Key message

- Instead of designing a concrete solution we design a solution space

- Solution space represents a set of all possible solutions under given assumptions, constraints and design rules

- We use complexity metrics and decision trees to identify decision rules which result in a viable Global SOA
Agenda

- Introducing Credit Suisse
- SOA @ Credit Suisse
- Global SOA
- Related work
- How to build global SOA
- Roadmap
Credit Suisse is a leading global bank headquartered in Zurich. It is focused on serving its clients in three business lines: investment banking, private banking and asset management. Credit Suisse is renowned for providing expert advice, holistic solutions and innovative products to a wide range of corporate and institutional clients and high-net-worth individuals globally, as well as retail clients in Switzerland.
Credit Suisse locations
### Credit Suisse vs. peers: December 31, 2010

#### Market Capitalization, USD bn

<table>
<thead>
<tr>
<th>Bank</th>
<th>Market Capitalization</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPMorgan Chase</td>
<td>165</td>
</tr>
<tr>
<td>Citigroup</td>
<td>138</td>
</tr>
<tr>
<td>Bank of America</td>
<td>134</td>
</tr>
<tr>
<td>Goldman</td>
<td>91</td>
</tr>
<tr>
<td>Sachs</td>
<td>63</td>
</tr>
<tr>
<td>UBS</td>
<td>48</td>
</tr>
<tr>
<td>Deutsche Bank</td>
<td>48</td>
</tr>
<tr>
<td>Credit Suisse Morgan</td>
<td>48</td>
</tr>
<tr>
<td>Morgan Stanley</td>
<td>41</td>
</tr>
</tbody>
</table>

#### B/S Assets, CHF bn

<table>
<thead>
<tr>
<th>Bank</th>
<th>B/S Assets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deutsche Bank</td>
<td>2,387</td>
</tr>
<tr>
<td>Bank of America</td>
<td>2,123</td>
</tr>
<tr>
<td>JPMorgan</td>
<td>1,985</td>
</tr>
<tr>
<td>Chase</td>
<td>1,795</td>
</tr>
<tr>
<td>Citigroup</td>
<td>1,317</td>
</tr>
<tr>
<td>UBS</td>
<td>1,175</td>
</tr>
<tr>
<td>Credit Suisse</td>
<td>1,032</td>
</tr>
<tr>
<td>Goldman</td>
<td>854</td>
</tr>
<tr>
<td>Sachs</td>
<td>757</td>
</tr>
</tbody>
</table>

#### Assets under Management, CHF bn

<table>
<thead>
<tr>
<th>Bank</th>
<th>Assets under Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBS</td>
<td>2,152</td>
</tr>
<tr>
<td>Deutsche Bank</td>
<td>1,477</td>
</tr>
<tr>
<td>Credit Suisse</td>
<td>1,253</td>
</tr>
<tr>
<td>JPMorgan</td>
<td>1,217</td>
</tr>
<tr>
<td>Chase</td>
<td>788</td>
</tr>
<tr>
<td>Goldman</td>
<td>709</td>
</tr>
<tr>
<td>Sachs</td>
<td>604</td>
</tr>
</tbody>
</table>

#### Employees - Number of FTEs, t

<table>
<thead>
<tr>
<th>Bank</th>
<th>Employees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bank of America</td>
<td>287</td>
</tr>
<tr>
<td>Citigroup</td>
<td>260</td>
</tr>
<tr>
<td>JPMorgan</td>
<td>240</td>
</tr>
<tr>
<td>Chase</td>
<td>102</td>
</tr>
<tr>
<td>Deutsche Bank</td>
<td>65</td>
</tr>
<tr>
<td>UBS</td>
<td>63</td>
</tr>
<tr>
<td>Morgan</td>
<td>50</td>
</tr>
<tr>
<td>Stanley</td>
<td>36</td>
</tr>
</tbody>
</table>

1) IFRS accounting standards
2) Citigroup does no longer disclose any information on AuM

