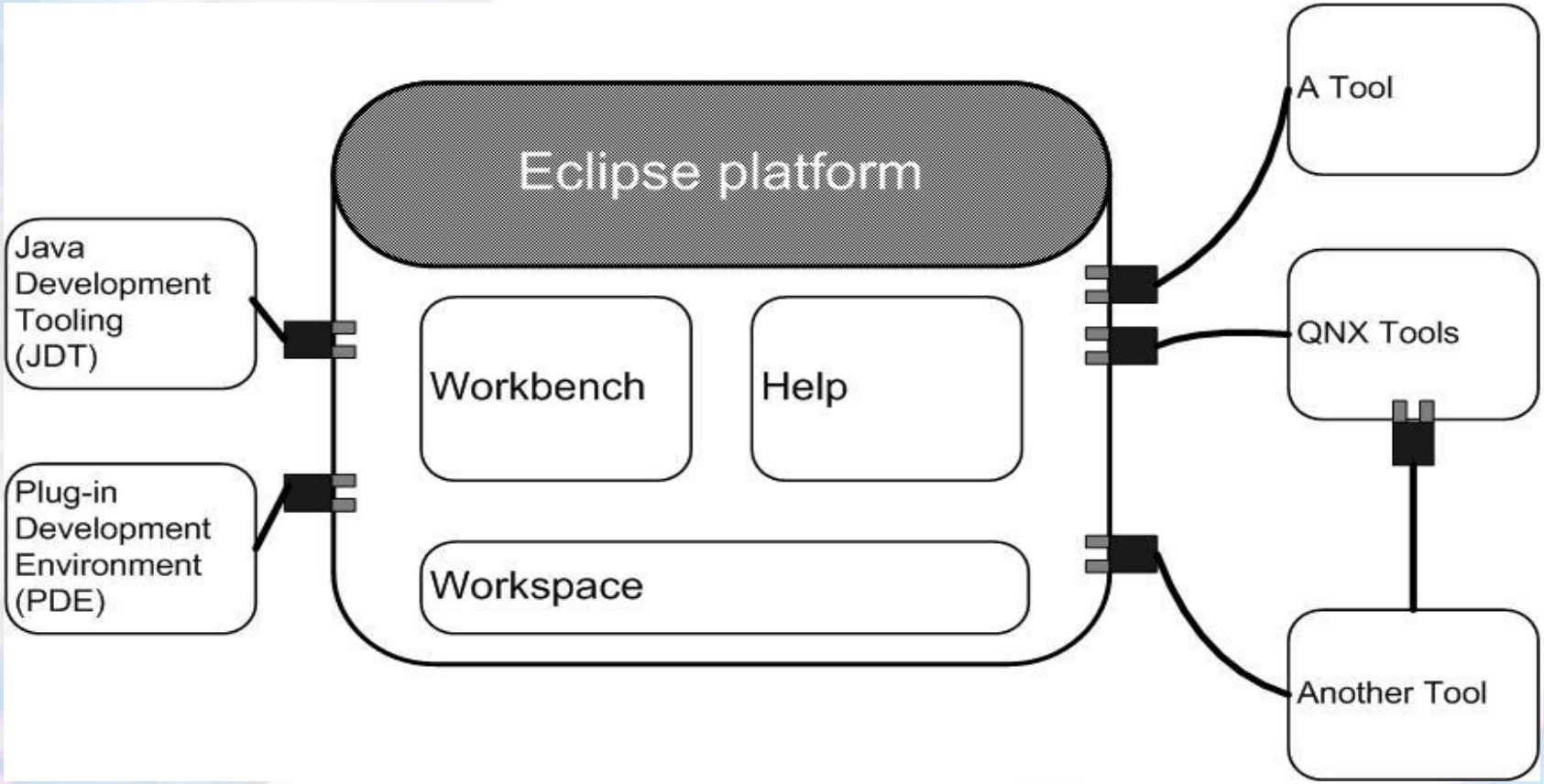
IDE QNX – практическая работа

Трофимов Александр
SWD Software

Архитектура Eclipse



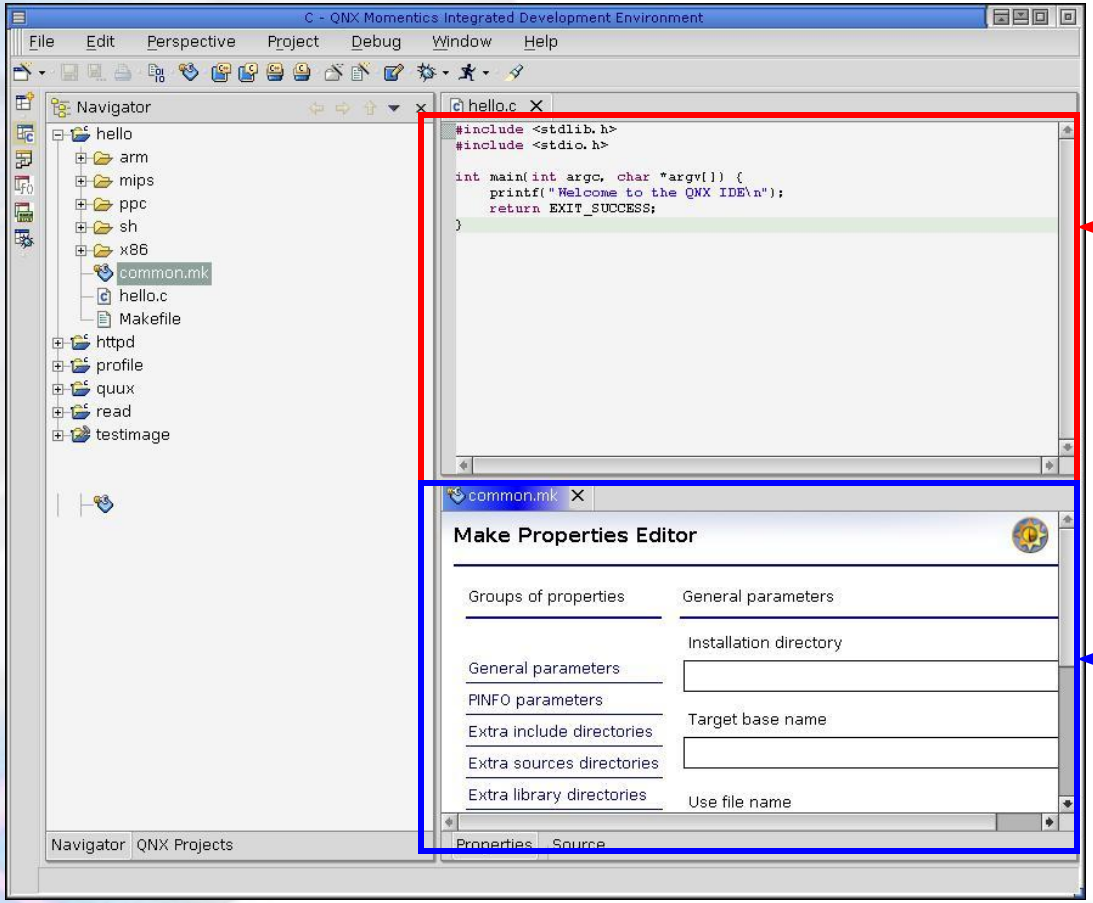
Основные возможности

- Организация ресурсов (проекты, папки, файлы)
- Редактирование ресурсов
- Организация работы над проектами в команде
- Компиляция, запуск и отладка программ
- Наблюдение и анализ работы системы
- Построение загрузочных образов для встраиваемых систем

Фундаментальные концепции

- ***perspectives*** – зависящий от конкретной задачи набор ***view'***ов и ***editor'***ов.
- ***views*** – специальный инструмент для просмотра и работы над ресурсами, обеспечивающий организацию информации и навигацию по ней, а также некоторую поддержку для ***editor'***ов.
- ***editors -*** визуальные компоненты для редактирования и просмотра ресурсов (например, файл с исходным C++ кодом).

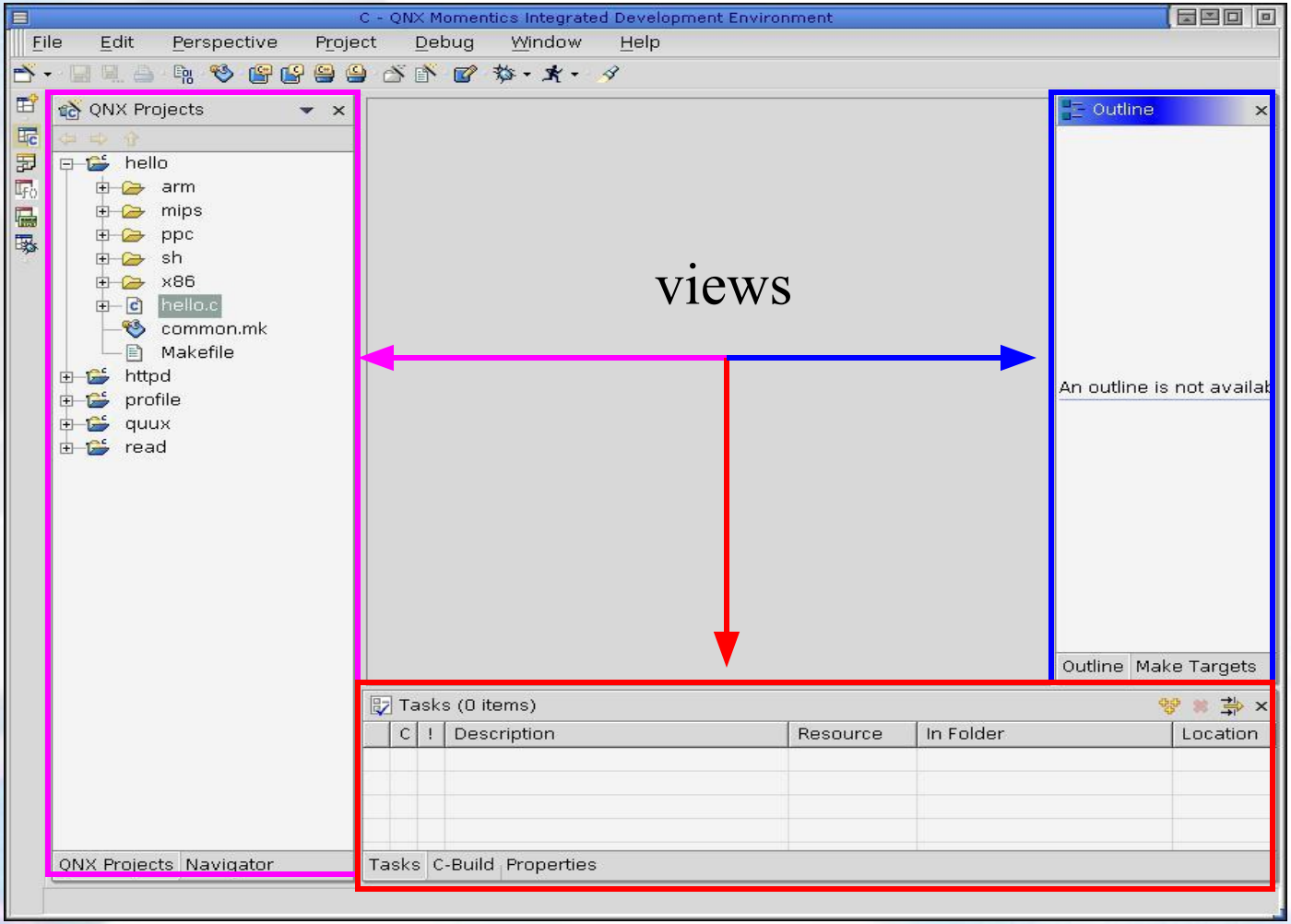
Пример ***editor***



C editor

Make Properties Editor

Пример *view*


C - QNX Momentics Integrated Development Environment
File
Edit
Perspective
Project
Debug
Window
Help
QNX Projects
hello
arm
mips
ppc
sh
x86
hello.c
common.mk
Makefile
httpd
profile
quux
read
QNX Projects
Navigator
views
Outline
An outline is not availab
Outline
Make Targets
Tasks (0 items)
C
!
Description
Resource
In Folder
Location
Tasks
C-Build
Properties

