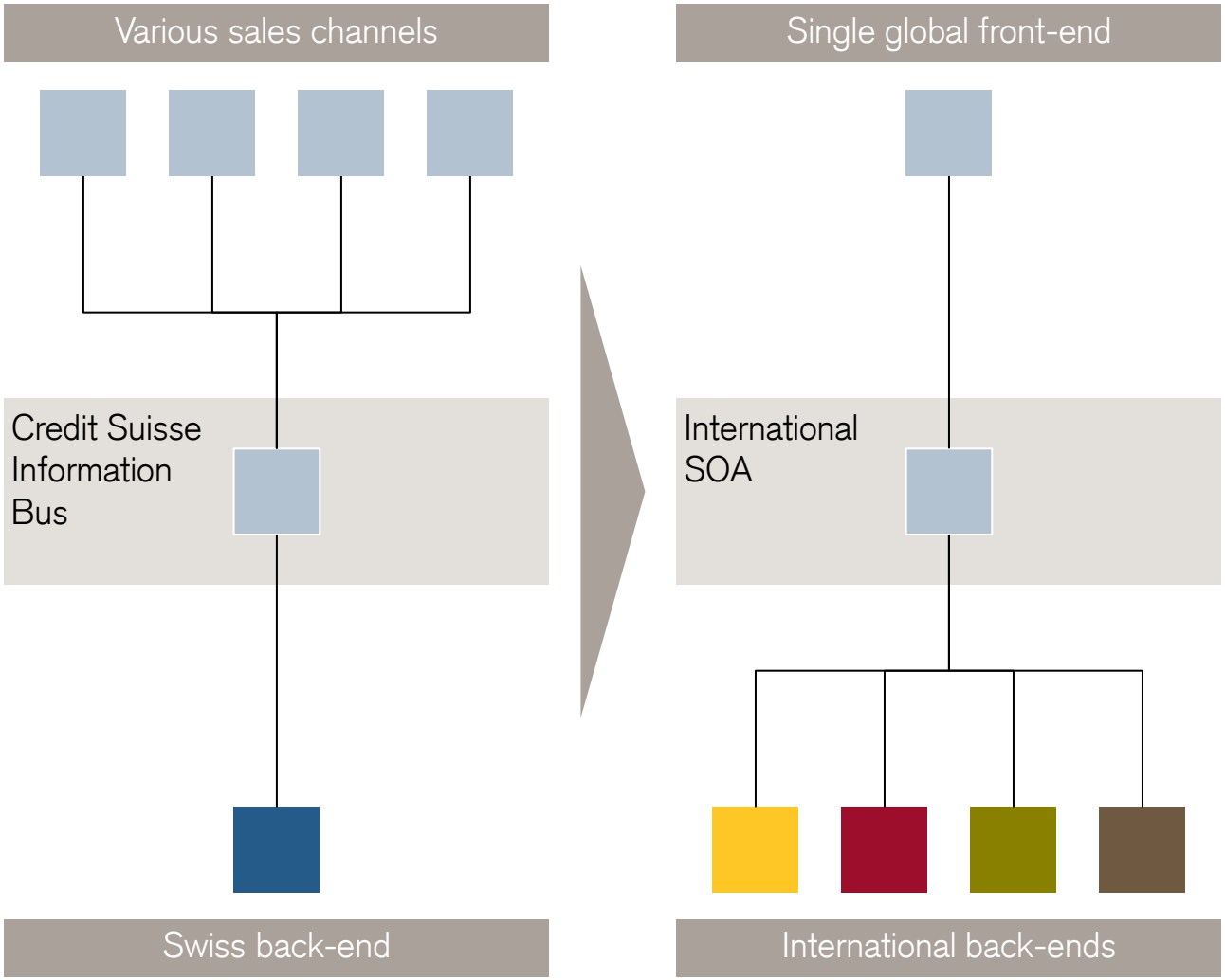
Lessons Learned

- Strong governance is essential
- Organization needs time to learn (years for large organization)
- Bottom-up approach helped in gaining critical mass, but…
- …supporting service design with common data model would greatly improve service quality and simplify QA
- Need improved interface management system (IFMS), market didn't deliver
- Formal service descriptions could be the basis for code generation
- Business rationale necessary – Architectural beauty not good enough!