Produced by: Claudio Jossen, Roberto di Paolo, Tarmo Pлом
Date: 15.04.11 Slide 6
Agenda

- Introducing Credit Suisse
- **SOA @ Credit Suisse**
- Global SOA
- Related work
- How to build global SOA
- Roadmap
IT Complexity @ Credit Suisse

- Very large scale
  - ca 6000 applications
  - more than 100 000 000 LOC code

- High complexity
  - Large number of tightly coupled components

- Aging
  - parts of the system are becoming obsolete and must be replaced (obsolete technology, end-of-life applications)

- High change rate
  - more than 3000 production changes per week

- Demanding operational quality
  - systems must have high reliability, good availability, sufficient security etc.

=> SOA approach to manage complexity in the IT-landscape
Managed Services

- Managed services
  - Clearly defined operation signature or message/record structure
  - Usage of typed attributes with documented semantics
  - Usage of standard middleware

- Managed service categories
  - Synchronous services
  - Asynchronous messaging
  - Bulk (asynchronous communication)

- KPI-s
  - ca 1200 managed synchronous services
  - ca 10 managed bulk services
Global target operating model

**Investment Bank**  **Private Bank**

- **Client Booking and Advisory**
  - Separation of client books and records from firm books
  - Anonymization of client data at source
  - Confidential client data sequestered in “home” booking entity

- **Product Processing**
  - Single system per product, deployed in “best” location for the Bank
  - Client-independent processing
  - Multiple entity capable, supporting ‘trade anywhere, book anywhere’
  - Repository of transaction data and product positions
  - Potentially separate legal entity for processing

- **Risk mgmt and Financial control**
  - Consolidated product positions
  - Consistent reference data, pricing and models
  - Consolidated cash and securities accounting

**Shared Components**
- Reference data federated to appropriate areas
- Based on common ‘meta-architecture’
- Single, common, standards-based messaging platform

**Infrastructure**
- Rigorous adherence to platform standards
- No compromise on resiliency/availability
- Scalability is a must

Produced by: Claudio Jossen, Roberto di Paolo, Tarmo Ploom
Date: 15.04.11 Slide 10
Agenda

- Introducing Credit Suisse
- SOA @ Credit Suisse
  - Global SOA
- Related work
- How to build global SOA
- Roadmap
3 Main dimensions of SOA

- **Logical view**
  - Main area of IT architecture governance
  - There are lots of existing assets to rely on, e.g. design guidelines
  - Hard to change but very sustainable on the long run

- **Security view**
  - Essential for any international company (due to external/internal policies and law)

- **Infrastructure view**
  - Mostly solved on a local or even regional level
  - Adjustable through central IT strategy and respective investments
7 sub dimensions of local SOA

- **Logical view**
  - Orchestration variability

- **Security view**
  - Authentication: no, week, strong
  - Authentication systems variability
  - Authorization systems variability
  - Trust zones

- **Infrastructure view**
  - Middleware variability
  - Executables distribution variability
21 sub-dimensions in global SOA

- **Logical view** (additional sub-dimensions of global SOA)
  - Business process variability
  - Workflow variability
  - Federation variability
  - Business rule variability
  - Service variability
  - Meta-/Data variability

- **Security view** (additional sub-dimensions of global SOA)
  - Identity management: central vs. federated
  - Role management: central vs. federated
  - Standardized realms vs. non-standardized realms
  - Legal data flow restrictions

- **Infrastructure view** (additional sub-dimensions of global SOA)
  - Executables reuse globally
  - Middleware reuse globally
  - Registry, repository, BPMS and BRMS distribution & visibility
  - Service namespaces
Complexity explosion

- 21 different complexity dimensions by global SOA
  - If each dimensions has four variants than overall $4^{20}$ (ca $10^{12}$) solution options

- Lets make the assumption that logical, security and infrastructure views are independent

- In above example
  - $4^7$ (16384) variants for logical view
  - $4^8$ (65536) variants in security view
  - $4^6$ (4096) variants in infrastructure view

