OmniPayments

HP NonStop X
First User
Experience

Yash Kapadia, CEO
Yash@OmniPayments.com

Reasons for upgrade

- Infiniband – Our Big Data and Dashboards run on Linux
- Performance improvements
- Critical for our OmniCloudX offering

First impressions - its beautiful
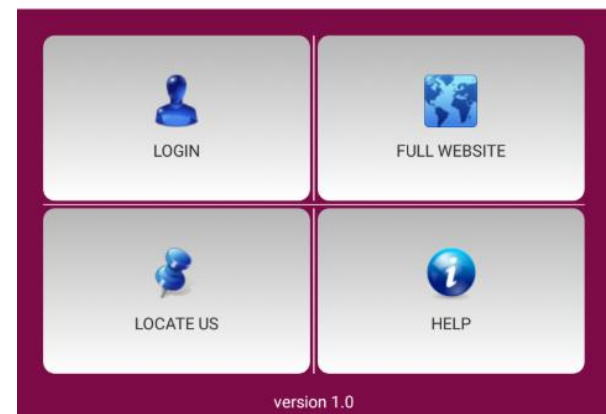
The back

Surprises !!

- TUXEDO is not supported !! HP had been talking about this for many years but we were hopeful

- Performance is fantastic

# Project Scope - OmniPayments Suite

OmniPayments Mobile app

- Connects using SOA
- Full SSL encryption
- Integrated with Google Maps

OmniPayments Services

- Various supported options are displayed
- Accounts – Displays the account details of the customer
- Credit Cards – Displays the credit card details of the customer
- Funds Transfer – Provides fund transfer facility to already registered Payees
- Bill Pay – Provides facility to pay your bills such as electricity, water
- Recharge – Provides facility to recharge a cell phone
- Loyalty – Provides an interface for Loyalty cards and features
- Request – Provides facility to raise requests such as Block Credit Card, Re-issue statement, Stop Check
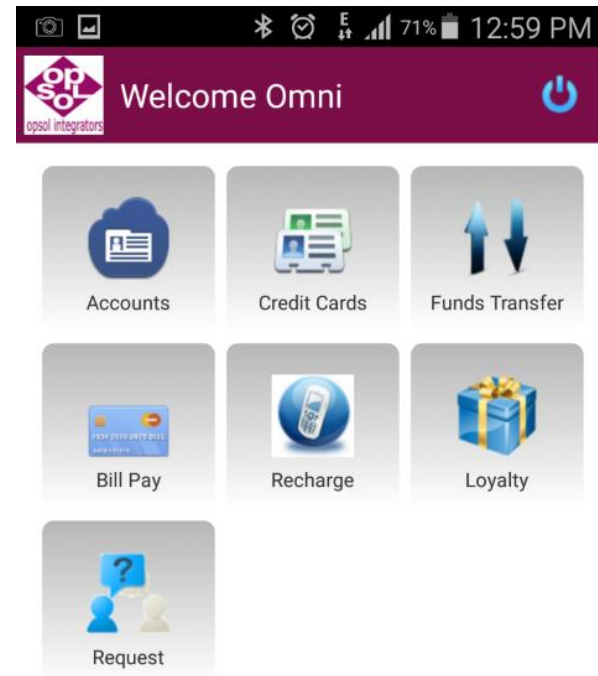
OmniPayments Loyalty Module