Целевая система

Резидентная среда разработки:

PC под управлением QNX Neutrino

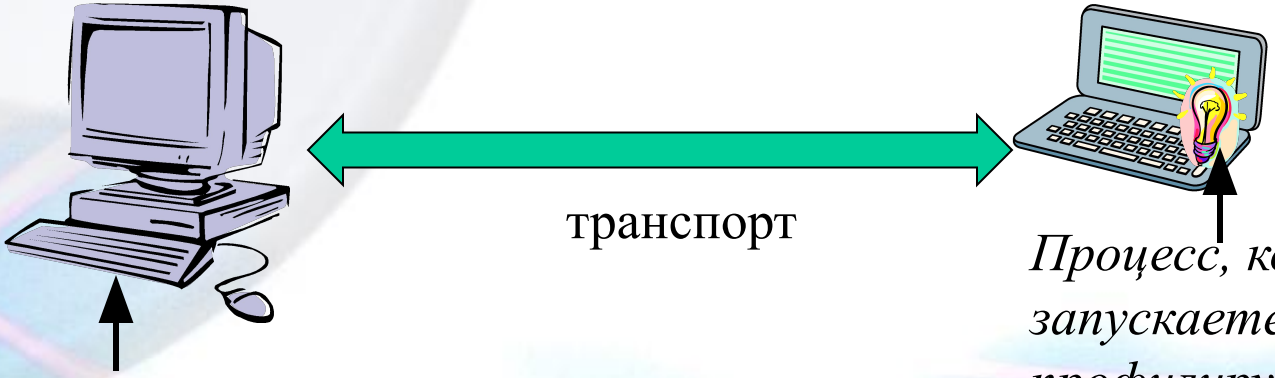
QNX Momentics IDE

Процесс, который вы запускаете, отлаживаете, профилируете

Кросс разработка:

Инструментальная система под управлением Windows/Solaris/Neutrino

Целевая система под управлением QNX Neutrino



транспорт

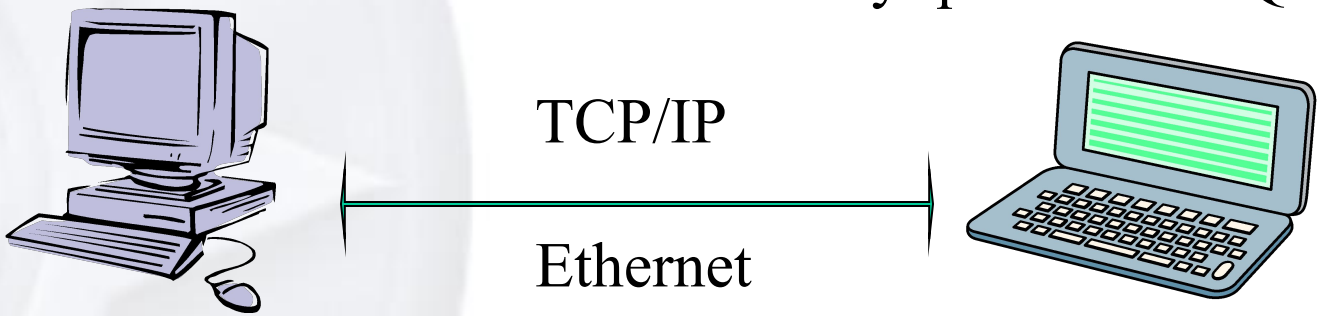
Процесс, который вы запускаете, отлаживаете, профилируете

QNX Momentics IDE

Что для этого необходимо

Инструментальная система с IDE

Целевая система под управлением QNX Neutrino

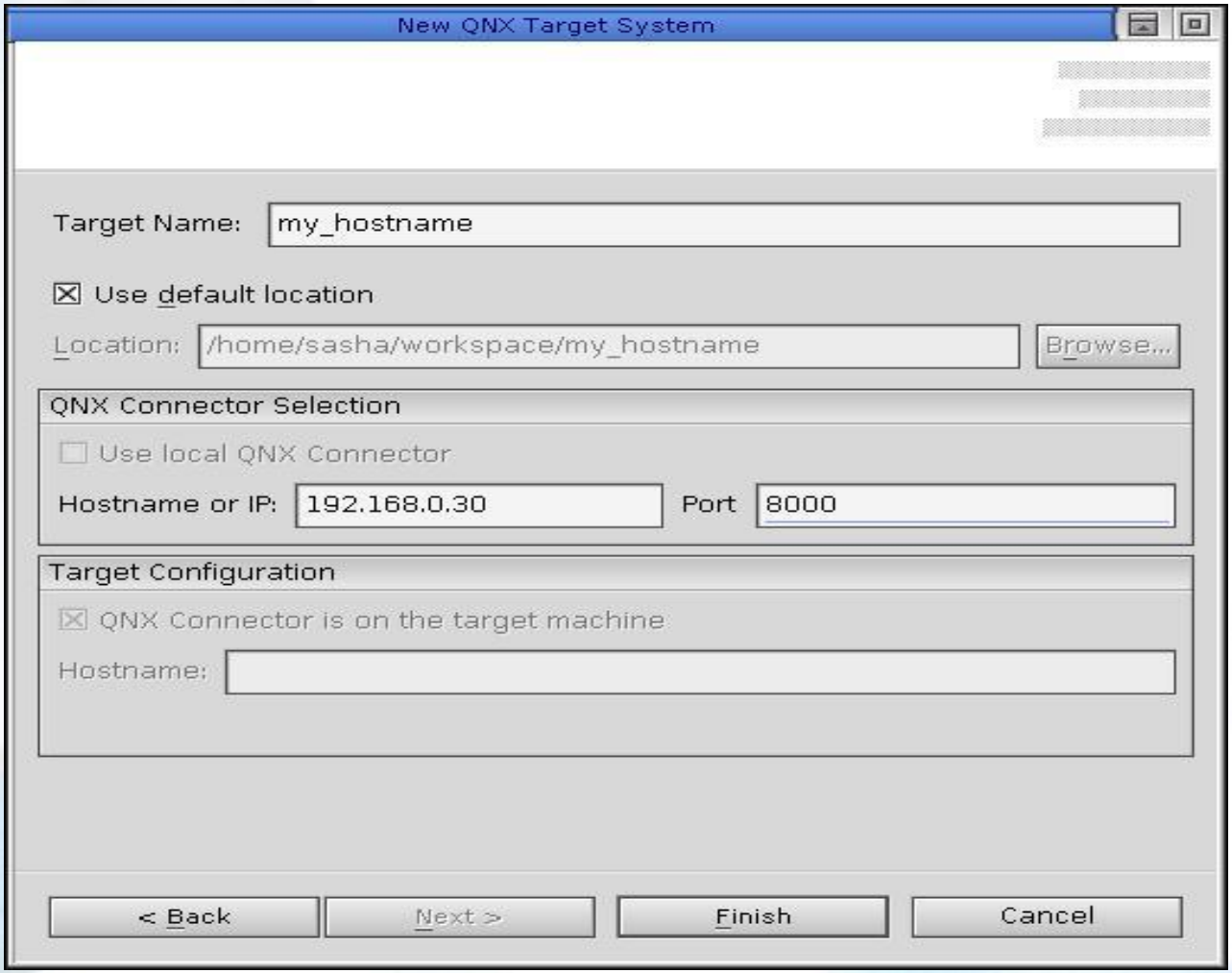


- io-net
- devc-pty
- qconn
- pdebug
- *ваша_программа*

Что может быть целевой системой

- ARM/Xscale
- MIPS
- PowerPC
- SH-4
- x86

Настройка целевой системы в IDE



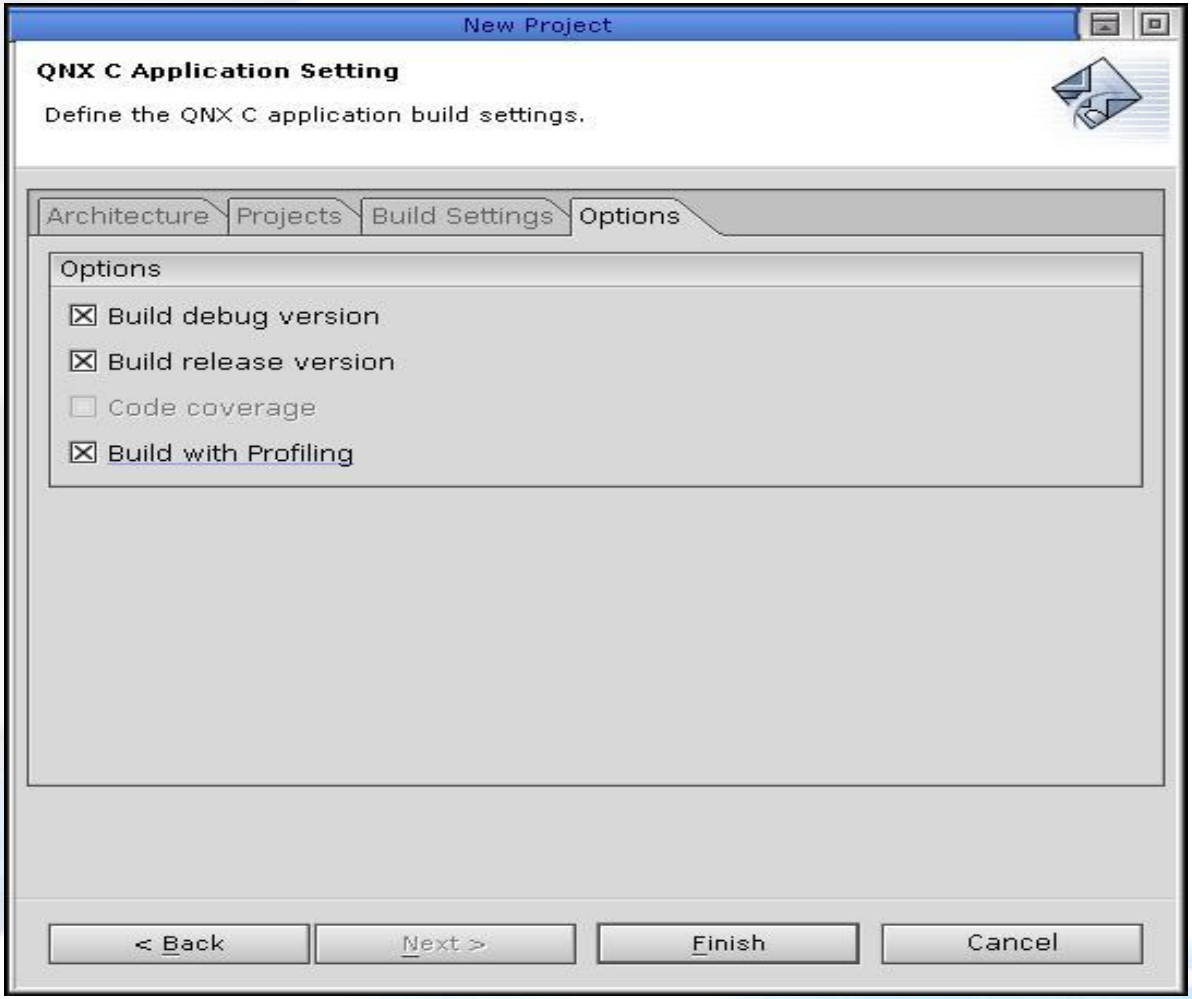
Что такое проект?

