EVOLUTION NOT REVOLUTION ?

The Data Warehousing Strategy at Credit Suisse Financial Services

Dr. Markus Tresch

ETH Zurich, July 4, 2002

TABLE OF CONTENTS

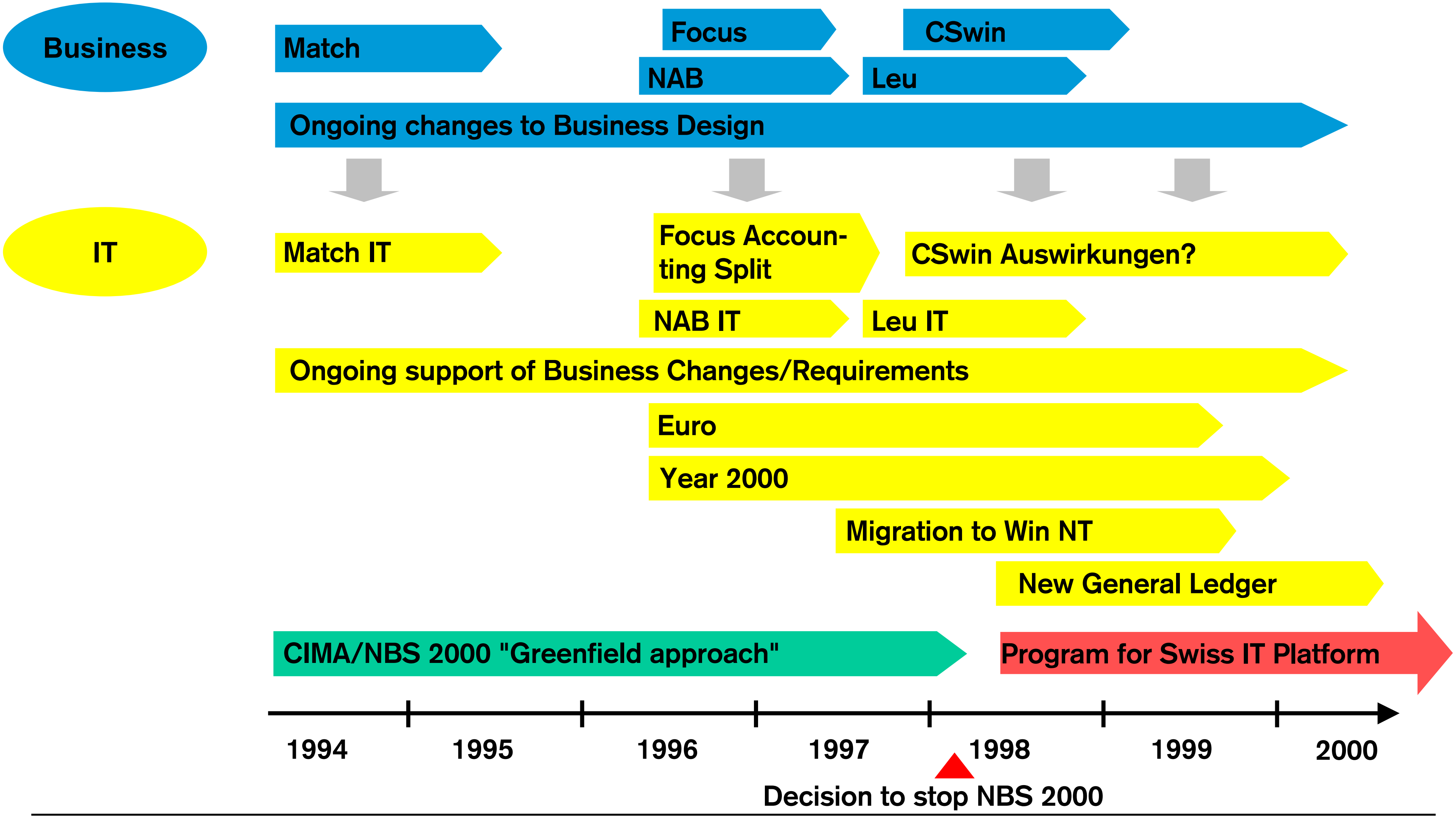
Introduction

Strategy of Managed Evolution

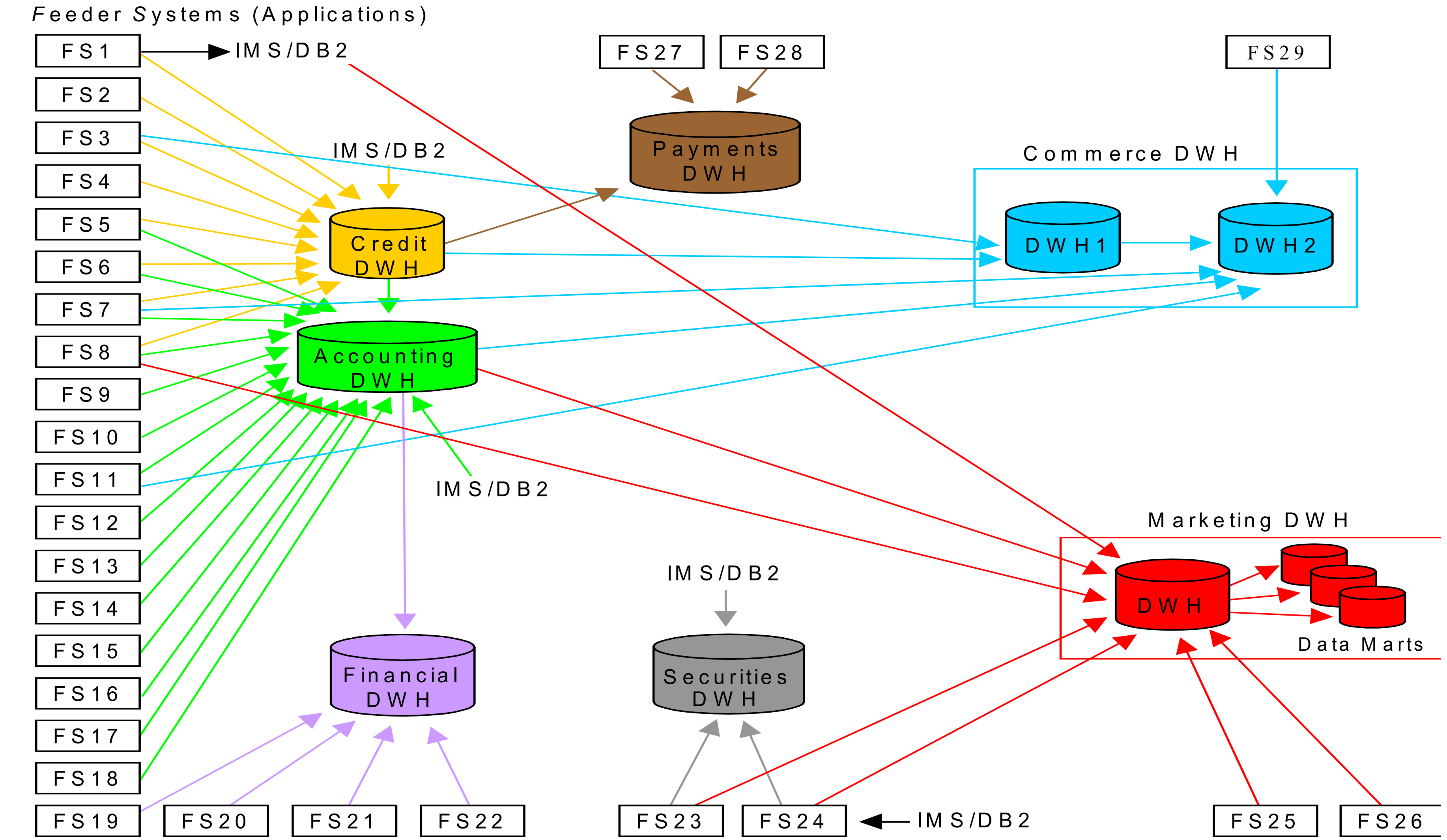
Data Warehouse Re-Architecture Program

Experiences and Open Issues

INTRODUCTION
CHANGES OF BUSINESS DESIGNS


Business
Match
Focus
CSwin
NAB
Leu
Ongoing changes to Business Design
IT
Match IT
Focus Accounting Split
CSwin Auswirkungen?
NAB IT
Leu IT
Ongoing support of Business Changes/Requirements
Euro
Year 2000
Migration to Win NT
New General Ledger
CIMA/NBS 2000 "Greenfield approach"
Program for Swiss IT Platform
1994
1995
1996
1997
1998
1999
2000
Decision to stop NBS 2000