- If a customer is enrolled for Loyalty scheme, then this feature is enabled.
- Click on My Account to log into the Loyalty app.
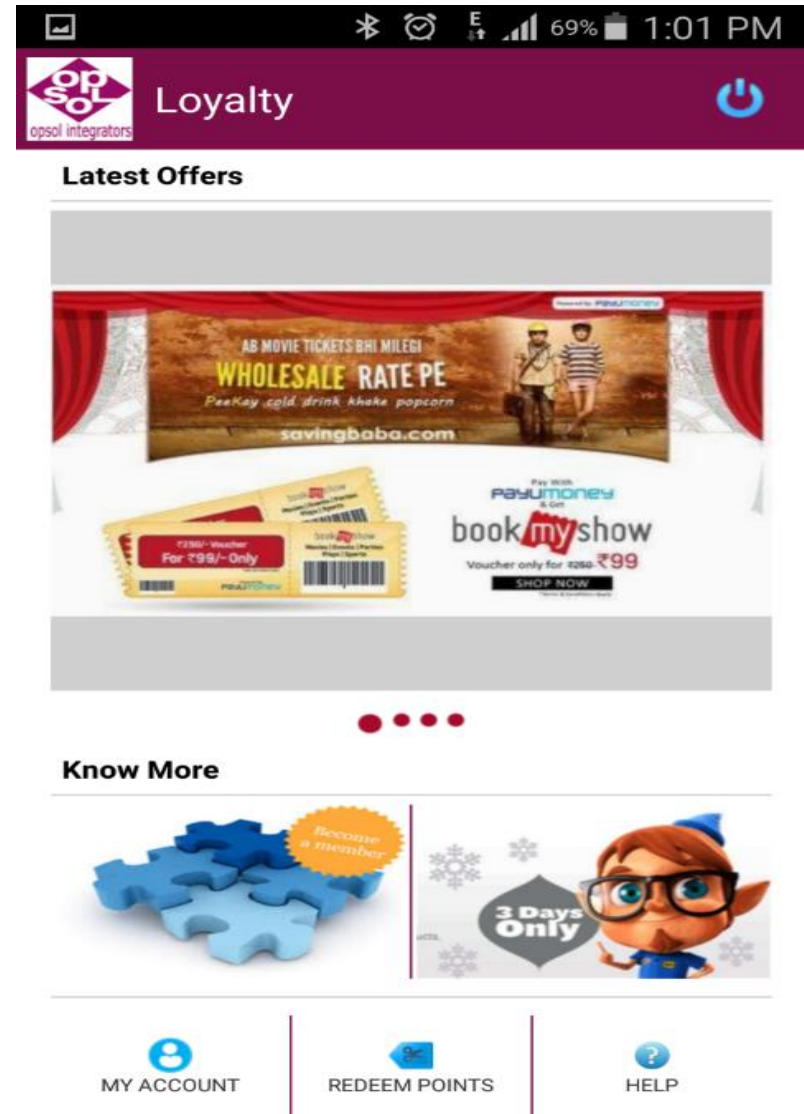
Library Structure

- OmniPayments has created libraries to support its infrastructure. These libraries need to be linked in TD.

- Below are some important libraries used in TD infrastructure.

Header Files	Objs	Libraries
--------------	-------	--------------------------------------

Header Files
----
od_intf.h
authOds.h
authRule.h
authsor.h
authCommon.h
sendMail.h
adapt.h
mime.h
srvcomn.h
opsgenipc.h
iso8583.h
utils.h
feesCommon.h
currencyCommon.h
extractOps.h
feecalc.h

….

Objs
----
biztd.o
od_intf.o
omnirouter.o
genipc.o
bsMsgHandler.o
lic.o

….