В IDE проект – это набор связанных ресурсов (например, каталогов или файлов) для управления вашими программами.

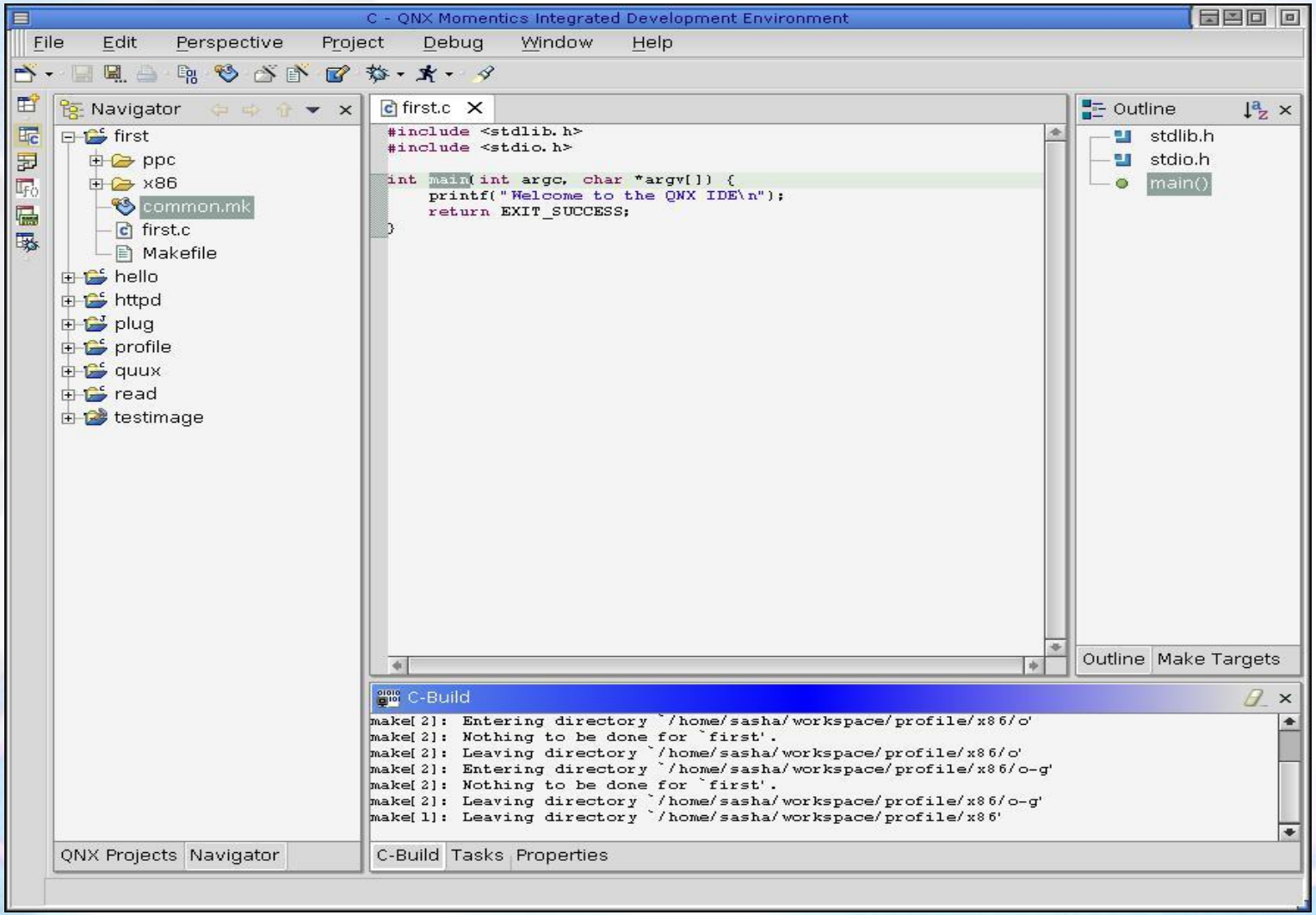
Большинство того, что делается в IDE – проектно-ориентированно – построение проектов, контроль версий, совместная работа над проектами.