INTRODUCTION
HISTORICALLY EVOLVED DATA WAREHOUSES



INTRODUCTION
HISTORICALLY EVOLVED DATA WAREHOUSES

Business View

- end-to-end ownership / sponsorship of the data warehouses
- no data quality process in place
- many novel complex requirements (General Ledger, CRM, Channel Management, ...)
- different states in product life cycle (antic … state-of-the-art)

Application View

- about 12 Data Warehouses in Credit Suisse (existing and planned)
- complex application dependencies
- no overall view on what data ist stored where and used for what (meta data)

Technical View

- many different heterogeneous platforms, tools, architectures, ...
  - mainly Mainframe / DB2, IBM RS 6000 SP / DB2, Sun E10000 / Oracle, Host / SAS
- variety of SW tools and products
  - SAS, ETI, Trillium, PowerCenter, Microstrategy, Darwin, Brio, …

INTRODUCTION
IMPACT ON DATA WAREHOUSING

- Data warehouse development has become expensive and slow because of the high complexity of the historically evolved environments and the very different warehouse platforms being used. Even worse, a stable production environment could not be provided any more.
- A high percentage of resources were required to *maintain* old systems instead of developing new solutions for business users.
- The integration of off-the-shelf software was very difficult because interfaces were either not compliant with standards or were not clearly defined and documented.
- The implementation of new business requirements, like alternative distribution channels, was very hard.
- An enterprise-wide data quality process could hardly be established, since each warehouse implemented its own data semantics.

TABLE OF CONTENTS

Introduction

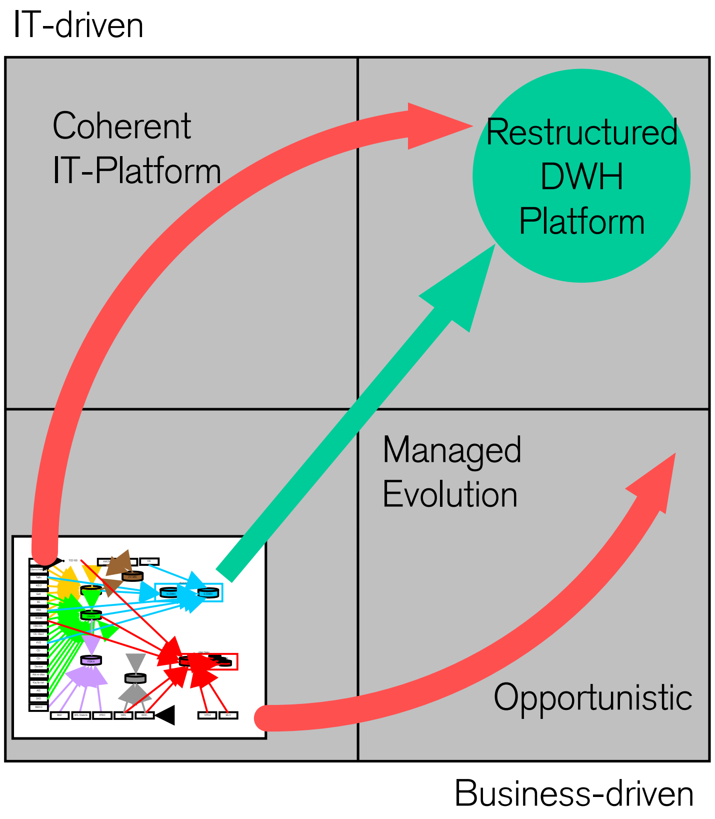
Strategy of Managed Evolution

Data Warehouse Re-Architecture Program

Experiences and Open Issues

STRATEGY OF MANAGED EVOLUTION

MAJOR CHOICES



1. In-house developed "greenfield" solution	No option: too high risk and cost, lack of skills
2. Migration to standard package solution	No general option for products and volume of target solution
3. Take-over proven solution of another bank	Only realistic in general M&A situation
4. Opportunistic enhancement	No long term option: cost, risk and time-to-market gradually decreases
5. Managed evolution	Realistic approach for this very demanding goal