Case 2: Global Private Banking SOA

Facts	■ Started in 2005 ■ About 200 services built up to now ■ Services implemented in different countries (Monaco, UK, Germany, Singapore, etc.) on top of different backends
Objectives	■ Re-use the same frontend applications with different local backends ■ Initial driver: standardized front desk application ■ Currently we see a wave of new frontend applications
Technology	■ Web Services (only synchronous communication needed)
Footprint	■ Small compared to CSIB ■ Growth due to implementation of global strategy

From Many Clients Using the Same Service to Many Providers Offering the Same Service



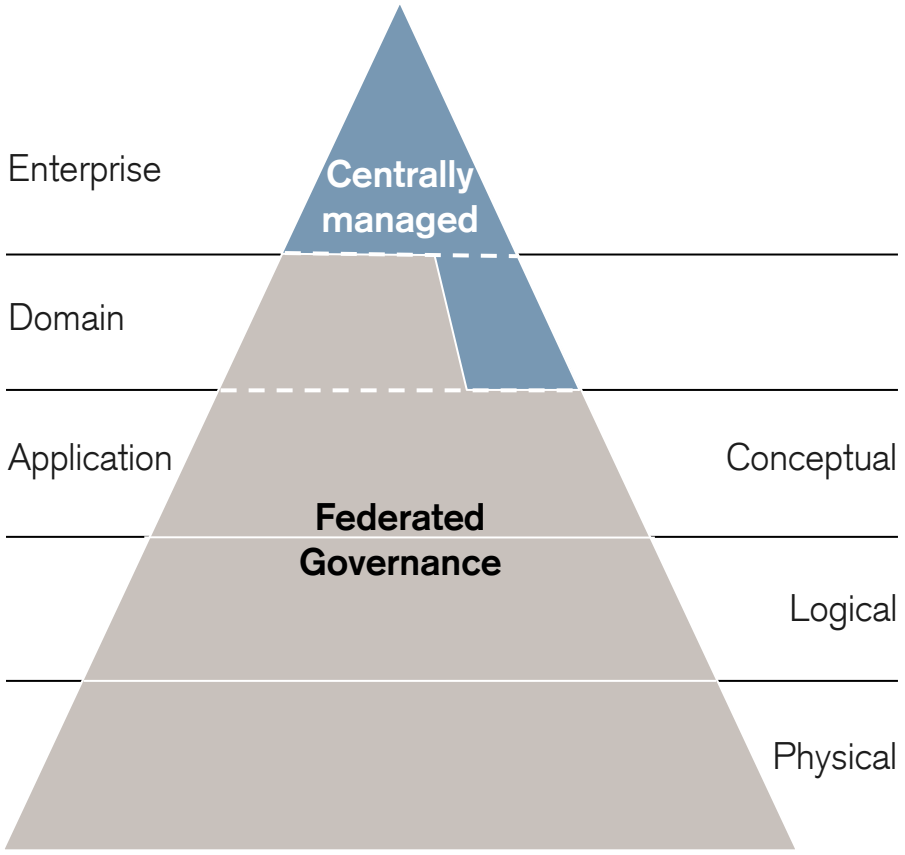
- Service provider defines semantics
- Implicitly defined with single service provider
- Challenging with multiple service providers

- Explicit semantics, especially explicit information model needed

Business Object Model

Common Federated Model Semantically Aligns Information

- **Precise semantics of information exchanged through services needed**
  - SOA means many producers communicating with many consumers
  - Bottom-up service design needs overarching information model to ensure semantic consistency
  - Top-down approaches to service design require information model defining the overall structure
- **Business Object Model delivers**
  - Provides reference naming for data elements
  - Typed data elements for correct representation
  - Relationships and annotations give semantic clarity
  - Used to define consistent service interfaces
  - Used to map service interfaces onto different implementations



Case 2: Global Private Banking SOA

Experiences

Strengths	■ Re-uses proven governance developed with the CSIB ■ Adds business object model to improve transparency and quality
Challenges	■ Dealing with the heterogeneity of different locations and their local backends ■ Dealing with a large diversity of teams and responsibility in the different locations ■ Long-term blur of distinction between Swiss and International platforms
Lessons Learned	■ The business object model really helps. Need to introduce it for all SOA environments. ■ We are still learning how to balance local agility needs and global coordination/ quality requirements