Разработка программ

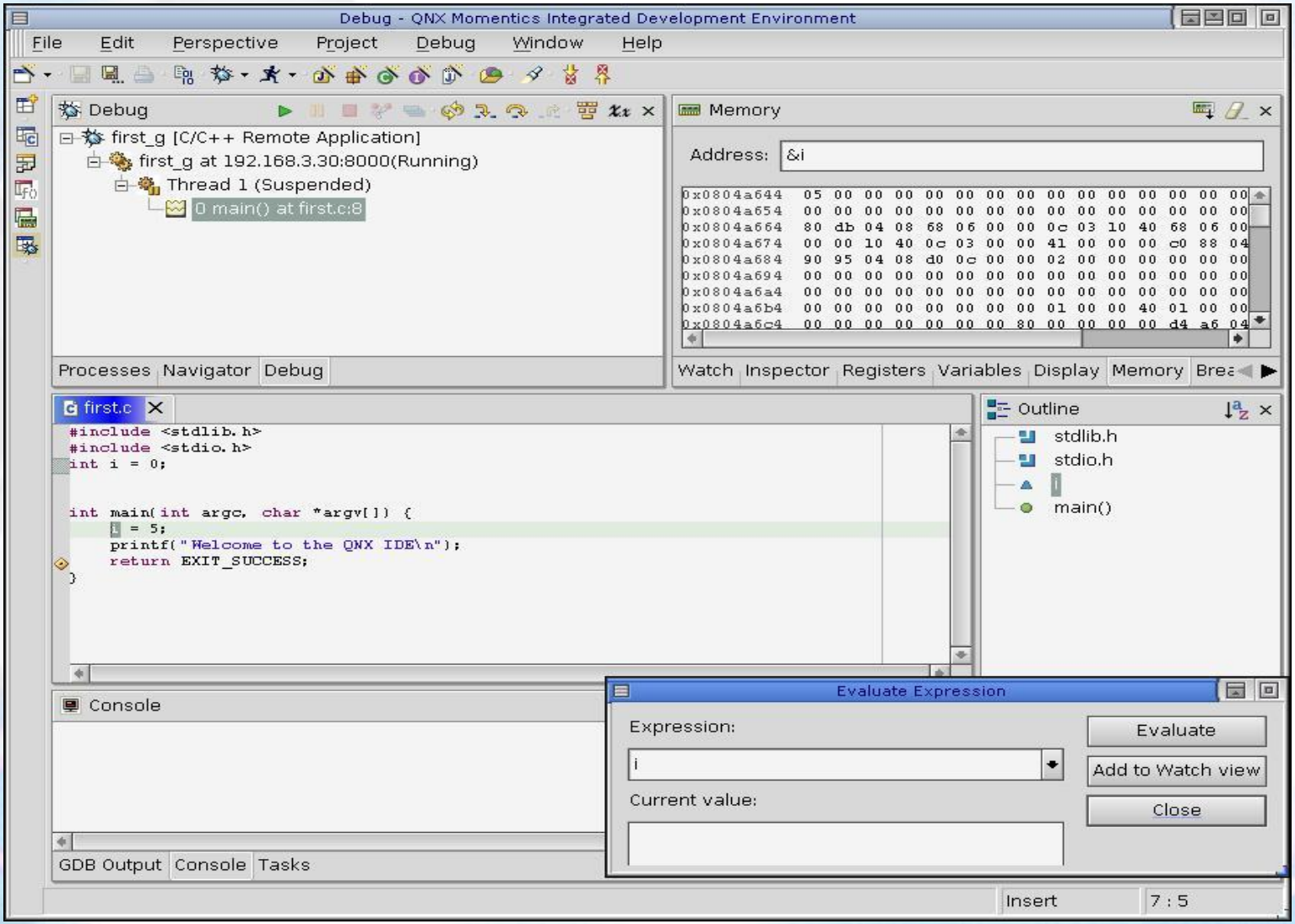
Создание QNX C/C++ проекта



C Perspective



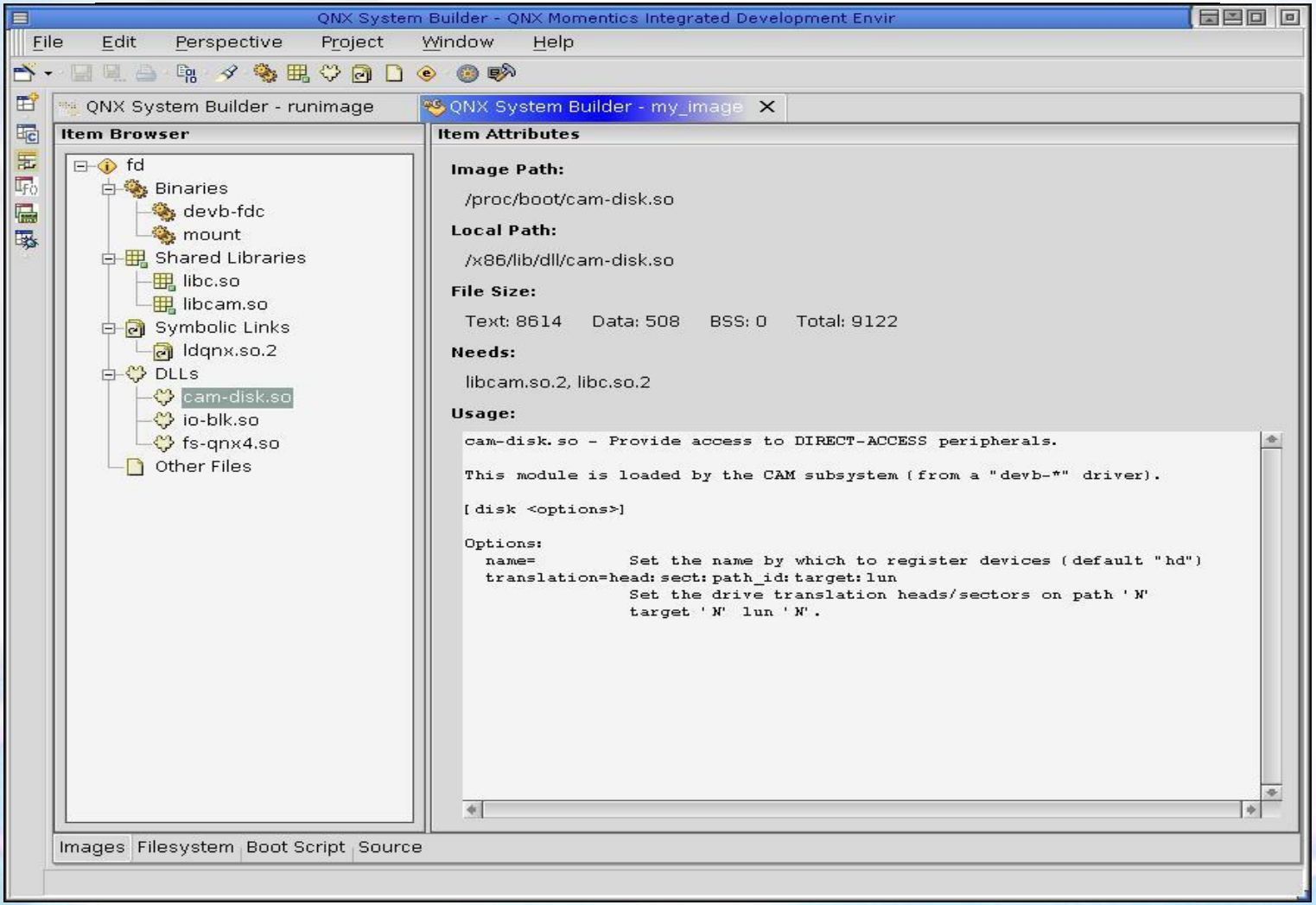
Отладка


Debug - QNX Momentics Integrated Development Environment
File Edit Perspective Project Debug Window Help
Debug
first_g [C/C++ Remote Application]
first_g at 192.168.3.30:8000(Running)
Thread 1 (Suspended)
0 main() at first.c:8
Processes Navigator Debug
Memory
Address: &i
0x0804a644 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0804a654 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0804a664 80 db 04 08 68 06 00 00 0c 03 10 40 68 06 00 00
0x0804a674 00 00 10 40 0c 03 00 00 41 00 00 00 c0 88 04
0x0804a684 90 95 04 08 d0 0c 00 00 02 00 00 00 00 00 00 00
0x0804a694 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0804a6a4 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0804a6b4 00 00 00 00 00 00 00 00 01 00 00 40 01 00 00
0x0804a6c4 00 00 00 00 00 00 00 80 00 00 00 00 d4 a6 04
Watch Inspector Registers Variables Display Memory
first.c
#include <stdlib.h>
#include <stdio.h>
int i = 0;
int main(int argc, char *argv[]) {
i = 5;
printf("Welcome to the QNX IDE\n");
return EXIT_SUCCESS;
}
Outline
stdlib.h
stdio.h
i
main()
Console
GDB Output Console Tasks
Evaluate Expression
Expression:
i
Current value:
Evaluate
Add to Watch view
Close
Insert
7 : 5

System Builder

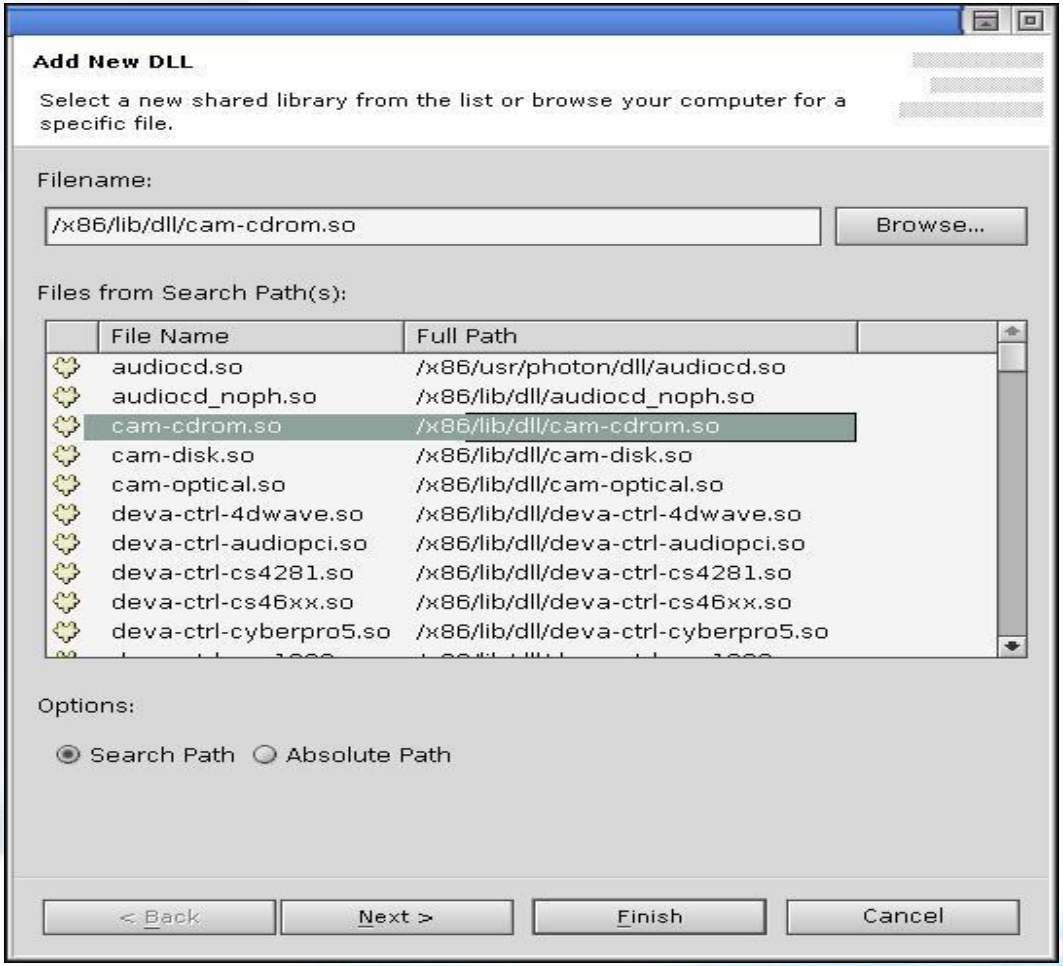
- Построение загрузочных образов системы
- Построение образов ПЗУ
- Оптимизация образа
- Разрешение зависимостей
- Взаимодействие с ROM monitor или QNX IPL на целевой машине для загрузки образа
- Содержит TFTP сервер для загрузки целевых машин, умеющих загружаться по сети с использованием TFTP протокола