STRATEGY OF MANAGED EVOLUTION
BASIC IDEA

Well balanced approach, between

- short-term implementation of new business requirements (quick wins) and
- long-term improvement of IT architecture

Principles

- each data warehouse project has to provide ist share to the completion of the DWH strategy
  - building of new data bases
  - consolidation of previous applications
  - definition of clear interfaces
  - replacement of legacy systems
- continuous monitoring of well defined development of the target IT Architecture
- step-wise extension of the DWH platform, such that each single step contributes to the final solution
- structuring the system into modules, each of which can be extended or replaced by a standard software package

STRATEGY OF MANAGED EVOLUTION
BUSINESS CASE

Business return and vision

- a DWH platform for comprehensive analysis of operational banking businesses
- an enabler für novel strategic banking businesses, like bank-assurance / all-finance

Quantifyable results (examples)

- reduce developing costs
- implement data quality control loops
- improve time-to-martket

Complexity reduction

- recucing the number of warehouses from 12 to 6
- reducing the number of platforms from 3 to 1
- reducing all multiple data extraction to 1 per feeder
- reducing the number of used tools to 1 per category

TABLE OF CONTENTS

Introduction

Strategy of Managed Evolution

Data Warehouse Re-Architecture Program

Experiences and Open Issues

DATA WAREHOUSE RE-ARCHITECTURE PROGRAM
PROJECT MANDATE

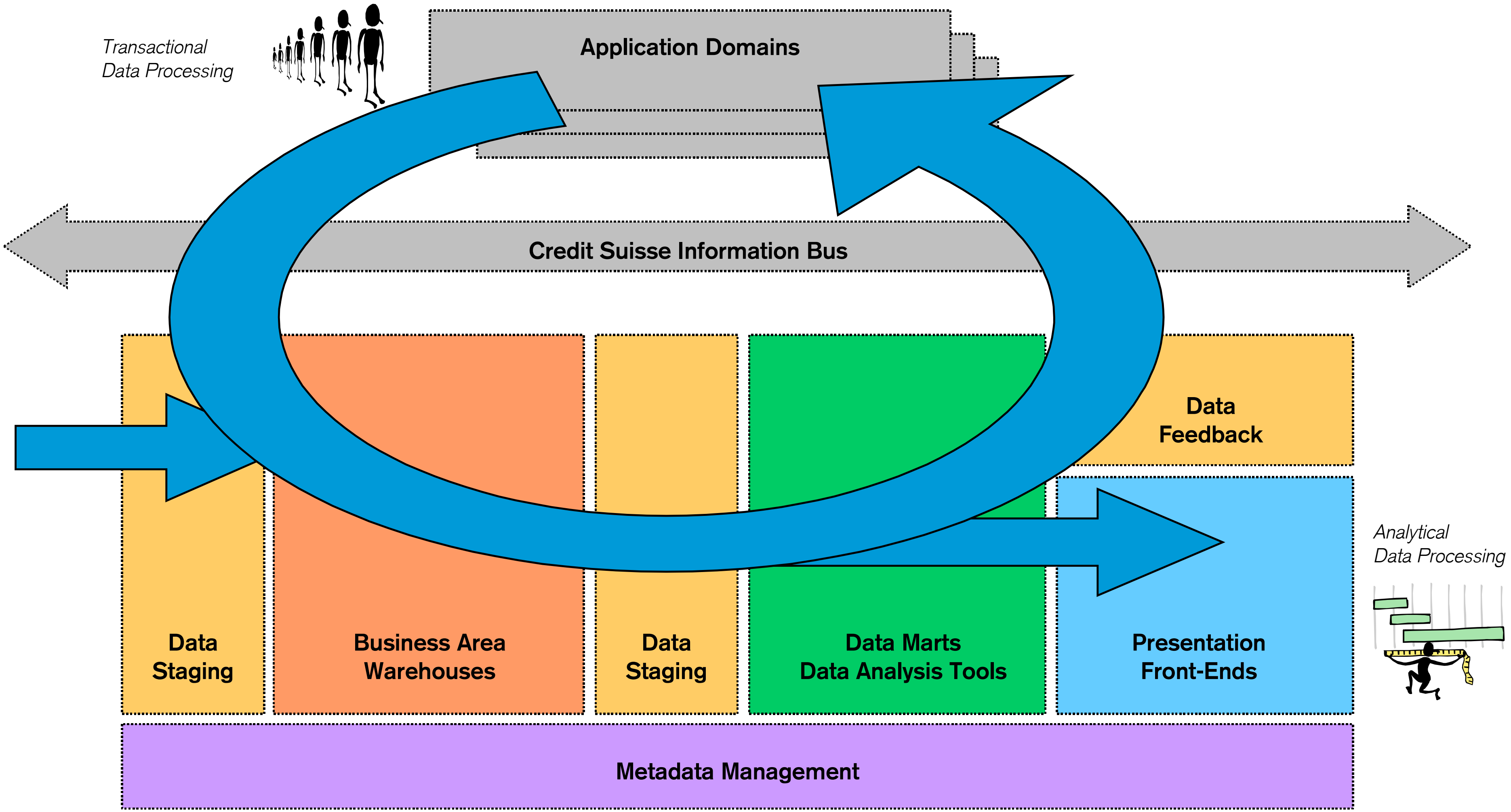
Data Warehouse Strategy

- migrate into an enterprise wide data warehouse target architecture following the principle of managed evolution

Project Mandate

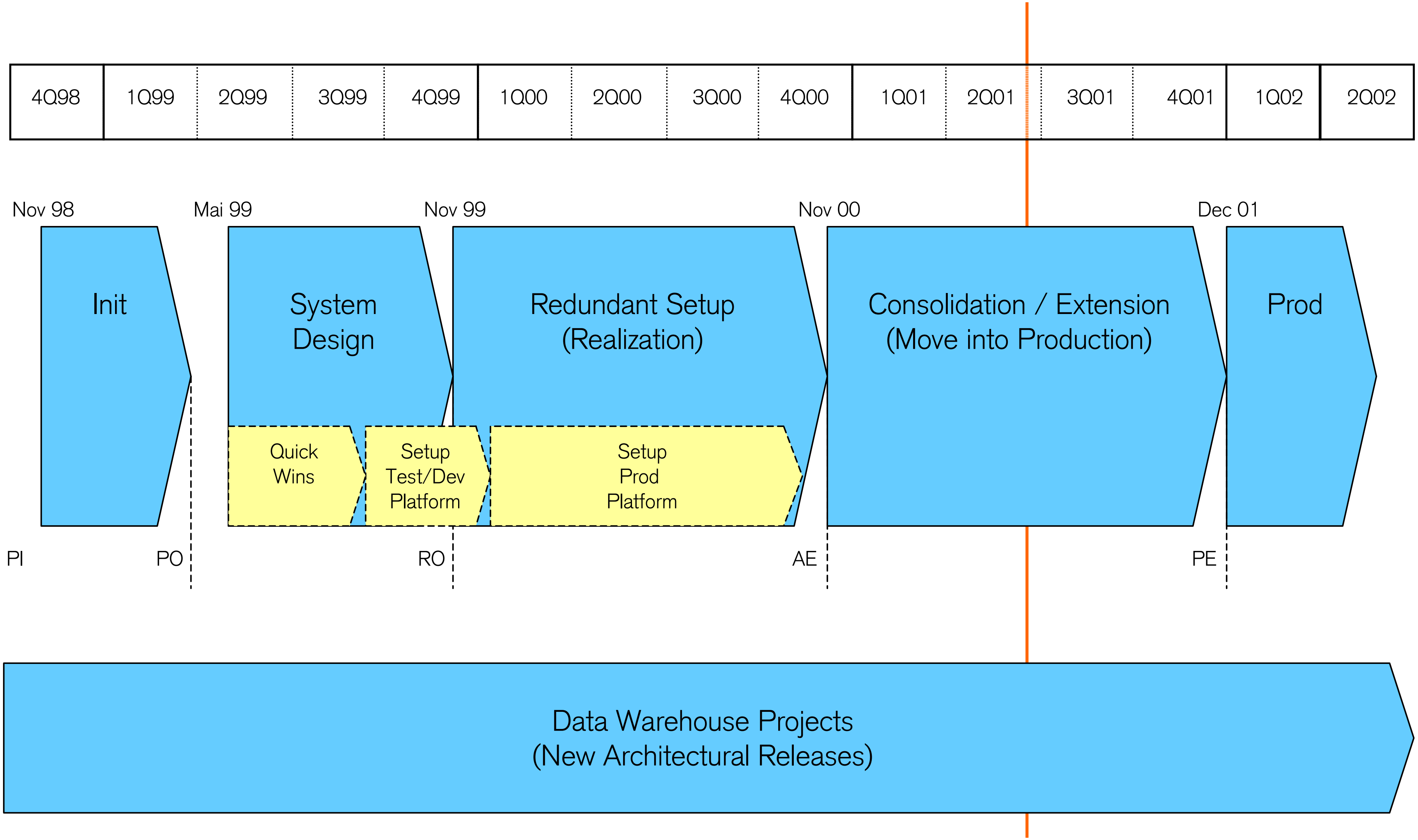
- Develop and maintain an overall DWH Architecture (target architecture)
  - Application, Data and Meta Data Architecture
  - Software Architecture (Reuse CS Thin Client 3-tier Architecture)
  - Technical System and Security Architecture
- Develop a migration path for all DWH applications towards the target architecture
- Ensure the conformity of all implementation projects to the target architecture
- Define the standard development tool set for Data Warehouse Design, Implementation and Maintenance
- Define, build and maintain a dedicated technical Data Warehousing Platform
- Develop and implement Business Organisation and Processes/Responsibilities to manage DWH

