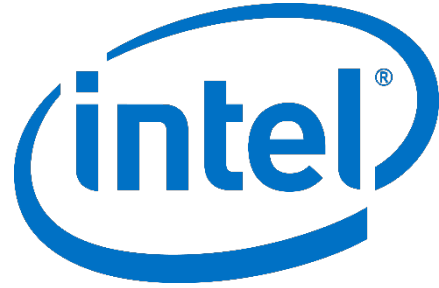




**Hewlett Packard  
Enterprise**



# **NSDEE Example**

## Development Workflow Example

Cor Geboers, ATC

---

# About me

## Cor Geboers

Senior Consultant in NonStop ATC, based in Belgium

35+ years in IT development and support

25+ years NonStop experience

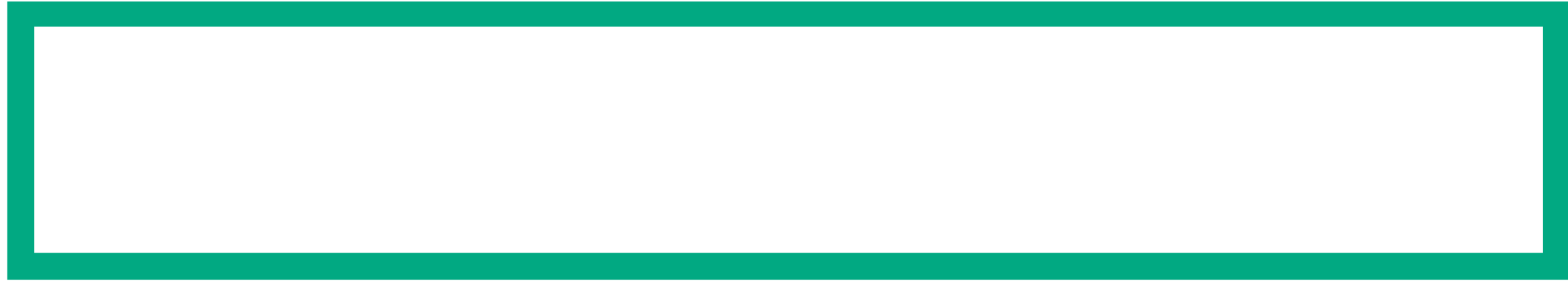
Working with middleware and new technologies

OSS adept since the beginning

---

# Agenda

- Eclipse and CDT
- What is NSDEE?
- Using version control tools
- Continuous Integration example



# Eclipse Overview

---

# Eclipse — Open Source Development Platform

- History:

- Originally developed by IBM.
- Managed since 2004 by the Eclipse Foundation
  - An independent not-for-profit corporation

- Supports:

- The construction of a variety of tools and programming languages for application development.
- Tools to manipulate arbitrary content types (e.g., HTML, Java, C, JSP, EJB, XML, and GIF).
- Seamless integration of tools within and across different content types and tool providers.

- Runs on a wide range of operating systems, including Windows®, Linux™, Mac OS X, Solaris AIX, and HP-UX.

---

# What is the CDT

- C/C++ Development Tooling
- Used to build C/C++ applications
- Extensible platform
- Features:
  - Project and Build
  - Editor
  - Source code navigation and refactoring
  - Parser and Indexer
  - Debugger



# NonStop Development Environment for Eclipse

---

# HP NonStop Development Environment for Eclipse

NSDEE:

- Adds:
  - Platform-specific enhancements to the open-source Eclipse IDE
- Preserves:
  - The ease of PC-based software development
  - The extensibility of the Eclipse IDE
  - A common platform for developing for multiple systems
- Increases:
  - productivity of NonStop developers
- Lowers
  - The cost for maintenance and training of developers



---

## Key Features of NSDEE (1 of 4)

- A seamless integration with the open-source Eclipse development environment.
- Single development environment for
  - Native C/C++, Native COBOL, pTAL, and Java languages.
  - Integrity NonStop systems
  - NonStop S-series systems.
- Support for project types specific to NonStop servers.

---

## Key Features of NSDEE (2 of 4)

- Support for databases:
  - HP NonStop SQL/MP and HP NonStop SQL/MX.
- An integrated debugger
  - Java
  - Cobol, C/C++ (optional NSDEE Debugger product required)
- The ability to use the same tools and skills across platforms.
- A functional alternative to Enterprise Toolkit (ETK).
  - Tool to migrate ETK-generated NonStop projects to Eclipse.