Libraries

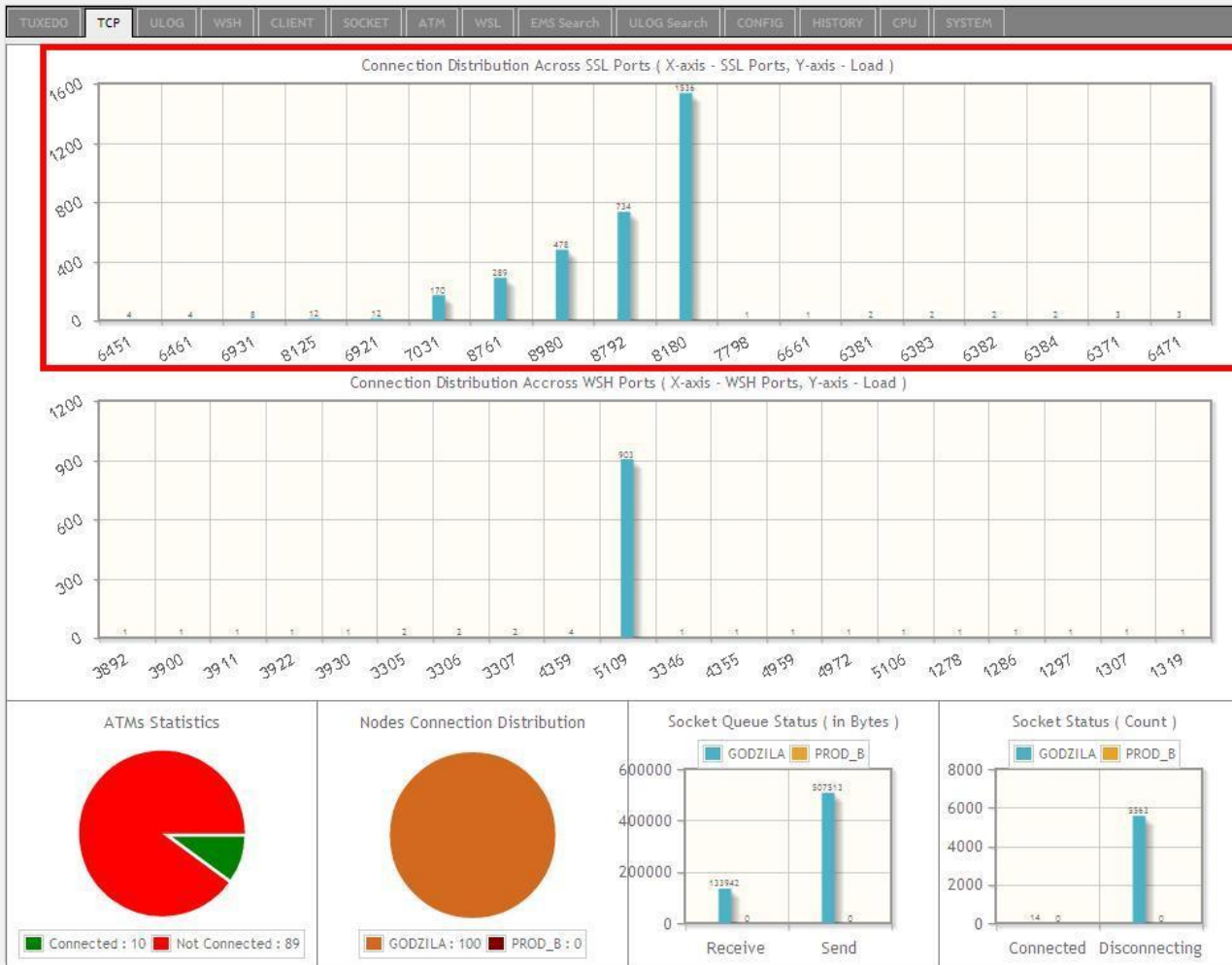libauth.a   liblogproc.a
libauthods.a   libmime.a
libbsdb.a   libmt.a
libcrypto.a   libocintf.a
libcurrency.a   libocomni.a
libdash.a   libocssha.a
                   libodsapilite.a

…..

…..

10

OmniDash TCPIP Dashboard

Steps to build TD user exits

- User needs to use OmniPayments infrastructure libraries in TD compilation.
- Copy all required libraries, objects, header files at common location. e.g.
  1. /build/common/libs
  2. /build/common/objs
  3. /build/common/include
- Create build location and Copy source files, header files, makefile, defines.makefile.
- Point common location and bind required libraries in TD executable compilation
- Create TD executable using below command
  make all

Linker Changes

eld is the TNS/E linker.

xld is the TNS/X linker.

To specify linker options to c89 you should replace all instances of -Weld in your makefiles with -Wxld.