DATA WAREHOUSE RE-ARCHITECTURE PROGRAM
OVERAL TARGET ARCHITECTURE



DATA WAREHOUSE RE-ARCHITECTURE PROGRAM

RE-ARCHITECTURE MASTER PLAN



CREDIT SUISSE | FINANCIAL SERVICES

Date July 2002
Author Markus Tresch

TABLE OF CONTENTS

Introduction

Strategy of Managed Evolution

Data Warehouse Re-Architecture Program

Experiences and Open Issues

EXPERIENCES AND OPEN ISSUES
WHAT WE LEARNED 1/3

- The platform proved to be very powerful and reliable
  - The staging process is on average 4 times faster than on the old platforms
  - Some performance problems are caused by poor physical designs of data marts or poor application designs
  - An efficient load process becomes increasingly important, batch load window becomes smaller and smaller
  - The next release of the reference architecture will include an operational data store (ODS) as part of the staging area to support nearly real-time warehousing
  - One should not spend too much time on evaluating platforms
- The new architecture allows for a very quick implementation of new data marts
  - True, if the required data are already available in business area warehouses and the project team is familiar with the environment
  - Now a completely new data mart can be implemented within 3 months (from design to production).

EXPERIENCES AND OPEN ISSUES
WHAT WE LEARNED 2/3

- Interface management between the warehouse platform and the feeder systems is crucial
  - In the past, the warehouse projects were responsible for implementing the data extraction programs (usually in PL/I) in order to generate the feeder files (pull principle)
  - When a feeder system is changed, the extraction programs have to be adapted, too. In a large-scale environment with hundreds of feeder systems and dozens of warehouses, such a synchronization is not easy.
  - Moreover, if the warehouse platform and the platform of the feeder system are different, the warehouse project team has to maintain additional skills which often results in a waste of resources.
  - Therefore, we switched over to a *push principle*. The warehousing and the feeder system projects sign *service level agreements* that define which data must be delivered in which format at which time to the warehouse platform.
  - When the push principle is used, warehouse systems need not care whether the feeder files have been generated by a data extraction program during the end of day processing, or have been generated continuously using, e.g., message-oriented middleware