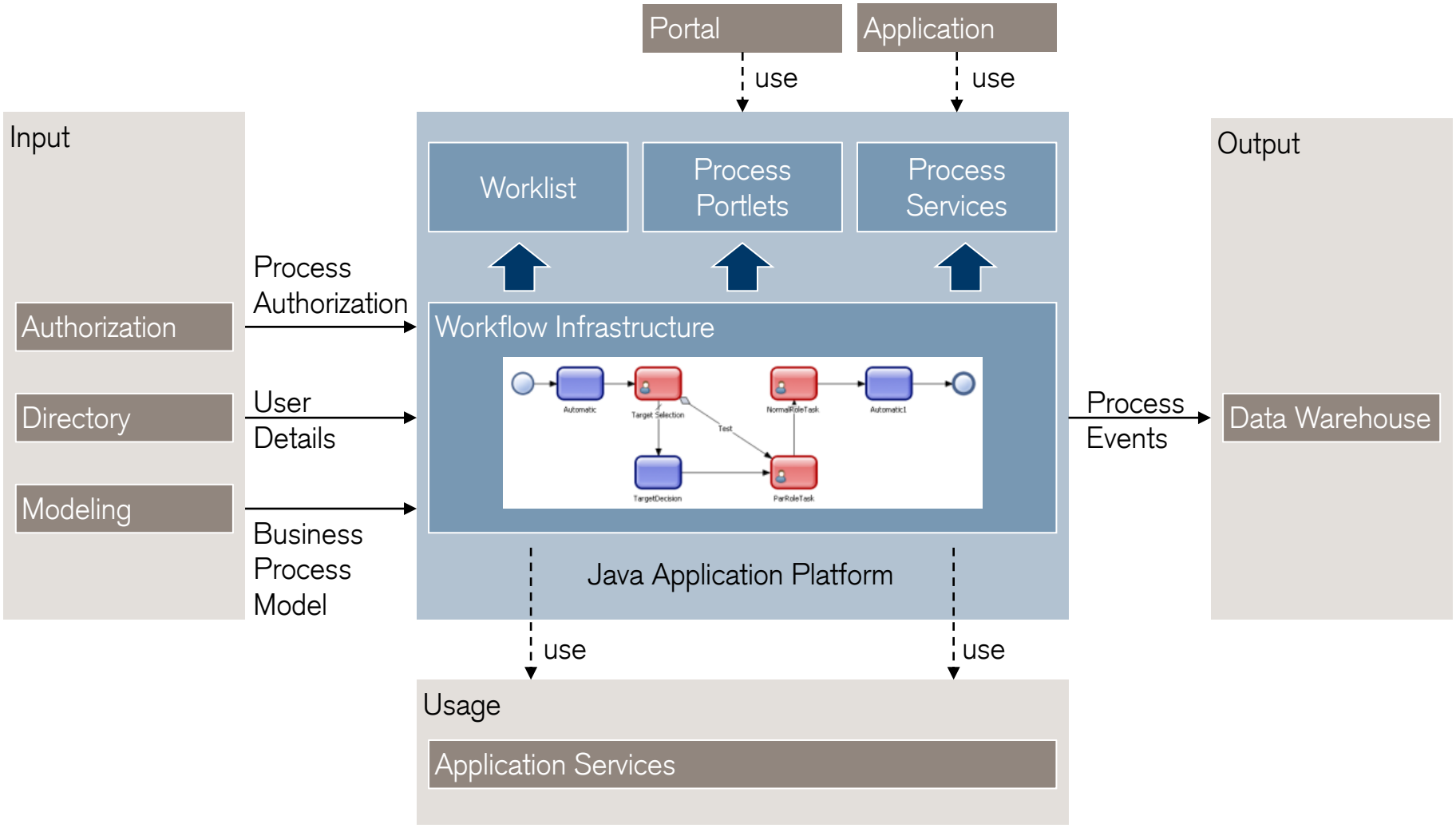
Case 3: Workflow Infrastructure

Process Integration

Facts	■ Started in 2000 ■ Currently broadly used internationally
Objectives	■ Decouple process management from application logic ■ Flexible service orchestration to support adaptation of application landscape to different business processes
Technology	■ Oracle BPM (current technology), IBM MQSeries Workflow (past technology), based on BPMN ■ Middleware migration successfully under way
Footprint	■ In use in more than 30 applications globally, with about 150 processes types. ■ About 100,000 process instances per month.