Blades:

c89flags = -Wextensions -Wsystype=oss -Weld="-set floattype Tandem_FLOAT" -Wsaveabend -Wallow_cplusplus_comments -Wverbose -Whighpin=on -Wnowarn=11213 -Weld="-allow_duplicate_procs"

X:

c89flags = -Wextensions -Wsystype=oss -Wxld="-set floattype Tandem_FLOAT" -Wsaveabend -Wallow_cplusplus_comments -Wverbose -Whighpin=on -Wnowarn=11213 -Wxld="-allow_duplicate_procs"

Library changes

Old TNS/E library names like zcpp2dll, zppcdll
New Libraries have names starting with 'x' instead of 'z'

Blades:
SYSSRLS = -l zcpp2dll -l zcppcdll -l c -l C

X:
SYSSRLS = -l xcpp2dll -l xcppcdll -l c -l C

NSDEE Project compilation steps

Difference in H/J series and L series

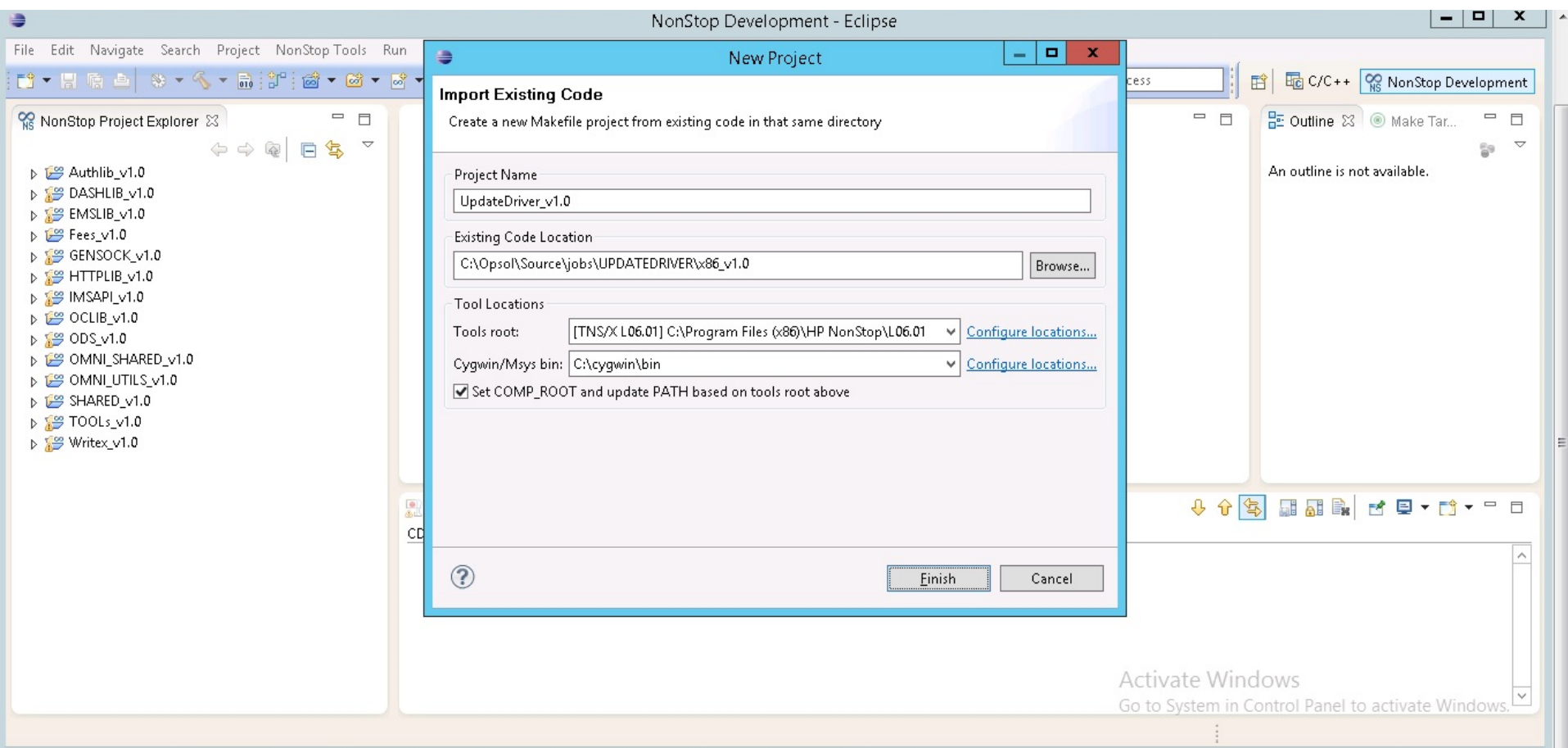| Library	H- and J-Series DLL Name		L-Series DLL Name		Shared DLL Name
	NSK[1]	PC	NSK[1]	PC	Abbreviated Form
COBOL run-time library	ZCOBDLL	libcob.so	XCOBDLL	libcobx.so	-l cob