EXPERIENCES AND OPEN ISSUES
WHAT WE LEARNED 3/3

- Reengineering existing systems ”on-the-fly” is a tedious process
  - It can only be successfully completed if this process is strongly supported by the top management
  - A business steering committee must be established to decide strategic issues of the whole warehousing platform and keeps all business projects on track.
  - A general problem is that it is easy to find business owners for data marts, but much more difficult to find business owners for business area warehouses
  - Clearly defined responsibilities are even more important for business area warehouses than for data marts, since these databases are the common foundation of all data marts
  - The organization of data warehouse developers needs to be adapted in accordance with the new architecture

EXPERIENCES AND OPEN ISSUES
WHERE WE DIDN'T MAKE THE EXPECTED PROGRESS 1/4

- Metadata management
  - metadata management tools are still immature
  - warehousing tools (in particular staging and reporting tools) have poor interfaces to upload metadata
  - an automated management of metadata (bidirectional) is essential for end user acceptance
  - user-friendly interfaces are required to present these data to the end users
  - data lineage at data item level is a fundamental problem where script-based transformation rules (staging tool) and reporting *programs* make it very questionable whether it is possible to achieve any progress at all
  - at least a case study would be helpful that describes examples of successful metadata management implementations, highlights success/failure factors (from the technological to the business point of view) and gives a deeper understanding of realistic expectations
- Archiving in the context of evolving data models (schemas)
  - it is easy to generate backups, but systems evolve over time
  - In most cases, it is not feasible to migrate all existing backups when new system releases are introduced

EXPERIENCES AND OPEN ISSUES
WHERE WE DIDN'T MAKE THE EXPECTED PROGRESS 2/4

- Systems management (in particular performance management)
  - how to guarantee a defined minimum of system resources (in particular CPU) for critical applications if database instances are shared
  - performance management is only possible either at operating system level or database level, both are not compatible with each other
  - Another issue is performance monitoring in n-tier environments. It is easy to check whether the warehousing components are up and running. However, it is hard to find out where the bottleneck actually is if end users experience a poor end-to-end performance
- Performance improvements for on-line analytical processing (OLAP)
  - OLAP based on very high data volumes (terabyte range) is only possible using relational OLAP (ROLAP)
  - response times of ROLAP tools are rather high, hence we would prefer to use a multidimensional or hybrid OLAP system
  - these tools are not yet able to handle such high data volumes, in particular if the underlying data change every day (creating a huge data cube can take several days)

EXPERIENCES AND OPEN ISSUES
WHERE WE DIDN'T MAKE THE EXPECTED PROGRESS 3/4

- Implementing security (in particular access control)
  - over the whole warehousing chain such that data are consistently protected irrespective of their actual location and representation is almost impossible
  - today's warehousing tools each have their own security manager
  - a comprehensive end-to-end security model would be required that allows for an integrated privilege management over heterogeneous tools (including ad-hoc reporting)
  - the provision of a *complete* audit trail over all warehousing tools is an open issue, all tools must provide for the required log data and these log data must be integrated into a global log database
  - what we need is a publish/subscribe mechanism for all tools so that warehouse administrators can collect all data needed according to *their* requirements without introducing too much overhead on the warehouse platform