Workflow Infrastructure

Architecture Embedding



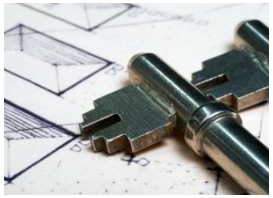
Employee Onboarding

Setting up IT Accounts

Challenge:
New hires could wait several days to begin work until they have access to all systems

Hire new employee

Create HR record



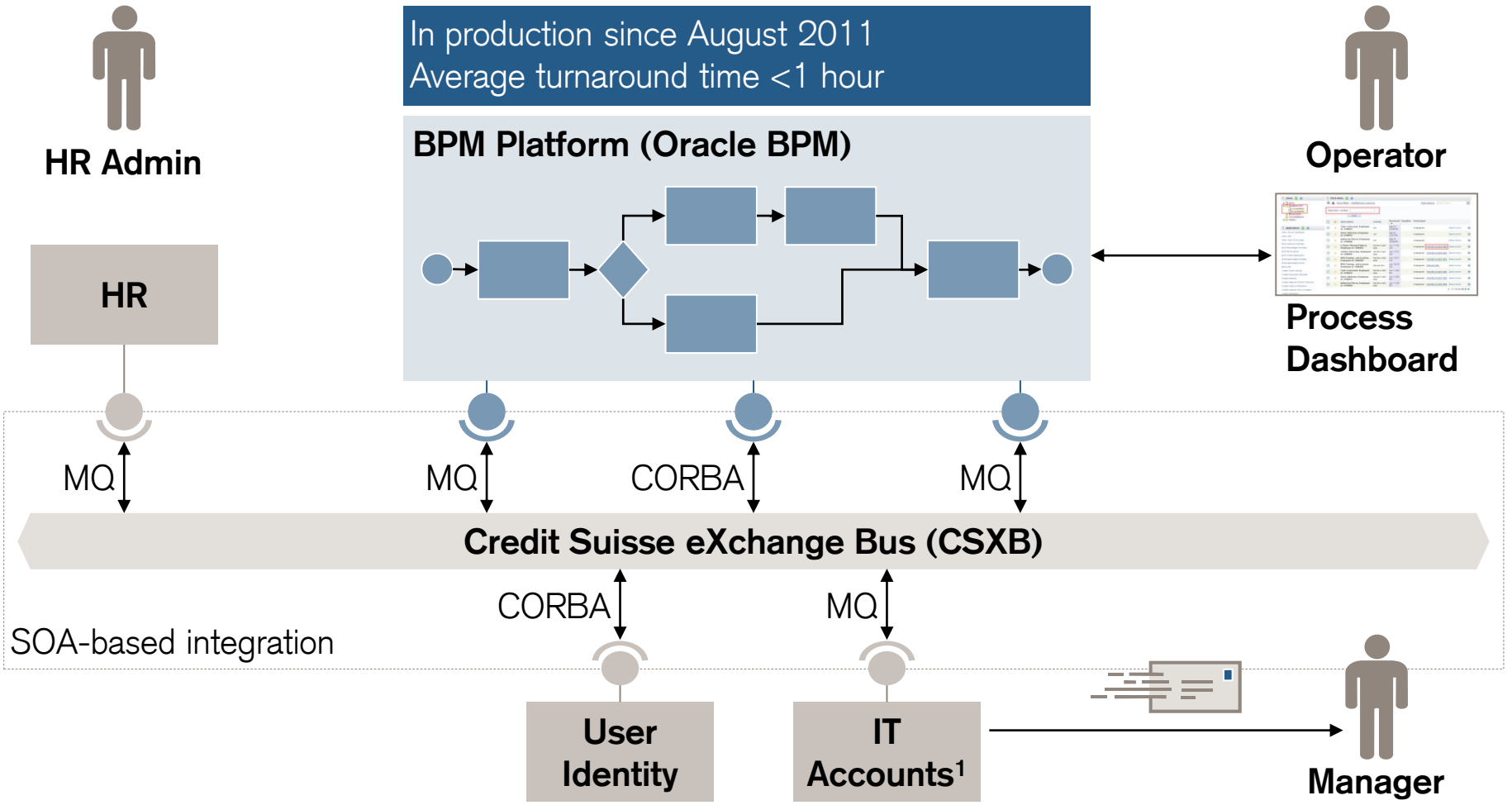
Create user identity

Set up IT accounts

Create email account

Issue smartcard

Service Orchestration through Managed Interfaces

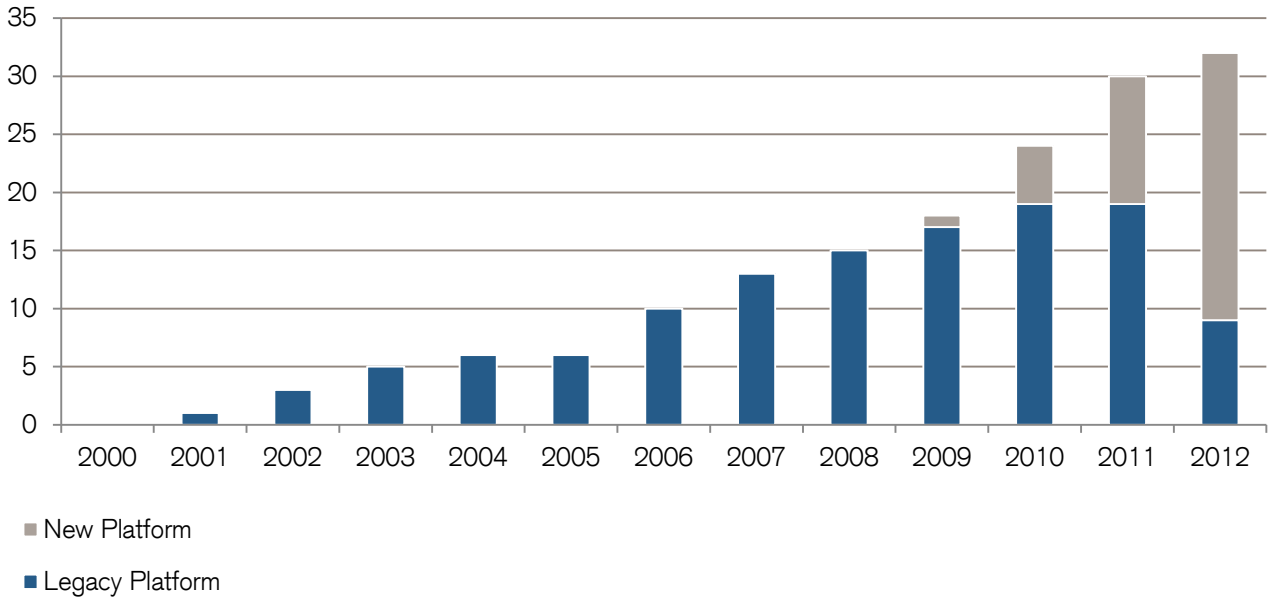


[1] IT account provisioning is handled by several different systems, which are not shown here in detail

Workflow as Shared Infrastructure

KPIs

Number of workflow applications



- Shared infrastructure used by multiple applications
- Central engineering, support and consulting team
- Infrastructure available in all major hubs
- Since 2010 transition from legacy solution (IBM MQSeries workflow) to new technology (Oracle BPM)
- Decouple workflow apps from server product as much as possible

Key numbers (November 2012)	Legacy platform	New platform	Total
Applications	9	23	**32**
Process models	118	28	**146**
Process activities	748	159	**907**
Process instances per month	109,000	48,00	**114,000**

Case 3: Workflow Infrastructure

Experiences

Strengths

- Widely accepted across the firm
- Clean isolation of workflow engine through robust APIs proved successful. Engine is being exchanged without too much impact on applications.