COBOL arithmetic	Not available: this library is not used on TNS/E systems.	Not available: this library is not used on TNS/E systems.	XCOBADLL	libcobax.so	-l coba[2]
Common Run-Time Environment (CRE)	ZCREDLL (32-bit) YCREDLL (64-bit)	libcre.so (32-bit) libcrey.so (64-bit)	XCREDLL (32-bit) WCREDLL (64-bit)	libcrex.so(32-bit) libcrew.so(64-bit)	-l cre
C run-time library	ZCRTLDLL (32-bit) YCRTLDLL (64-bit)	libcrt.so (32-bit) libcrtly.so (64-bit)	XCRTLDLL (32-bit) WCRTLDLL (64-bit)	libcrtx.so libcrtw.so	-l crtl
C++ VERSION2 standard run-time library	ZCPPCDLL ZCPP2DLL	libcppc.so libcpp2.so	XCPPCDLL XCPP2DLL	libcppcx.so libcpp2x.so	-l cppc -l cpp2
C++ VERSION3 standard run-time library	ZCPPCDLL (32-bit) ZCPP3DLL (32-bit) YCPPCDLL (64-bit) YCPP3DLL (64-bit)	libcppc.so (32-bit) libcpp3.so (32-bit) libcppcy.so (64-bit) libcpp3y.so (64-bit)	XCPPCDLL (32-bit) XCPP3DLL (32-bit) WCPPCDLL (64-bit) WCPP3DLL (64-bit)	libcppcx.so (32-bit) libcpp3x.so (32-bit) libcppcw.so (64-bit) libcpp3w.so (64-bit)	-l cppc -l cpp3