---

## Key Features of NSDEE (3 of 4)

### –Local Projects

- Build (compile/link) performed on workstation
- Deploy executable to NonStop
- Launch application and view results from Eclipse
- NSDEE debugger (Integrity NonStop systems)

### –Remote Projects

- Build (compile/link) performed on NonStop
- Integrity NonStop systems only
- Secure connection support
- All source and object files stored on NonStop
  - Files copied to workstation for editing
  - Simultaneous updates detection

---

## Key Features of NSDEE (4 of 4)

- NSDEE debugger
- Complementary tools:
  - File transfer.
  - Visual Inspect.
- Integrated help system.
- Sample Programs

---

# Eclipse and NSDEE Debugger

- Integral part of Eclipse
- Natively supports Java (covered in Java module)
- Extended by NSDEE
  - TNS/E and TNS/X only
    - C/C++ and COBOL only
    - Use Visual Inspect for TNS
- Facilities
  - Breakpoints
  - Execution control
  - Variables display and modification
  - Display output on the console view
  - Can attach to running process
  - Snapshot/Save-abend file analysis

# Editing Source Files

The screenshot displays the Eclipse IDE interface. The main editor window shows the source code for 'helloserver.c'. The code includes headers for `stdio.h`, `string.h`, and `stdlib.h`. It defines a `debug` integer, a `range` integer, a `sourceChars` character array, a `scLen` integer, and several functions: `process_message`, `getRandomAlfa`, `gen_random`, `randomString`, `convertToUpper`, and `copyToUpper`. The `main` function starts by printing 'Starting PWSERVER01\n', opens a file named '\$RECEIVE', and enters a loop to process requests. In the loop, it reads data into a buffer and checks for a system message. If a system message is received (specifically -103), it copies the first two bytes of the buffer into a `sys_msg` variable and increments an `open count`.

```
46     fprintf(stderr, "Starting PWSERVER01\n");
47
48     /* Open $receive */
49     error = FILE_OPEN("$RECEIVE", (short) strlen("$RECEIVE"), &recvfile, 0, 0, 0, 1);
50     if (recvfile == -1)
51     {
52         fprintf(stderr, "Error opening $RECEIVE: %d\n", error);
53         exit(1);
54     }
55
56     /* seed the random generator */
57     srand(time(NULL));
58
59     /* Process requests */
60     for (;;)
61     {
62         READUPDATEX(recvfile, (char *) reqbuf, (short) sizeof(reqbuf), &count_transferred);
63         error = FILE_GETINFO(recvfile, &last_error);
64         FILE_GETRECEIVEINFO((short *) &receiveinfo);
65         if (last_error == 6)
66         { /* system message */
67             short sys_msg;
68             memcpy((char *) &sys_msg, reqbuf, 2);
69             switch (sys_msg)
70             {
71                 case -103:
72                     open count++;
```

# Debugging Programs

The screenshot displays the Eclipse IDE interface during a debugging session. The top toolbar includes standard IDE functions like File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The left sidebar shows the project structure with 'pwclient01' selected. The main editor window shows the source code for 'pwclient01.c', with line 92 highlighted: `return error;`. The right sidebar contains the 'Variables' view, which lists local variables: `error` (short, 0), `e1` (short, 902), and `e2` (short, 14). Below the variables is the 'Outline' view showing the project's file structure and function definitions. The bottom console window displays the output: `Error during SERVERCLASS_SEND_: 902 - 14`. The status bar at the bottom indicates '184M of 265M' memory usage and 'Launching pwclient01: (57%)'.

Name	Type	Value	Location
(x)= error	short	0	0x6fffeed8
(x)= e1	short	902	0x6fffeed6
(x)= e2	short	14	0x6fffeed4

```
84 {
85     error = SERVERCLASS_SEND_(pathmon, (short) strlen(pathmon), serverclass, (short) strlen(serverclass), (char *) buffe
86         (short) zinput_def_size, (short) zoutput_def_size, &reply_len);
87
88     if (error)
89     {
90         error = SERVERCLASS_SEND_INFO(&e1, &e2);
91         printf("Error during SERVERCLASS_SEND_: %d - %d\n", e1, e2);
92         return error;
93     }
94
95     reply = (zoutput_def *) buffer;
96     /* Do something with the response */
```



# NSDEE Deployment Workflow



# Using GIT

nsdee8a - pw-example/pwclient01.c - Eclipse

File Edit Source Refactor Navigate Search Project NonStop Tools Run Window Help

Quick Access

NonStop Projec...  
dbmetadata  
parking  
pw-example [pw-example ma]  
Binaries  
Includes  
scripts  
heloserver.c  
mydefs.h  
pwclient01.c  
pwclient01 - [x86\_64/be]  
pwserver01 - [x86\_64/be]  
Makefile  
README.md  
pwserver01 [pwserver01 ma]

```
75     exit(1);  
76 }  
77  
78 input = (zinput_def *) buffer;  
79  
80 input->reqcode = req_code;  
81 strncpy(input->message, message, strlen(message));  
82  
83 for (x = 0; x < count; x++)  
84 {  
85     error = SERVERCLASS_SEND_(pathmon, (short) strlen(pathmon), serverclass, (short) strlen(s  
86         (short) zinput_def_size, (short) zoutput_def_size, &reply_len);
```

Outline  
stdio.h  
stdlib.h  
unistd.h  
string.h  
cextdecs.h  
mydefs.h  
sum(int):int  
optarg:char\*  
entire:int

Problems Tasks Console Properties Git Staging Filter files

> pw-example [master]

Unstaged Changes (0)

Staged Changes (7)

- testing.sql - pw-example/scripts
- .cproject - pw-example
- create\_pathmon - pw-example/scripts
- create\_pwserver01 - pw-example/scripts
- denlov2domain - pw-example/scripts