Challenges

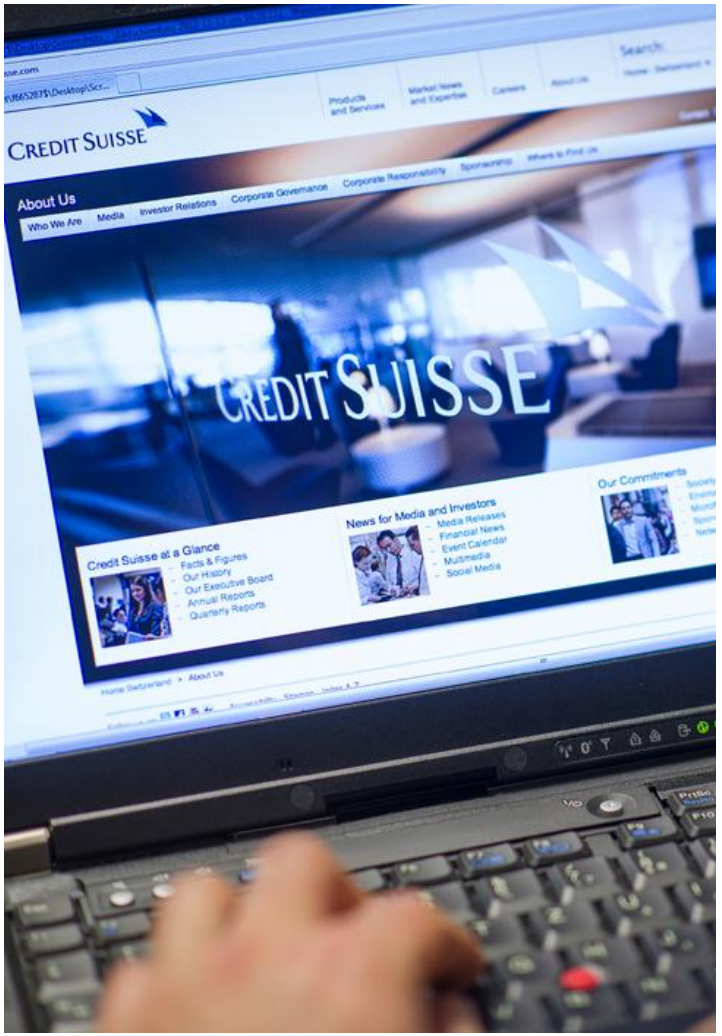
- Often off-the-shelf applications come with built-in workflow capability and don't lend themselves for external orchestration
- Competition from domain specific orchestration tools, leading to early successes, but failing in enterprise-robustness and end-to-end orchestration
- Business architecture needed to transform high level business processes into executable workflows

Lessons Learned

- Properly encapsulate your infrastructure
- Work closely with business
- Potential identified to use infrastructure in straight-through process
- Even "strategic" products don't always live long enough. In middleware, we need a "strategic" life of 15 years.

SOA is Everywhere in Credit Suisse

- **Front-to-back integration** → Data quality through well-defined interfaces
- **BIAN** → Market-wide service standardization
- **E-Commerce portal, Global Front Systems** → Desktop Integration
- **Financial Messaging Hubs** → Standardized message formats (SWIFT, ISO 20022, FPML, …) link internal and external applications
- **Reference data distribution** → Consistent reference data across the application landscape
- **Swiss platform renewal** → Replicated services for resilience and performance
- **Data Warehouse** → Managed bulk services feed data into warehouses