System Builder



System Builder

Добавление модулей в образ

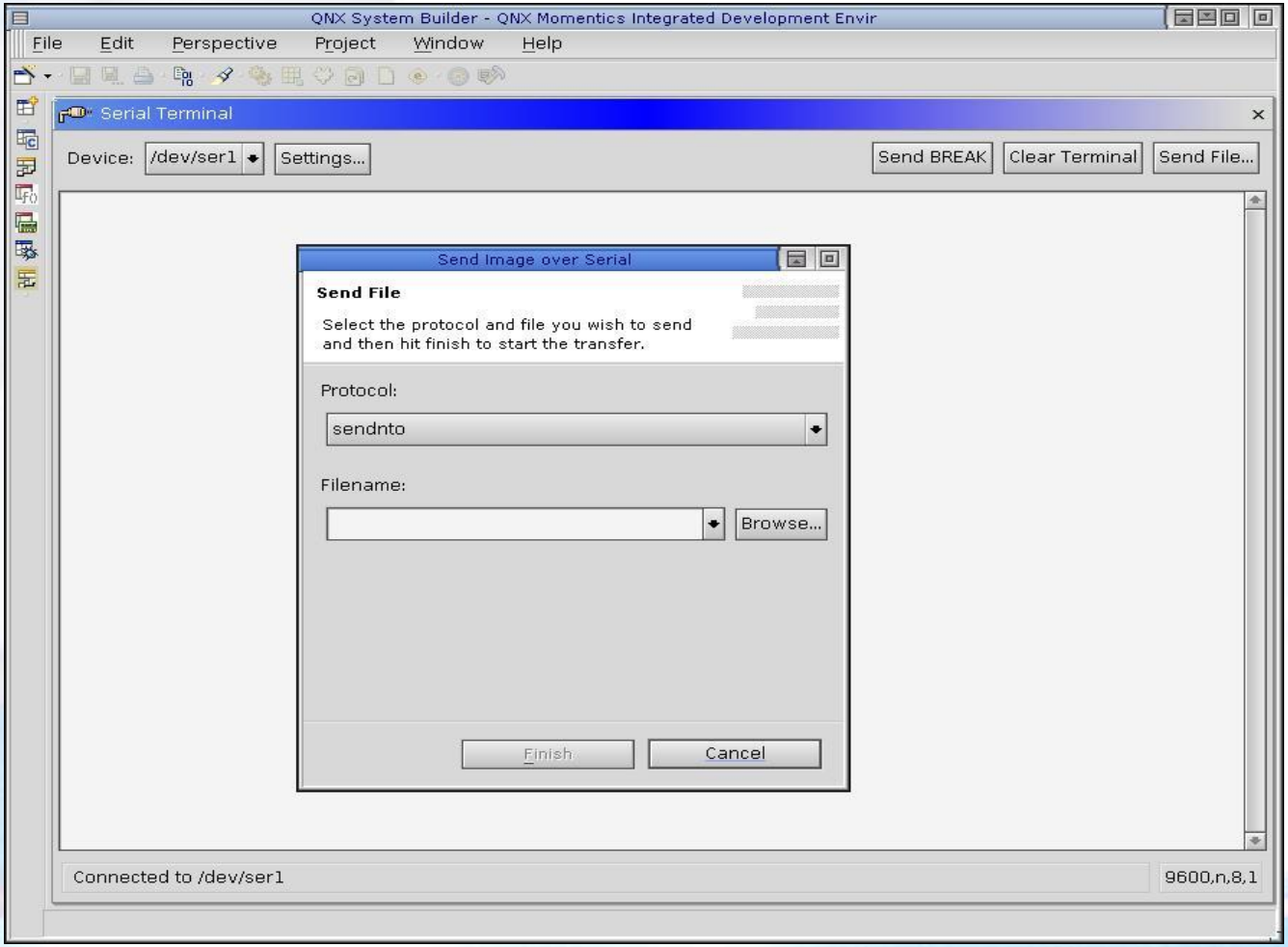


System Builder

Оптимизация образа

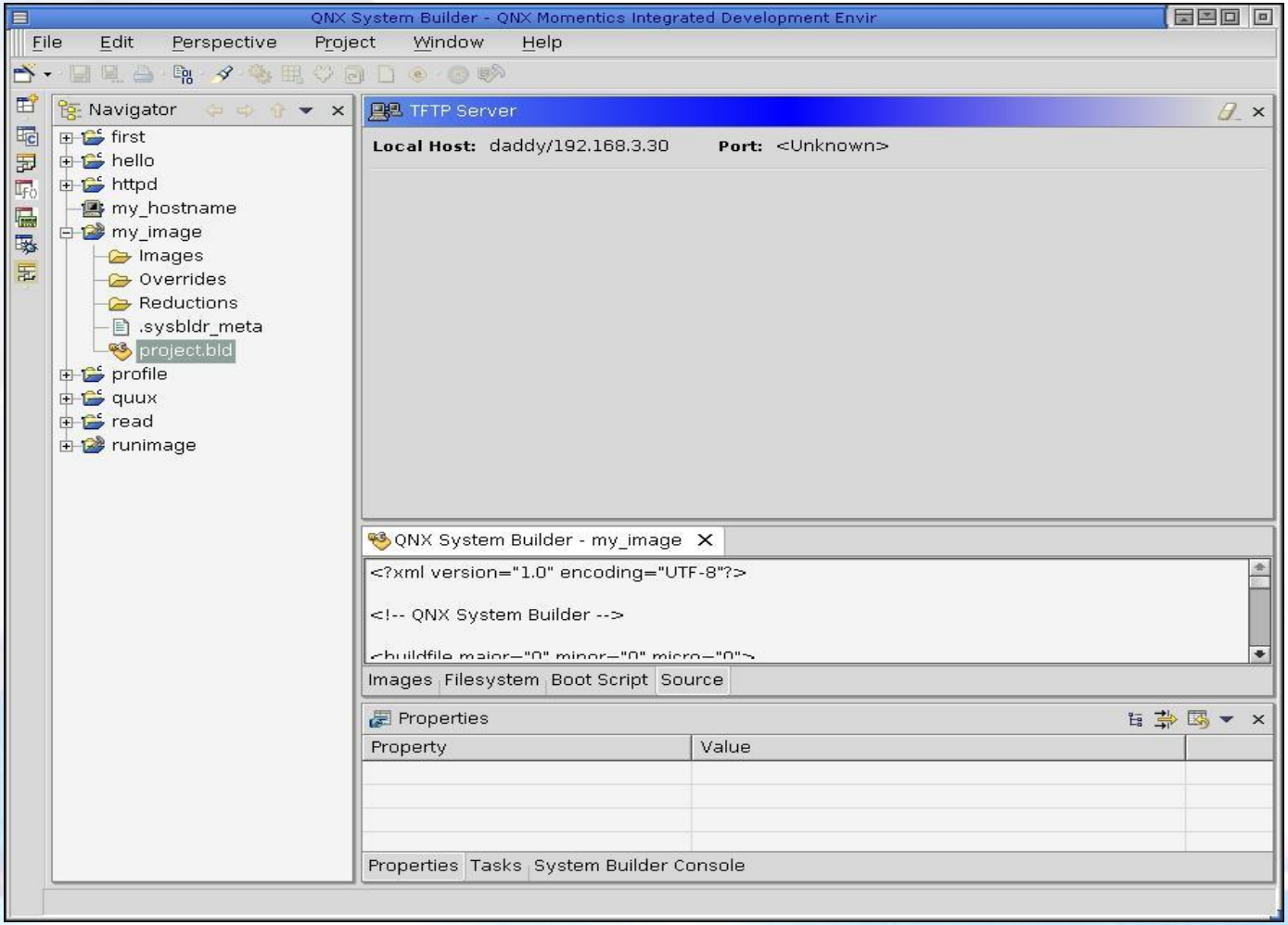
System Builder

Последовательный терминал



System Builder

TFTP сервер



Контроль версий

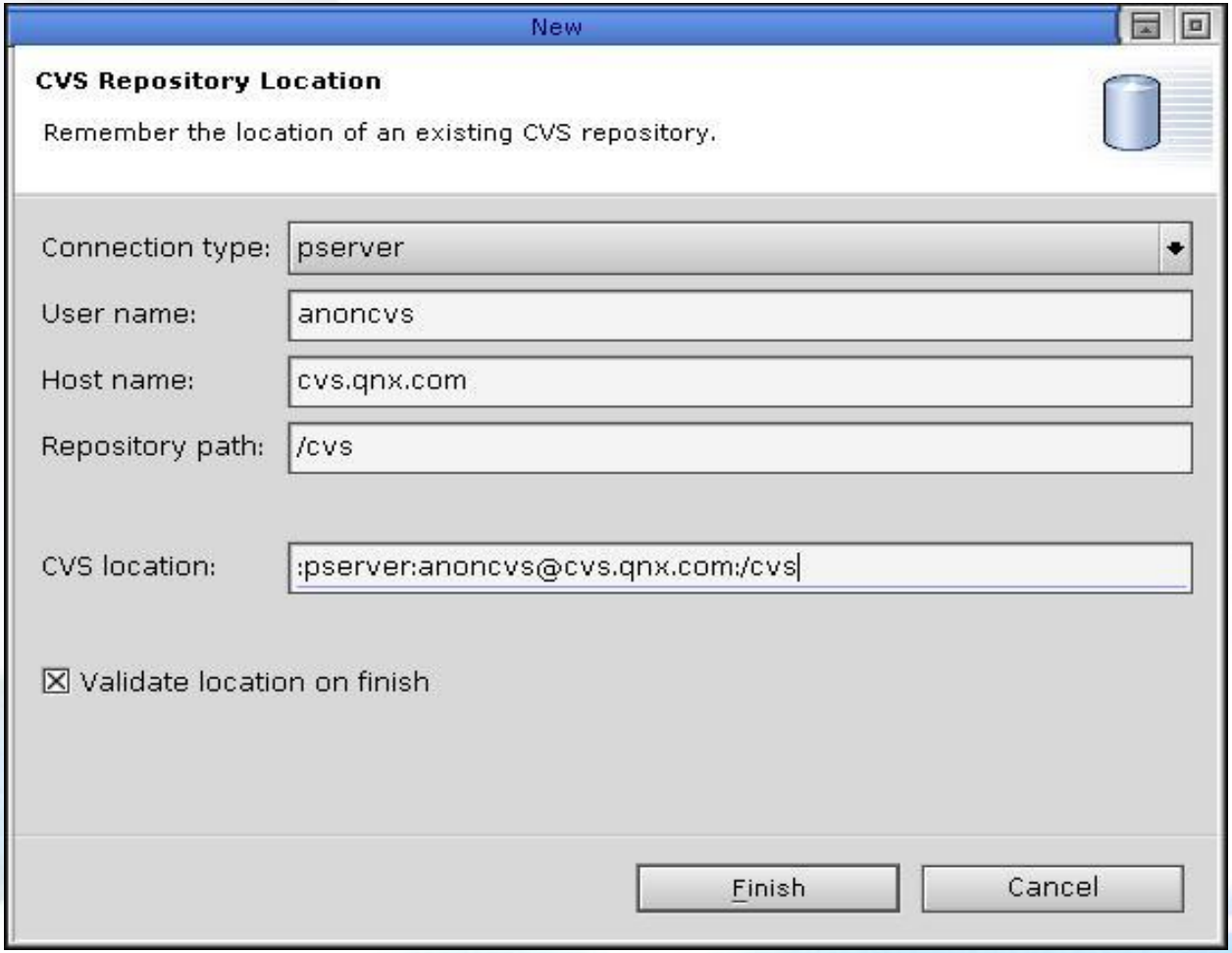
IDE предполагает, что вы будете использовать систему контроля версии для доступа к исходным кодам ваших проектов.

На данный момент IDE поддерживает следующие системы контроля версии:

- ✔ **CVS**
- ✔ **Rational ClearCase**

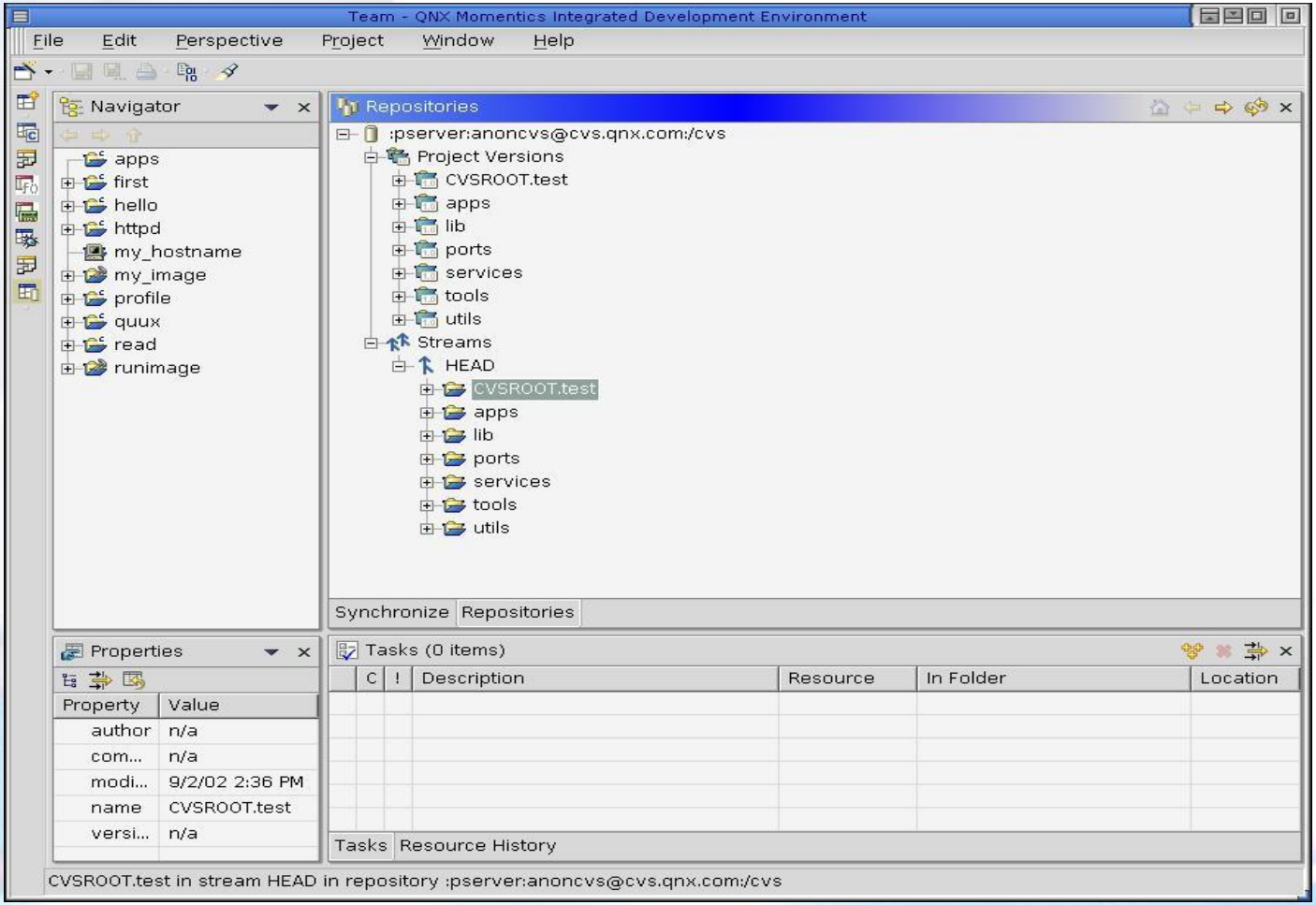
Контроль версий

Добавление репозитария



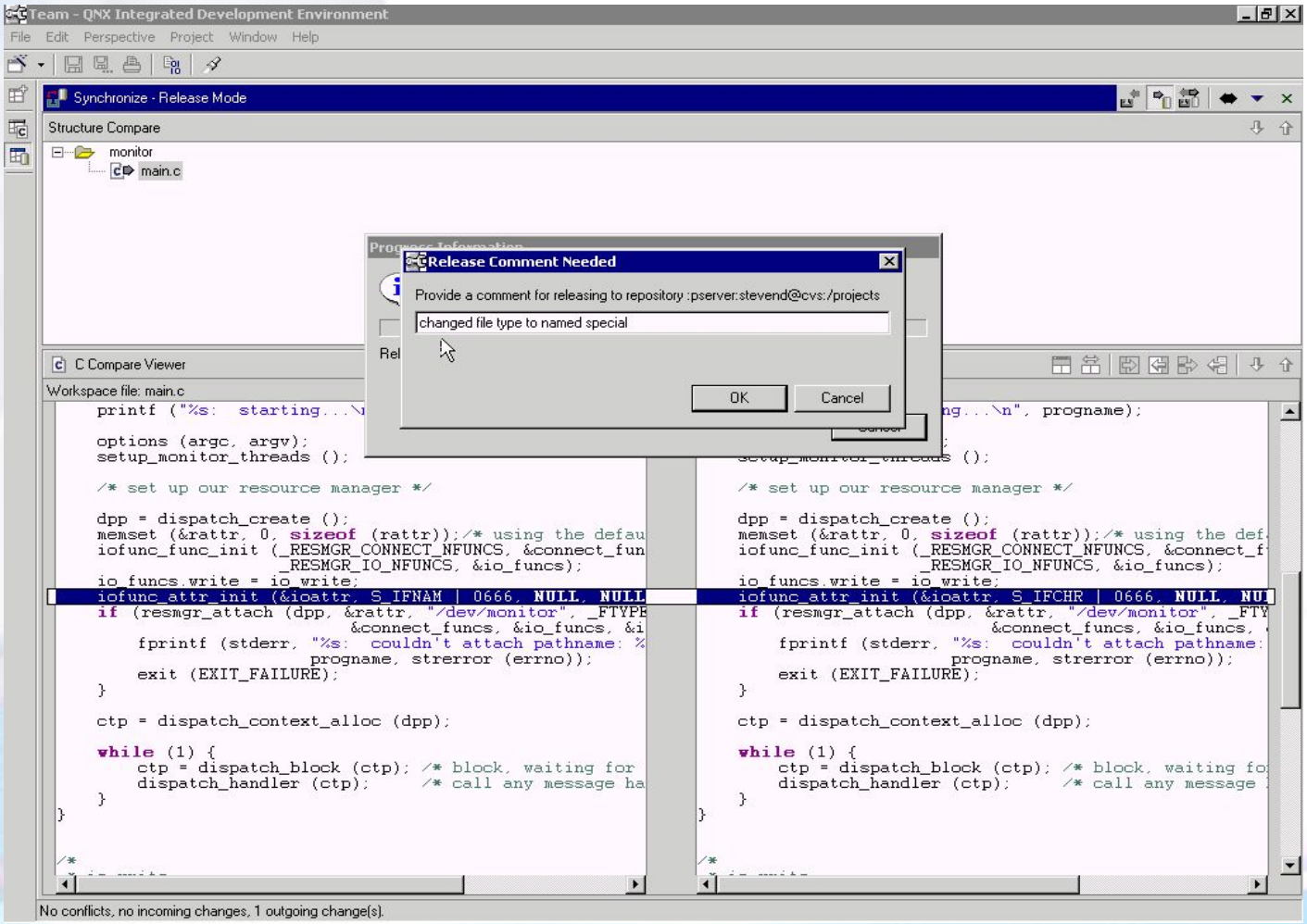
Контроль версий

Team perspective



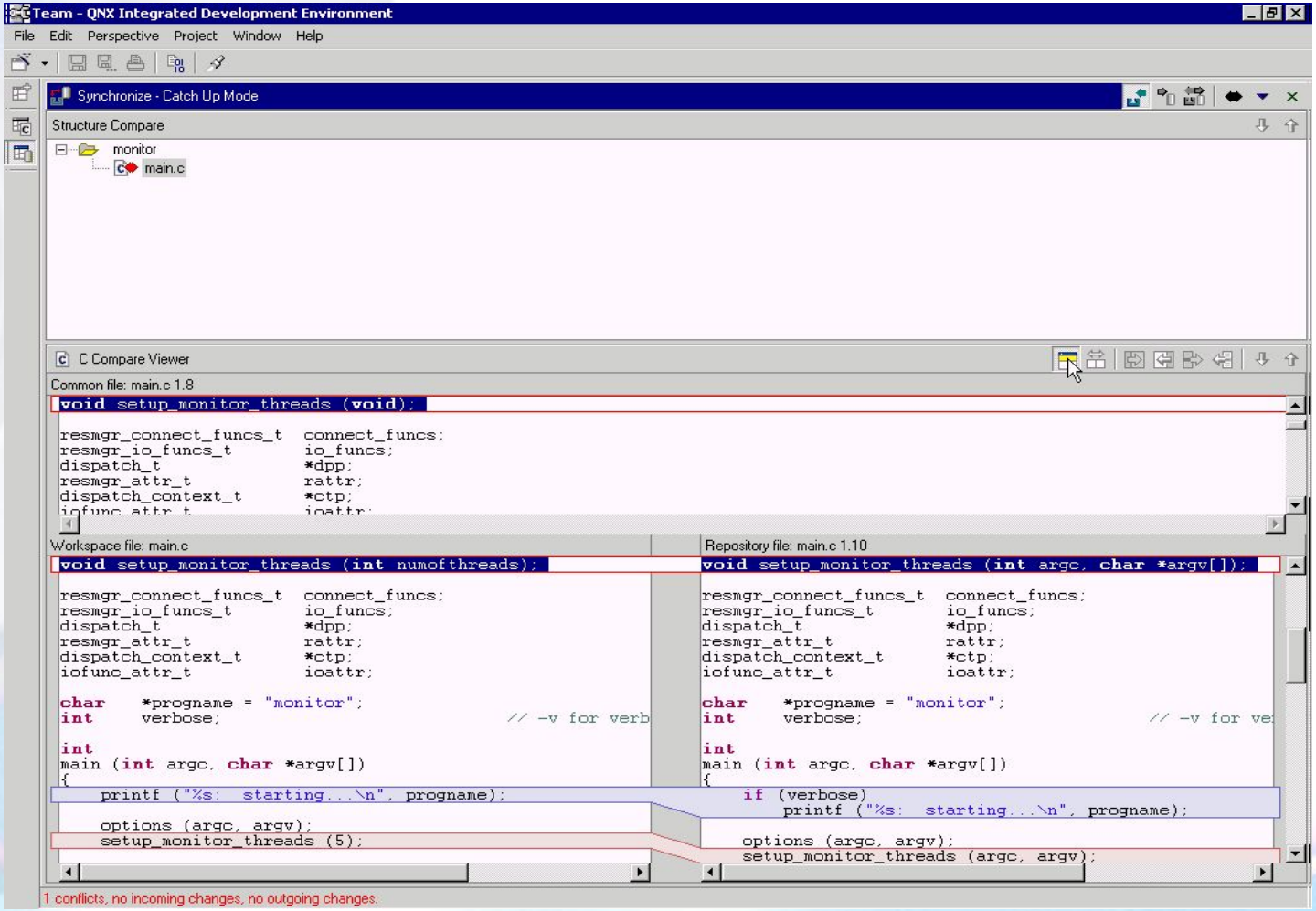
Контроль версий

Синхронизация с репозитарием



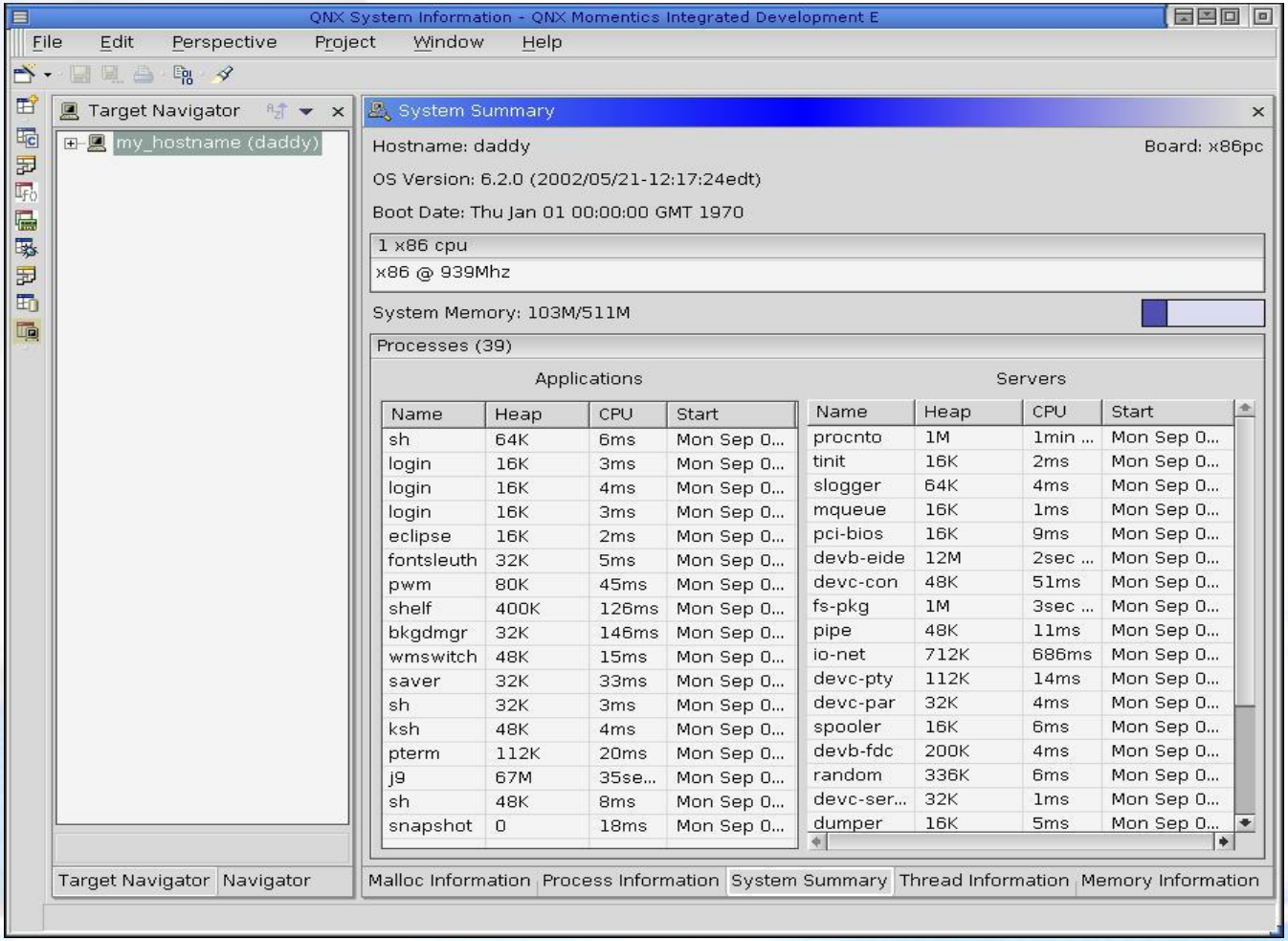
Контроль версий

Устранение конфликтов



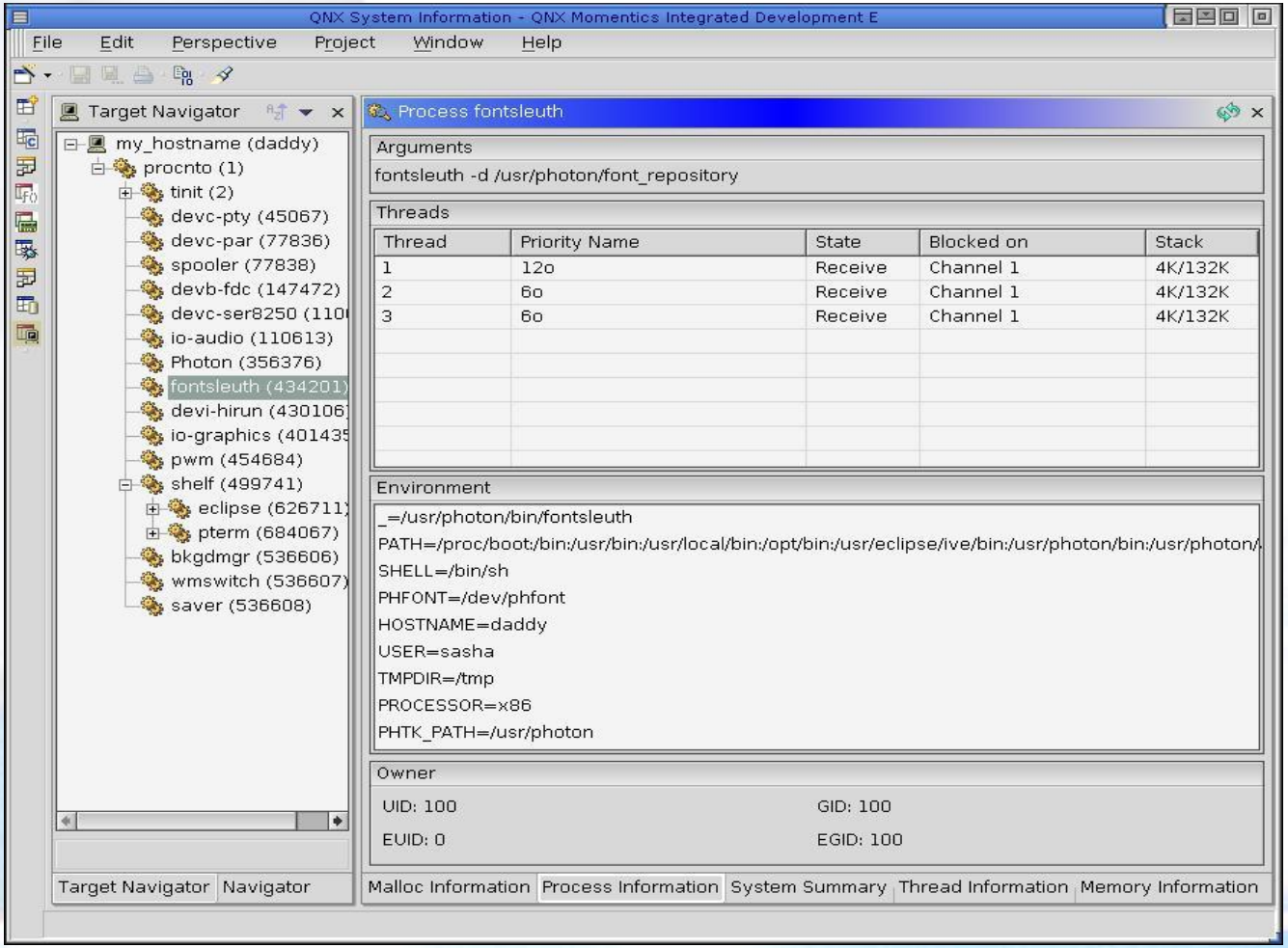
Информация о системе

System Information Perspective



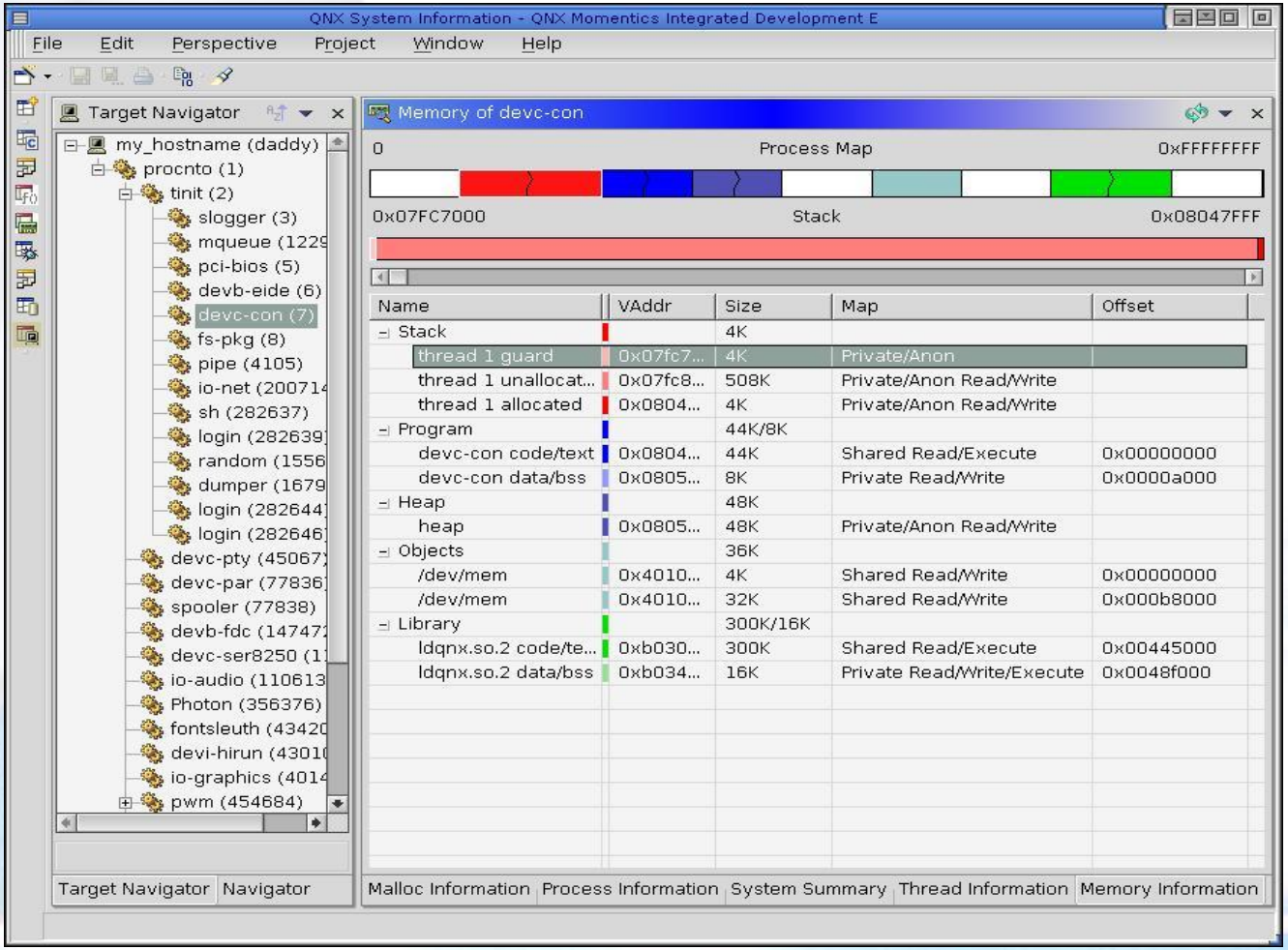
Информация о системе

Информация о процессах



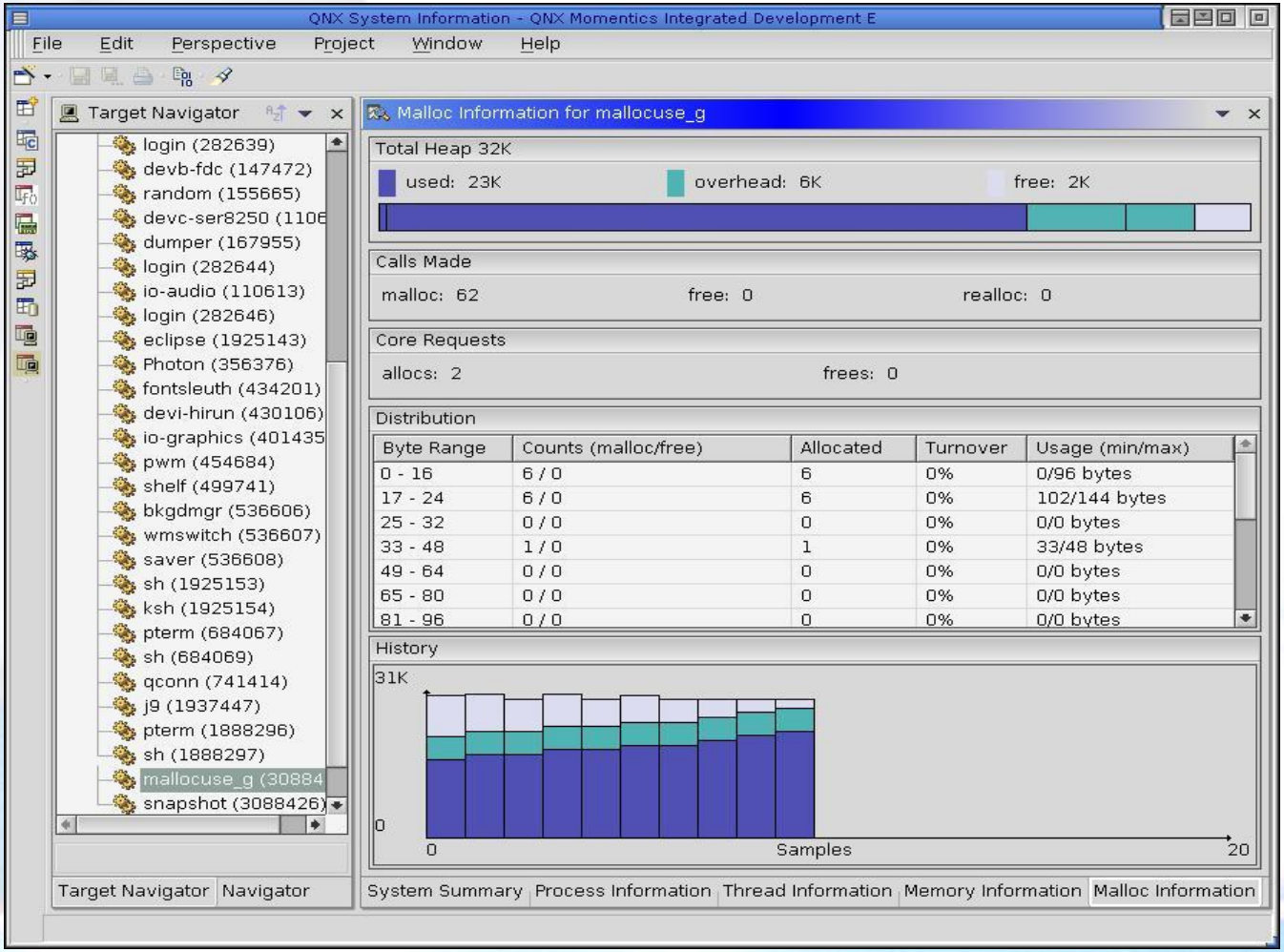
Информация о системе