C89 and C99 flags changed

Flag	Reason for Change	Action Required
-Weld	L-series has new linker when the target is TNS/X.	Replace with -Wxld if the target is TNS/X.
-Weld_obey (the corresponding pragma is LINKFILE, which is unchanged but submits commands to xld instead of eld)	L-series has new linker when the target is TNS/X.	Replace with -Wxld_obey if the target is TNS/X.
-Wglobalized	L-series has a new default for TNS/X; -Wglobalized is always turned on.	No action required; -Wglobalized is ignored.
-Wtarget	New supported value: -Wtarget=tns/x to generate x86 code. No longer supported value: -Wtarget=tns/r because TNS/R is not supported on L-series.	No action required. Default target platform is TNS/X.

NSDEE Project compilation steps
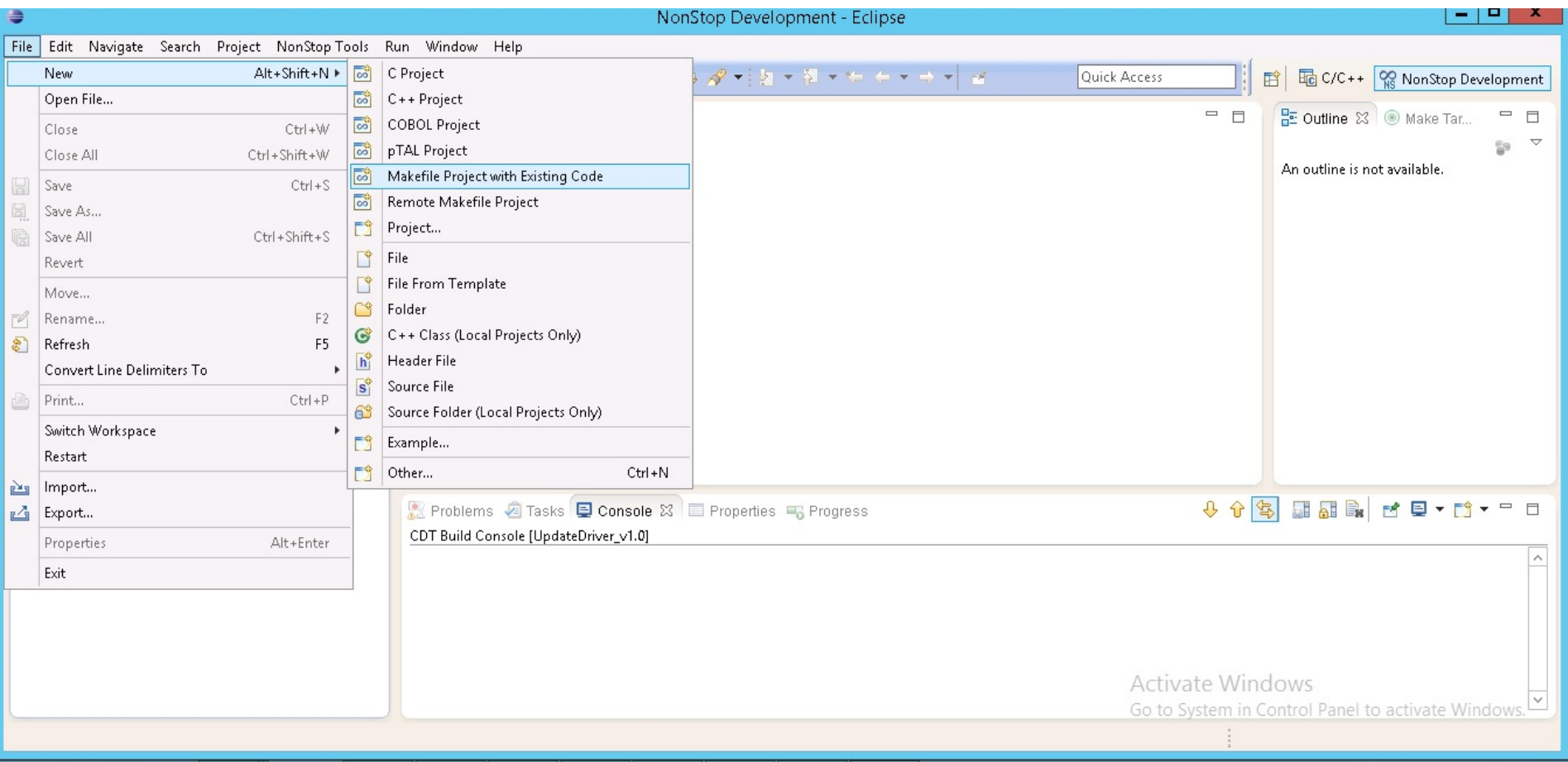
Notice the X Directory locations

Build in progress

Tuxedo replacement using HP NONSTOP REAL TIME INFORMATION DIRECTOR SW – Part Number BE348AC