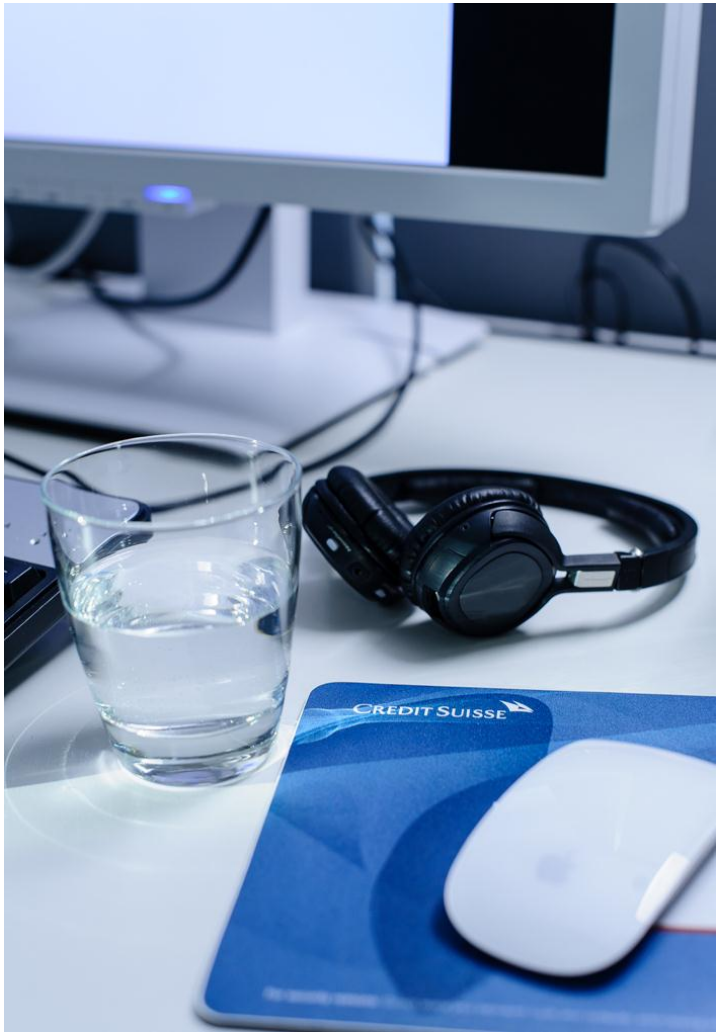


Where We Reach the Limits

Areas for Research (1/2)

- **Security remains a challenge**
  - Secure passing of the original initiator to all services involved (in the absence of sessions)
  - Second line of defense: Application level firewalls analyzing service traffic, distributed logging of service invocation, service monitoring for unexpected use
  - Fine-grained access control

- **Managing large service networks**
  - Version/release management with 1,000s of services and 100,000s of clients, semantic compatibility
  - System Management, service-level control, fault-tolerant designs, capacity management, service replication
  - Testing of complex component networks
  - Accounting of service use
  - Development governance, balancing reuse with demand-driven development
  - Cross platform interoperability



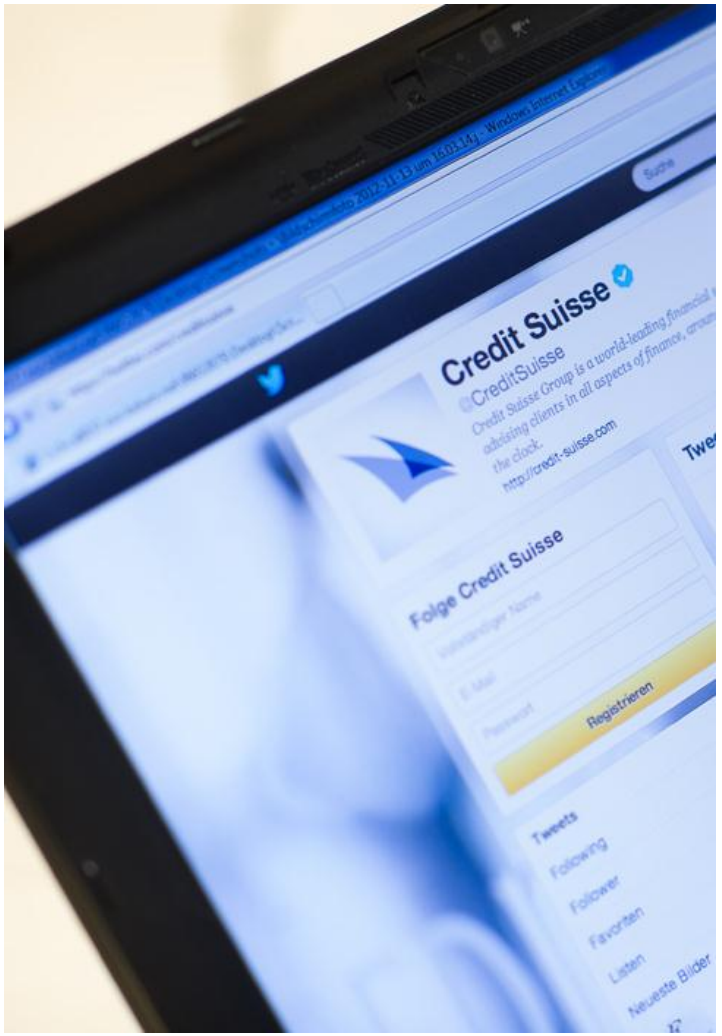
Where We Reach the Limits

Areas for Research (2/2)

- **Semantic alignment**
  - Keeping large service landscapes semantically aligned, federation
  - Semantics in dynamic service discovery
  - Systematically linking integration architecture to business architecture (information model, process model, function model)
  - Blending external standards with internal extensions

- **High volume, low latency implementations**
  - Market data, 100,000s of messages per second, distributed to many clients, publish subscribe pattern, sub-ms latency expectation
  - Special HW, FPGA
  - Special Network-Devices
  - Simplified protocols

- **Cloud services**
  - Service markets
  - Domain-based interface standardization
  - Security
  - Pricing



Thank You!

Questions