- How long would it take time to find solutions from $4^7$ options?
  - If we use 1 minute per option then we need 34 months
How to build global SOA in finite time?
Agenda

- Introducing Credit Suisse
- SOA @ Credit Suisse
- Global SOA
- Related work
- How to build global SOA
- Roadmap
Overview of related work

- **SOA Layering**
  - Macro flow composition layer, macro flow integration layer, dispatching layer, micro flow execution layer [Zdun]
  - Enterprise layer, process layer, intermediary layer, basic layer [Slama]

- **Decision models**
  - ATAM [ATAM]
  - Architectural decision models [Zimmermann]

- **Ultra large scale systems**
  - ULS [SEI]

- **Socio technical systems view**
  - Interactions of people and technology at the workplace in the organizational context
Agenda

- Introducing Credit Suisse
- SOA @ Credit Suisse
- Global SOA
- Related work
- How to build global SOA
- Roadmap
Solution space overview

- Usually architecture focuses on design of a concrete solution
  - In focus is finding of "the best" solution in given assumptions and constraints
  - Usage of decision trees or tradeoff analysis (ATAM)

- Instead of designing "the best" solution we focus on the design of "the best" solution space.

- An optimal solution space represents set of all possible solutions which are possible under
  - given assumptions
  - design constraints
  - design rules
Global SOA solution space

- **Solution space**
  - Set of all possible solutions to a given problem.
  - Set of solutions that satisfies all assumptions, constraints and design rules.

- **Global SOA solution space**
  - Solution space defines of all possible theoretical solutions under given assumptions, constraints and rules
  - Complexity of solution space is the maximum complexity of solution

- **Each SOA (sub-)dimension has a set of complexity functions**
  - Complexity function represents all possible variants of a dimension
  - Mathematical or graphical representation

- **All complexity function sets build a function space – the Global SOA Solution Space**
  - Solution space defines of all possible theoretical solutions
  - Complexity of solution space is the maximum complexity of solution
Maximum complexity

- Solution space can be represented as a graph which represents all possible solutions.
- To measure variety of such a graph we use cumulative component dependency (CCD) [Lakos] which measures the number of traversal paths in a graph. Also we make the assumption that probabilities for the occurrence of each traversal path are equal.
- CCD is related to system entropy

\[ S = -k \sum_i P_i \ln P_i \]

- We define maximum complexity of the system as the maximum CCD (MCCD) which is possible under a given design rules.

\[ \text{MCCD} = f \text{ (design rules)} \]

=> Design rule which results in smaller maximum complexity is preferred over design rule with higher maximum complexity.
Global SOA decision tree

Business Constraints
- Unpredictable
- Highly flexible
- Root of the tree

Design rules
- Defined by IT architecture
- Detailed subtrees

Produced by: Claudio Jossen, Roberto di Paolo, Tarmo Ploom
Date: 15.04.11 Slide 23
Decision sub trees

- All Global SOA (sub-)dimensions need to be considered for decision sub trees, some of them actually require sub trees of their own.

- Decisions trees are based on design rules and other types of IT constraints.

- Most decisions are about the service scope (i.e. global or local).

- Global SOA Sub dimension decision trees can be added to the overall Global SOA decision tree.

- The actual usage of the sub trees might be done through check lists where local is the the dominant attribute (i.e. if one check requires a local service and all other checks indicate a global service, than it has to be a local one).

```
Data Variability

No

Local

Yes

Global

Can service separated to local and global parts?

No

Local Service

Yes

Global Service + Local Service + WF
```
Cohesive design rule sets

- First three layers are out of control by IT. They are driven by ever changing business environment influences. These layers have to be described by
  - constraints and
  - assumptions

- We have to accept environmental flexibility and focus on building up viable SOA solution inside given constraints and assumptions.

