GTUG presentation
comForte’s solutions in the modernisation arena

27 April 2017,
Helmut Bernhard (comForte EMEA Senior Solutions Architect)
Every industry has its modernisation wake-up calls

> The telecommunications industry (1996)
> The banking industry (2010)
> Encore - The banking industry (2012)
> From M2M to IoT (2014)
Modernisation and transformation (Finance example)

1960s
- Semiconductor microprocessors
  - Allowed the replacement of physical recording by digital data

1970s
- Mainframes
  - Enabled batch overnight processing

1980s
- Terminals and PCs
  - Automated banks and branches and facilitated offline remote banking

1990s
- Local networks
  - Enabled data centres, intranets and corporate systems
  - Facilitated the global exchange of data and enabled a series of international businesses

2000s
- Internet
  - Created a new medium to interact with clients and collect data
  - Spearheaded frictionless payments

2010s
- Smart devices
- Mobile
  - Emerging technologies
  - Future

- Biometrics
- Cloud computing
- Cognitive computing
- Distributed ledger technology
- Machine learning / predictive analytics
- Quantum computing
- Robotics

Credit cards
- Messaging services (e.g. SWIFT)
- ATMs
- Electronic trading
- Digital banking

many transformative new technologies that will shape future financial services infrastructure and should be seen as part of a toolbox
Modernisation software architectures – scenario

> Mainly End-State software architectures found in the software arena → a risky option?

> The „usual“ promises, patterns and wording
  > Empowered by enabling infrastructure
  > Ensuring enterprise operation, continuity and stability
  > The architecture is solid and unbreakable

> The consequences of such theoretical (and yet unproven) designs and speculative architectures
  > Unable to launch software products on time
  > Sky rocking cost of the software development
  > Loss of potential revenue
The „Incremental Software Architecture“ is needed!

ComForte, customers, implementation partners and the solution partner did:

> extensive architecture discovery and analysis to generate an incremental software architecture
> Architecture decomposition (structural, behavioural and volatile attributes)
> Architecture verification (design substantiation, architecture stress-testing and capacity planning)

> To allow an incremental software Architecture following the
  > EA/Microservices principles and
  > Satisfies the management & business levels down to the technological & system levels

> All needed → to overcome lack of interoperability, performance degradation, over expended system federation, low scalability, low reuse and too tight coupling
Starting Point “Green Screen” 6530 access to Pathway Servers

1. “un natural” navigation requires extensive user training
2. Restrictive access
3. Limited to SCOBOL features
4. Requires skilled developer to enhance
comForte’s approach to modernisation

1. Client developed using familiar/modern tools (e.g., VS.NET and Eclipse)
2. Full benefits of modern GUI
3. Unlocks full potential of Pathway business logic
4. Minimal user training
Use case synchronized Travel Plan

> Leverage existing features/function of Pathway servers

> Provide secure services with tailored features to external users

- Access other SOA enabled platforms from the same client
Use Case Mobile Payment Transaction Web Services

REST/JSON Client
(CSL REST API Doco)

NonStop System

ISO Message

HTTP Server

REST (JSON)/HTTPS

DDL

Custom Pathway Server

CSL REST Server Wrapper

BASE24

XPNET
Use case „revitalizing”

> Client applications to enrich legacy apps
  > Temporary limits management
  > Merchant enquiries
  > Different architectures (SOAP, JSON/REST)

> Add new functionality with JEE
  > UI control
  > Message converters
  > New business logic
  > 2FA (Client Authentication)
  > Payment protocol adapters (ISO8583, SPDH, etc.)
CSL REST environment

Using CSL Studio:

> Define service based on DDL and Pathway info
> Generate REST API Doco to be used by client developer
> Generates “REST Wrapper” and can automatically deploy it to the NonStop
Use case travel industry
Use case NonStop Host in a BPEL environment
ComForte offerings in the Modernisation arena

> **JPath** – Modernising the graphical user interface
  > Keeps application unchanged
  > Only transforms screen(s) to GUI

> **CSL** – Modernising the interface to Pathway servers
  > Write software on other platforms than NonStop
  > Have this software communicate with services on NonStop
  > Expose existing NonStop applications as services
  > Foundation for Cloud, SaaS, DaaS, etc.

> **Escort SQL** – Modernising the database
  > Converts Enscribe databases to SQL/MP databases, with coexistence and without downtime
  > Modernises at database layer
  > No application source code changes
JPath - Approach

JPath - Dynamic Engine
- datastream management
- dynamic form definition
- transform fields
- dynamic layout changes
- Java GUI
- HTML GUI

HP NonStop
- SCOBOL Requestor
- Pathway Server
- Enscribe DB

configuration Rules

JPath Author

Telnet (6530/3270)
Summary Feature List

> Level 1
  > Out-of-the-box graphical interface to legacy screens
  > Function key choices displayed as buttons in a buttonbar
  > Solid or gradient colours derived from the host colours
  > Background images
  > Graphical images derived from host screen character patterns
  > Create push button menus directly from host screen choice lists
  > Support of double byte characters (Chinese, Japanese, Korean) and IME
  > Secure SSL or SSH connections

> Level 2
  > Graphical screens in proportional font
  > Replace simple data entry fields with checkboxes, dropdown boxes, radio buttons calendars etc.