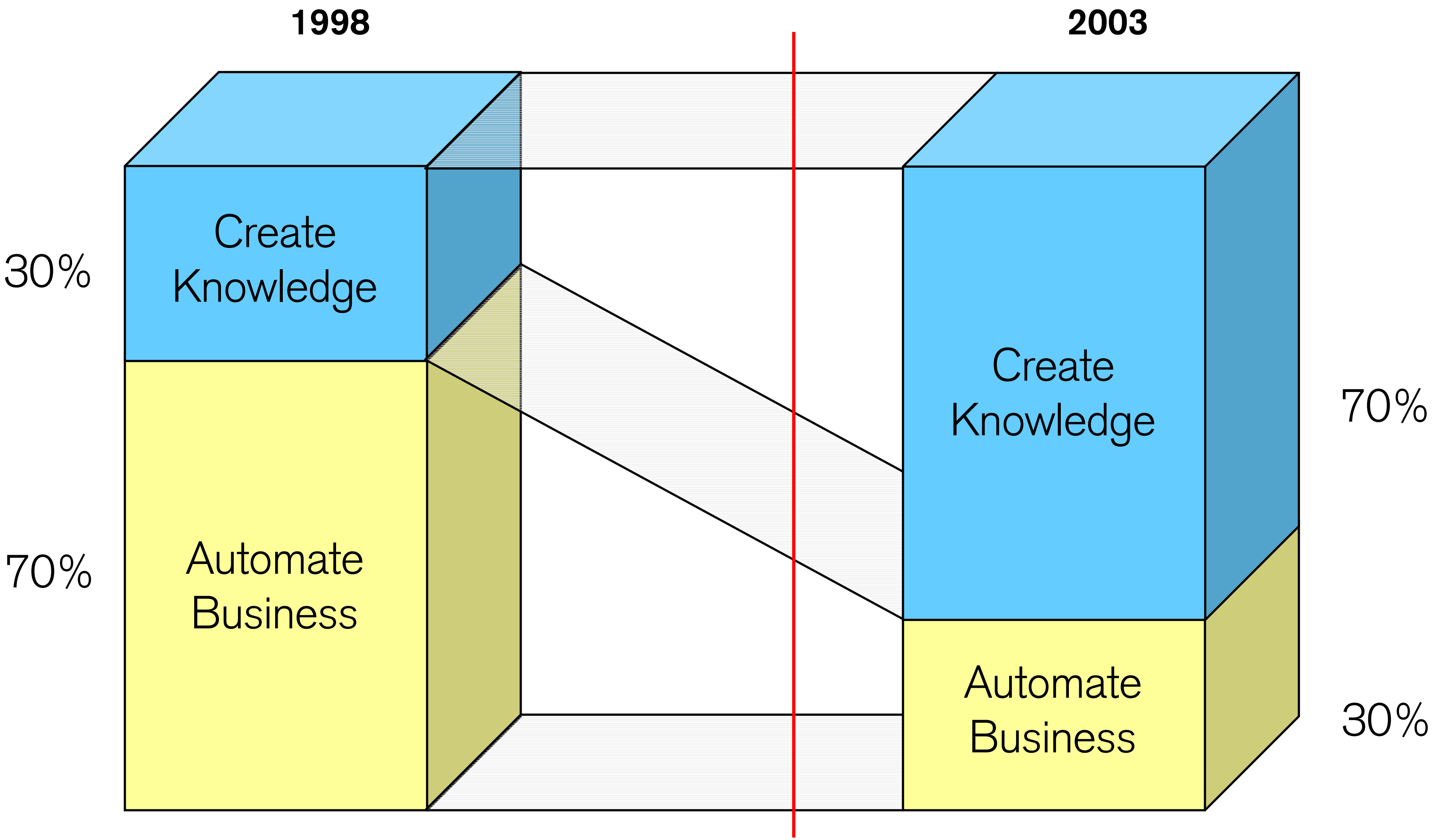
EXPERIENCES AND OPEN ISSUES
WHERE WE DIDN'T MAKE THE EXPECTED PROGRESS 4/4

- Data quality management
  - more than 500 feeder systems deliver data to our warehouse platform, the data quality differs considerably
  - warehouses can only measure the quality of data to some degree (using heuristics), but they cannot improve the quality
  - Data quality management has to be a business-driven process where the correction of errors is initiated by the business users
  - Unfortunately, business users often argue that these corrections should be made in the data warehouse. We strongly argue *not* to apply this "solution". The resulting system will tend to be unmanageable, since cyclic dependencies between systems at the data instance level are never fully understood in large-scale environments
- Management of semistructured and multimedia data
  - There are plenty of examples of such data in e-commerce applications, e.g. contracts, insurance policies, geographical data for mortgages, and for risk management in the insurance business
  - Today's analysis tools (reporting, OLAP and data mining) are not ready yet to cope with such kinds of data

CONLUSION

THE SHIFT IN SOFTWARE BUDGET ALLOCATION

Source: Meta Group, 1998



EXPERIENCES AND OPEN ISSUES
CONTACT

Dr. Markus Tresch

CREDIT SUISSE FINANCIAL SERVICES
Technology & Services
Cross Platform Services / KTX
P.O. Box 600
CH - 8070 Zurich

Tel	+41 1 332 43 43
Fax	+41 1 332 55 00
Email	markus.tresch@csfs.com
Internet	www.csfs.com