- Use MCCD to identify preferred design rules

- Use Global SOA decision tree to
  - achieve cohesive sets of design rules
  - connect different design rules to each other
  - show different levels of design rules
Orchestration example 1/5

Assumptions
- every element can be global
- every element can be invoked according to orchestration layering (top down)
- global and local variants are distinct (no global service is equal to a local one)
- each element is used only once in a sequence
- topology is not considered (single implementation = one global variant, multiple local variants)
Orchestration example 2/5

- **Symbols**
  - \( m_g \) – number of macro flows (global)
  - \( m_l \) – number of macro flows (local)
  - \( n_g \) – number of micro flows (global)
  - \( n_l \) – number of micro flows (local)
  - \( c_g \) – number of composite services (global)
  - \( c_l \) – number of composite services (local)
  - \( a_g \) – number of atomic services (global)
  - \( a_l \) – number of atomic services (local)
  - \( k \) – number of locations

- **Complexity Function**
  \[
  \text{MCCD} = (m_g + m_l k + 1) \times (n_g + n_l k + 1) \times (c_g + c_l k + 1) \times (a_g + a_l k)
  \]
Orchestration example 3/5
Orchestration example 4/5

- **Constraint**
  - Each layer is only allowed to access/use the next deeper level.
  - Hence, all intermediary levels have to be implemented.

- **Symbols**
  - $m_g$ – number of macro flows (global)
  - $m_l$ – number of macro flows (local)
  - $n_g$ – number of micro flows (global)
  - $n_l$ – number of micro flows (local)
  - $c_g$ – number of composite services (global)
  - $c_l$ – number of composite services (local)
  - $a_g$ – number of atomic services (global)
  - $a_l$ – number of atomic services (local)

- **Adopted Complexity Function**
  - $\text{MCCD} = (((m_g + m_l) \times (n_g + n_l) + 1) \times (c_g + c_l) + 1) \times (a_g + a_l)$
Orchestration example 5/5

- Random Example
  - 10,000 atomic services
  - 2,000 composite services
  - 500 micro flow services
  - 100 macro flow services
  - 3 locations
  - local/global ratio is e.g. 1:9 (9 services out of 10 are local)

- Orchestration complexity without constraint and local/global ratio 1:9:
  61,740,203,868,000 variants

- Orchestration complexity with constraint and local/global ratio 2:8:
  45,873,495,226,000 variants

=> Difference of about 1/4!
Agenda

- Introducing Credit Suisse
- SOA @ Credit Suisse
- Global SOA
- Related work
- How to build global SOA
- Roadmap
Managed Evolution

Balanced Development of IT Efficiency and Business Value

Driver: Solution delivery and IT operations efficiency

Project Contribution to Business Value

Unbalanced Development

Project Contribution to IT Efficiency

Unbalanced Development

Driver: Business requirements, Time to market

Business Value

Managed Evolution: Balanced Development of IT Efficiency and Business Value

[Murer]
Summary

- **Building up Global SOA**
  - Start first with small set of design rules
  - Ensure in practice that these rules are cohesive

- **Continue with iterations**
  - Revise existing design rules
  - As next identify and verify next design rules using MCCD
  - Use decision trees to ensure cohesiveness of new design rules
  - Apply new rules

- **Objectives**
  - Global SOA as a viable system
  - Managed complexity growth of Global SOA
  - Controlled MCCD of Global SOA

- We have always tried to come up with design principles and design rules, but we did not consider potential side-effects from other (sub-) dimensions

- Only difference is that by using solution space approach, we can consider all dependent (sub-) dimensions
References

[Murer] Stephan Murer, Bruno Bonati, Frank J. Furrer; Managed Evolution, A Strategy for Very Large Information Systems; Springer Verlag, 2011

[Zdun] Uwe Zdun; Modeling Process-Driven and Service-Oriented Architectures Using Patterns and Pattern Primitives; ACM-TRANSACTION September 19, 2007 23:50


[ATAM] ATAM: http://www.sei.cmu.edu/architecture/tools/atam/

[Zimmermann] Olaf Zimmermann, Jana Koehler, Frank Leymann; Architectural Decision Models as Micro-Methodology for Service-Oriented Analysis and Design


Q&A?