Custom

Orchestration

tpcall input buffer

FML to C Data Transformation

TUXEDO Reply buffer

XML preparation

SOA Service Invocation

XML to C

mapping

CORE SERVICES

Routing

Orchestration

Security

Load Distribution

Failover/Retry

Multithreading

INFRASTRUCTURE SERVICES

OmniConsole	OmniOffender
OmniLogger	OmniDash
OmniScheduler	
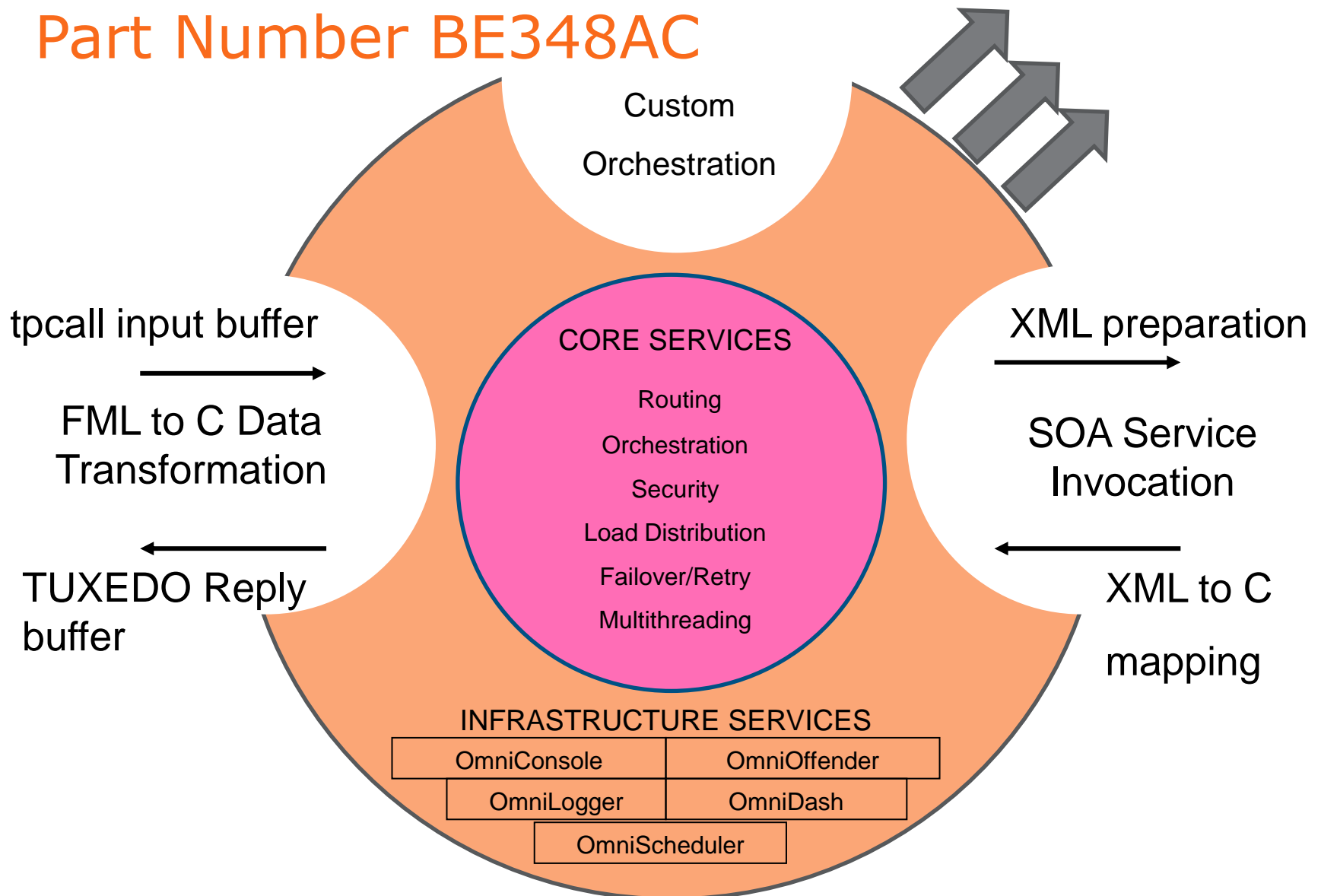
Our observations

- HP Installation team was outstanding
- Very easy migration – takes time – but easy
- TUXEDO not supported – caused us rework – we replaced using HP NONSTOP REAL TIME INFORMATION DIRECTOR SW – Part Number BE348AC
- Performance is fantastic

Next steps

- Test Infiniband for Hybrid configuration
- Upgrade OmniDash using Infiniband
- Upgrade our Big Data using Infiniband
- Run volume tests