Commit Message

⚠ Second line should be empty to separate commit message header from body

Adding support for different systems in the shell scripts to create pathmons

Author: Cor Geboers <cor.geboers@hpe.com>

Committer: Cor Geboers <cor.geboers@hpe.com>

Commit and Push... Commit

166M of 265M Launching pwclient01: (57%)

---

# Trigger webhook in Gitlab

```
Apr 13, 2018 2:59:49 AM org.eclipse.jetty.server.session.Session beginInvalidate
INFO: Session node0hsxz1h4noyp01bzuk2ruopzer0 already being invalidated
Apr 13, 2018 3:00:59 AM com.dabsquared.gitlabjenkins.webhook.GitLabWebHook getDynamic
INFO: WebHook called with url: /project/GetSources
Apr 13, 2018 3:01:00 AM com.dabsquared.gitlabjenkins.trigger.handler.AbstractWebHookTriggerHandler handle
INFO: GetSources triggered for push.
Apr 13, 2018 3:01:16 AM hudson.model.Run execute
INFO: GetSources #5 main build action completed: SUCCESS
```

# Jenkins Job executed

The screenshot displays the Jenkins web interface for a job named 'GetSources'. The browser address bar shows the URL 'nsx09:23080/job/GetSources/lastBuild/'. The Jenkins header includes the user 'Cor Geboers' and a 'log out' link. The main content area shows the details for 'Build #17 (08-May-2018 05:15:54)', which started 40 seconds ago and took 4.8 seconds. The build was initiated by a 'GitLab push by Cor Geboers'. The console output indicates 'No changes.' and provides the Git revision '5eb800a077721cb61bc7740dd5d5ea2202ee1826' with the branch 'origin/master'. A sidebar on the left offers navigation options like 'Back to Project', 'Status', 'Changes', 'Console Output', 'Edit Build Information', 'Delete Build', 'Polling Log', 'Git Build Data', 'No Tags', and 'Previous Build'. The footer of the page states 'Page generated: 08-May-2018 05:16:34 MDT' and includes links for 'REST API' and 'Jenkins ver. 2.107.2'.

---

# Console Output

```
Started by GitLab push by Cor Geboers
Building in workspace /home/hp/corg/dev/nsdee8/jenkins/workspace/GetSources
[WS-CLEANUP] Deleting project workspace...
[WS-CLEANUP] Done
Cloning the remote Git repository
remote: Counting objects
remote: Compressing objects
Receiving objects
Resolving deltas
Updating references
Checking out Revision 5eb800a077721cb61bc7740dd5d5ea2202ee1826 (origin/master)
Commit message: "Changed debug messages to '1'"
[GetSources] $ /bin/sh -xe /tmp/jenkins8369977841141642740.sh
+ cd /home/hp/corg/dev/nsdee8/jenkins/workspace/GetSources/pw-example
+ make all
c99 -g -O0 -Wextensions -Wsystype=oss -Wtarget=tns/x -Wenv=common -D__TANDEM -o pwserver01 helloserver.c
c99 -g -O0 -Wextensions -Wsystype=oss -Wtarget=tns/x -Wenv=common -D__TANDEM -o pwclient01 pwclient01.c
+ gtacl -p pdmcom
+ 0< /tmp/sh872415313.1
Finished: SUCCESS
```

---

# Console Output

```
PDMI - T0845L01 - (17AUG2017) - 08 May 2018, 05:15:59
(C) Copyright 2008-2016 Hewlett Packard Enterprise Development LP.
PDMI 1>>
open %PM00

PATHMON : \NSX09.$CG06

PATHMON : \NSX09.$CG07
PDMI 2>>
freeze server pwserver01

stop server pwserver01

+ /usr/coreutils/bin/cp -v pwserver01 /home/hp/corg/dev/nsdee8/bin
'pwserver01' -> '/home/hp/corg/dev/nsdee8/bin/pwserver01'
+ gtacl -p pdmcom
+ 0< /tmp/sh872415313.2
```

---

# Console Output

```
PDMI - T0845L01 - (17AUG2017) - 08 May 2018, 05:15:59
(C) Copyright 2008-2016 Hewlett Packard Enterprise Development LP.
PDMI 1>>
open %PM00

PATHMON : \NSX09.$CG06

PATHMON : \NSX09.$CG07
PDMI 2>>
freeze server pwserver01

stop server pwserver01

+ /usr/coreutils/bin/cp -v pwserver01 /home/hp/corg/dev/nsdee8/bin
'pwserver01' -> '/home/hp/corg/dev/nsdee8/bin/pwserver01'
+ gtacl -p pdmcom
+ 0< /tmp/sh872415313.2
```

---

# Console Output

PDMI 1>>

open %PM00

thaw server pserver01

start server pserver01

PATHMON : \NSX09.\$CG06

\$X28A: SERVER PWSERVER01, STARTED

PATHMON : \NSX09.\$CG07

\$X28A: SERVER PWSERVER01, STARTED

PDMI 4>>

Finished: SUCCESS



**Hewlett Packard**  
Enterprise

**Thank you**