> Level 3
  > Ability to specify a specific form layout

> Level 4
  > Automatic or manually triggered navigation macros
  > On-line translation of screen texts into a different language
  > Development of extensions by help of API
CSL Features

> CSL offers three approaches for Web service enablement:
  > **Approach 1**: CSL APIs
  > **Approach 2**: CSL Stubs (generated by CSL Studio)
  > **Approach 3**: CSL SOAP (WSDL generated by CSL Studio)

> Guardian-based Web service enabling solution (OSS not required)

> Requires no changes to Pathway servers

> Provides powerful GUI Web service enabling tools (CSL Studio)

> Highly-available and scalable
CSL Approach Summary

> Approach 1: CSL Adapter APIs
  > Low cost, entry-level solution
  > RSC replacement (without code change)
  > Client is responsible for construction of the native message

> Approach 2: CSL code stubs
  > Ideal for high volume, performance demanding environments
  > Rapidly deploy solution using CSL Studio
  > Supports mixed client environments (requires multiple code stubs)

> Approach 3: CSL SOAP
  > Ideal for mixed client environments (one WSDL satisfies all clients)
  > Enables both SOAP Server and SOAP Client capability
  > Rapidly deploy Web service solution using CSL Studio
  > *SOAP involves verbose messages—up to ten times the size of the native message—which can affect performance
  > *Converting messages from SOAP (XML) may be CPU intensive

*Note: SOAP characteristics applies to any SOAP implementation
Architectures in modernisation and connectivity arena

> Connection / interaction with:

> RSC = Realisation of inter-process communication

> J2EE = Java Enterprise Edition, middleware and platform for development and execution of Java

> .NET = MS platform for development and execution of application programs for various languages

> SOAP = network protocol for data transfer between systems as well as Remote Procedure Calls

> JSON Rest = Programming style that one URL is always one specific page

> Java API = Java platform for desktops, laptops, PDA, mobile phones communication with HP NonStop Pathway Server
Use case development under VS.NET/Eclipse

> Define service based on DDL and Pathway info (via CSL Studio)
> Generate WSDL
> Generates “SOAP Wrapper” and can automatically deploy it to the NonStop
Use case Smart Modernisation (final stage)

> CSL SOAP enable Pathway servers for Web service access to Pathway servers

> JPath
  > replaces terminal emulation – i.e. connects via Telnet protocol – Scobol unchanged
  > for rapid conversion of “Green Screen” to GUI. Data transfer via JPath Java API

> Coexistence/Migration Enscribe access via ODBC via Escort SQL
Escort SQL - Product Summary & Benefits

> Common data access without reprogramming

> Incremental Migration
  > Replace one file at a time with well designed NonStop SQL table(s)

> Convert any file, not just the “easy” ones
  > Converts challenging legacy schemas to well-designed, normalized, standardized SQL

> Execute existing programs with SQL tables
  > No reprogramming required
  > Excellent performance

> Lowest risk, highest return on investment
Architecture Nonstop IBM connectivity

API’s
- HLS
- APC
- SNALU
- Raw Sockets
- Pathsend
- IPC

uLinga Core
- Support for all NonStop platforms including HPE Nonstop X and Windows and LINUX
- SNA routing & switching functions (e.g. actPU/LU, bind, etc.)
- GUI based Management Cockpit
- Extensive tracing and logging facilities
- Support for legacy SNA environments (e.g. AS/400 via DLUR/S)

IBM-Interfaces
- EE (Enterprise Extender)
- IMS
- CICS
- DLSw
uLinga news 2016/2017

> DLUR/S for legacy LU’s (e.g. AS/400, i/400)
> X.25 over TCP/IP
> IPM (for specials like support of Safenet and other TCP/IP clients
Why comForte as a partner?

> 30+ years experience in HP NonStop server market place
  > the largest HPE Nonstop ISV
    > Growing Solution Set through
      > In-house Development
      > Acquisition

> Customers work with comForte because they need to
  > Lower Cost of Operations of HP Nonstop Applications
  > Lower Operational Risk by enhancing data protection and overall security
  > Support New Business Initiatives driving new application developments
  > Better integrate HP Nonstop within Enterprise IT
  > “Understand” the strategic importance of HP Nonstop in their Enterprise
Andreas Lutz
Senior Key Account Manager
Alte Rupminer Allee 74
16816 Neuruppin, Germany
Tel: +49-176-24481915
Email: a.lutz@comforte.com

Helmut Bernhard
Senior Solutions Architect
Steubenstraße 5
65189 Wiesbaden, Germany
Tel.: +49-171-6949285
Email: h.bernhard@comforte.com
THANK YOU
for your attention!