Информация о памяти



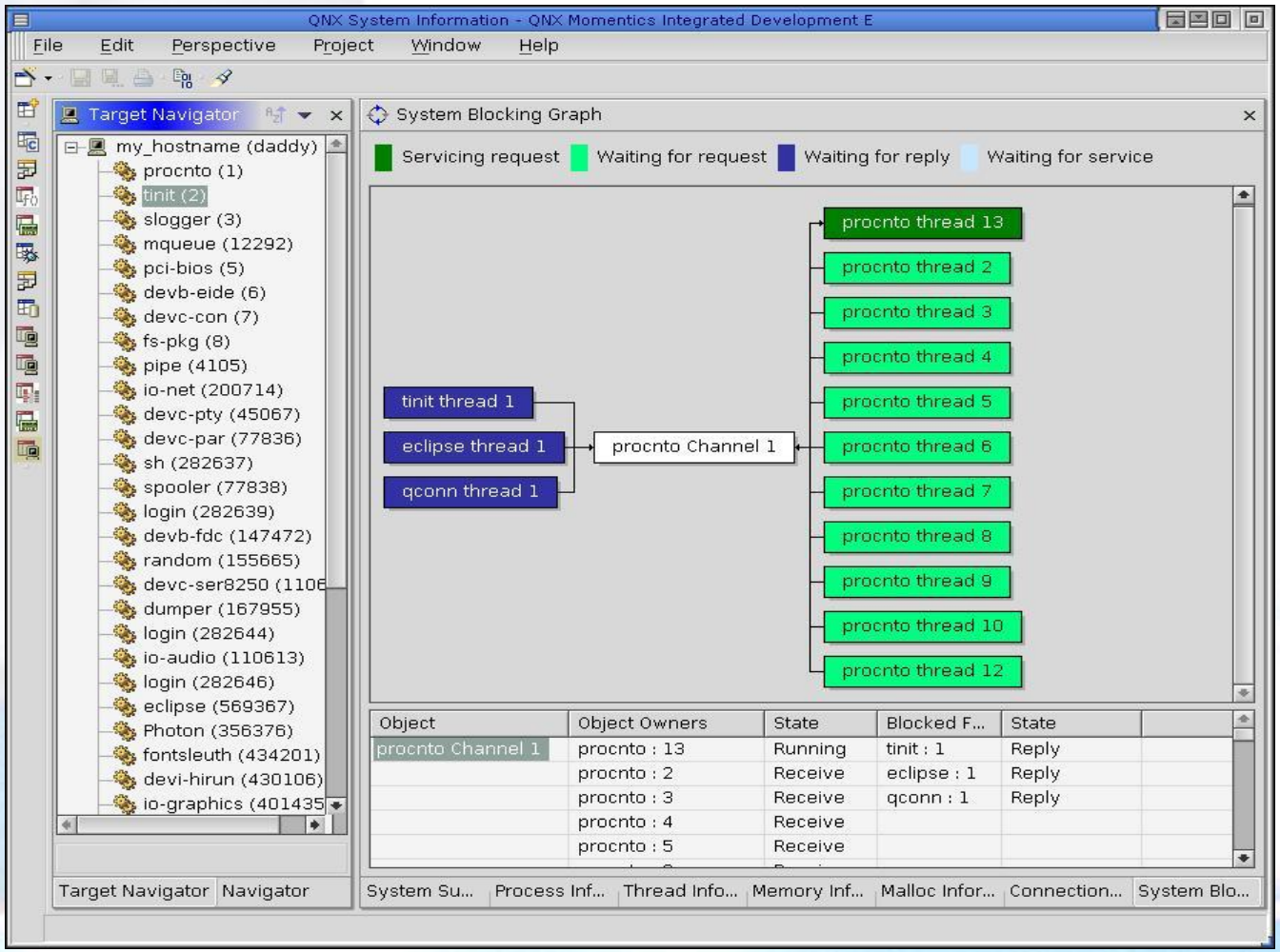
Информация о системе

Информация о выделении памяти



Информация о системе

System Blocking Graph



Информация о системе

Информация о сигналах

Profiling

Profiling (профилирование) – это определение участков кода, где ваша программа проводит большинство времени

Profiling помогает вам:

- ✔ сократить количество вызовов функций
- ✔ оптимизировать алгоритмы

Данные profiling состоят из двух частей:

Sampling:

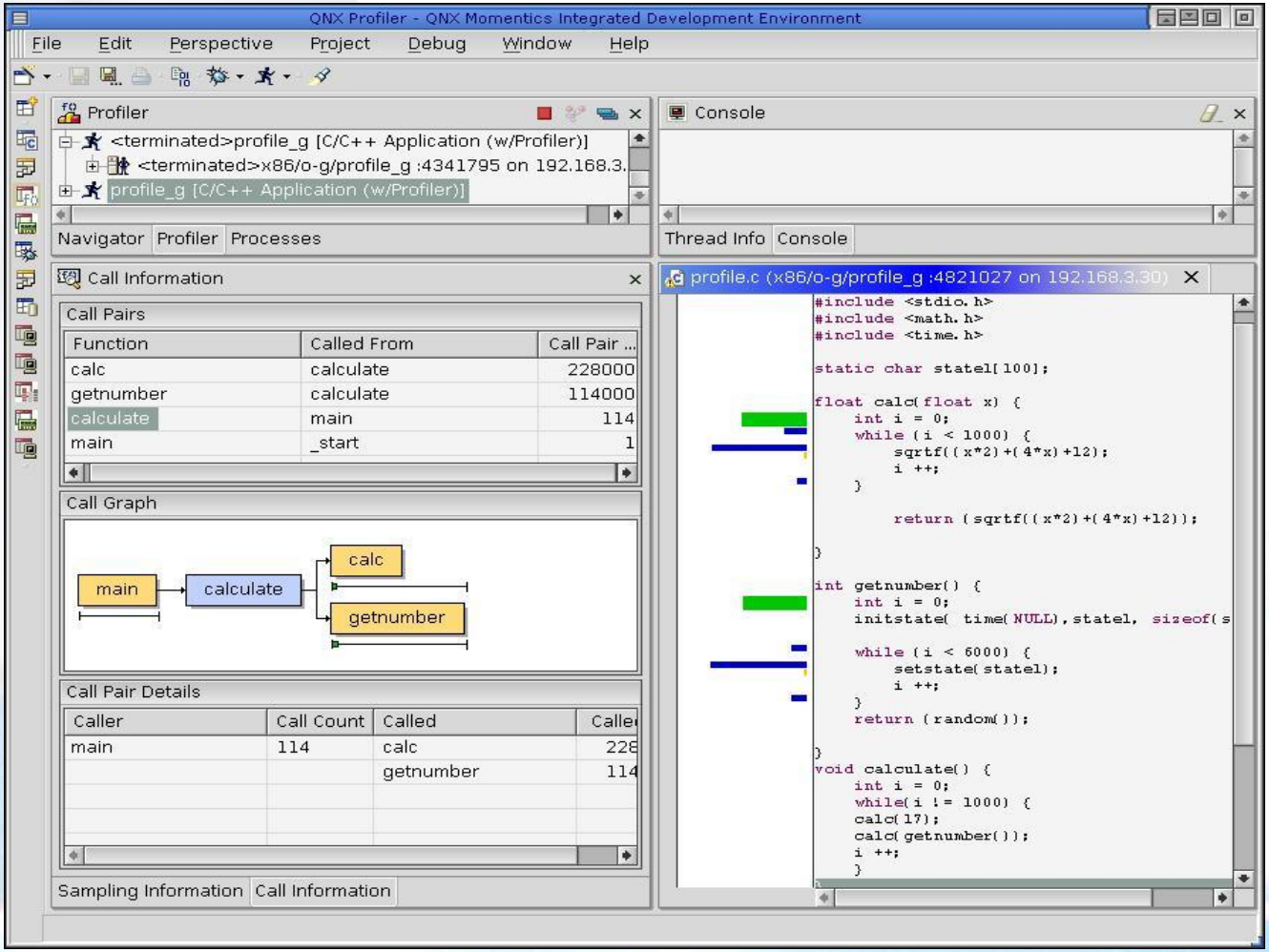
- ✔ Какой код исполняется
- ✔ Насколько часто исполняется код

Информация о вызовах:

- ✔ Количество вызовов функций
- ✔ Информация об инициаторе вызова

Profiling

Profile perspective



Profiling

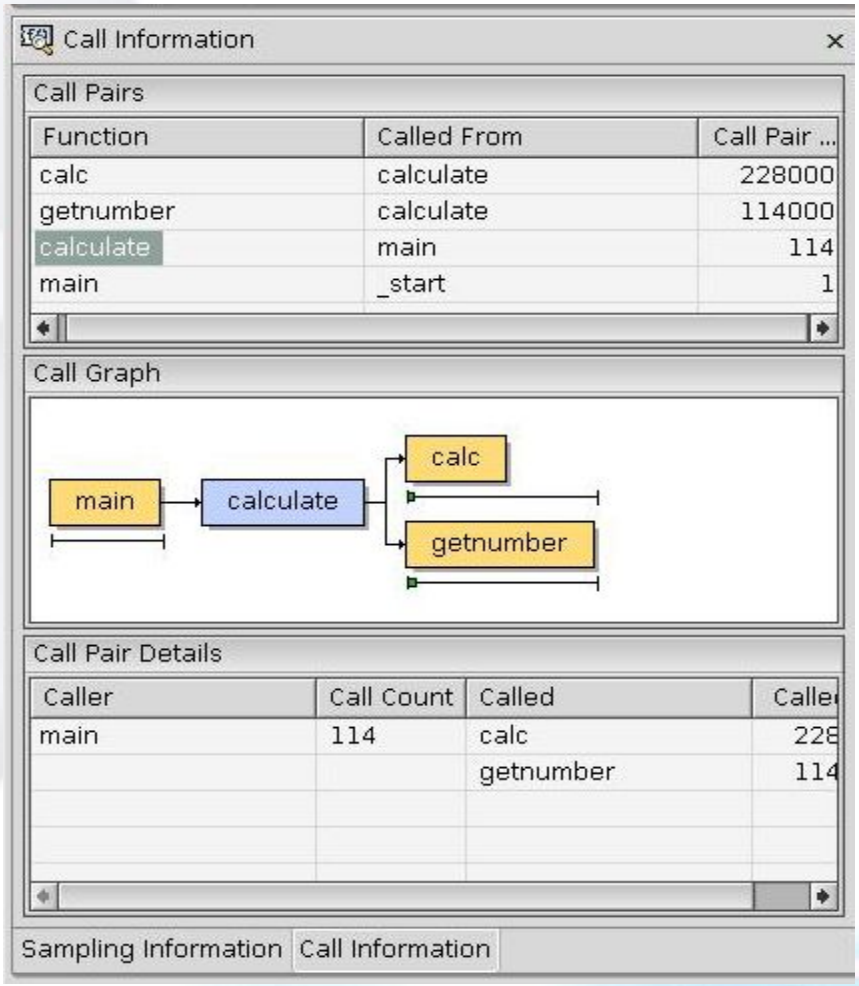
Sampling information

Sampling Information

Function	Total Time (s)	Time since ...	Call Count	usec/Call	% Time Usage
setstate	32.886	0.000			
sqrtf	19.100	0.000			
_FDunscale	4.535	0.000			
_FDscale	3.735	0.000			
calc	3.177	0.000	680000	4.672	
getnumber	2.966	0.000	340000	8.724	
.plt	0.650	0.000			
sqrtf	0.292	0.000			
random	0.213	0.000			
srandom	0.066	0.000			
__udivdi3	0.007	0.000			
initstate	0.006	0.000			
time	0.004	0.000			
pthread_self	0.002	0.000			
calculate	0.001	0.000	340	2.941	
main	0.000	0.000	1		

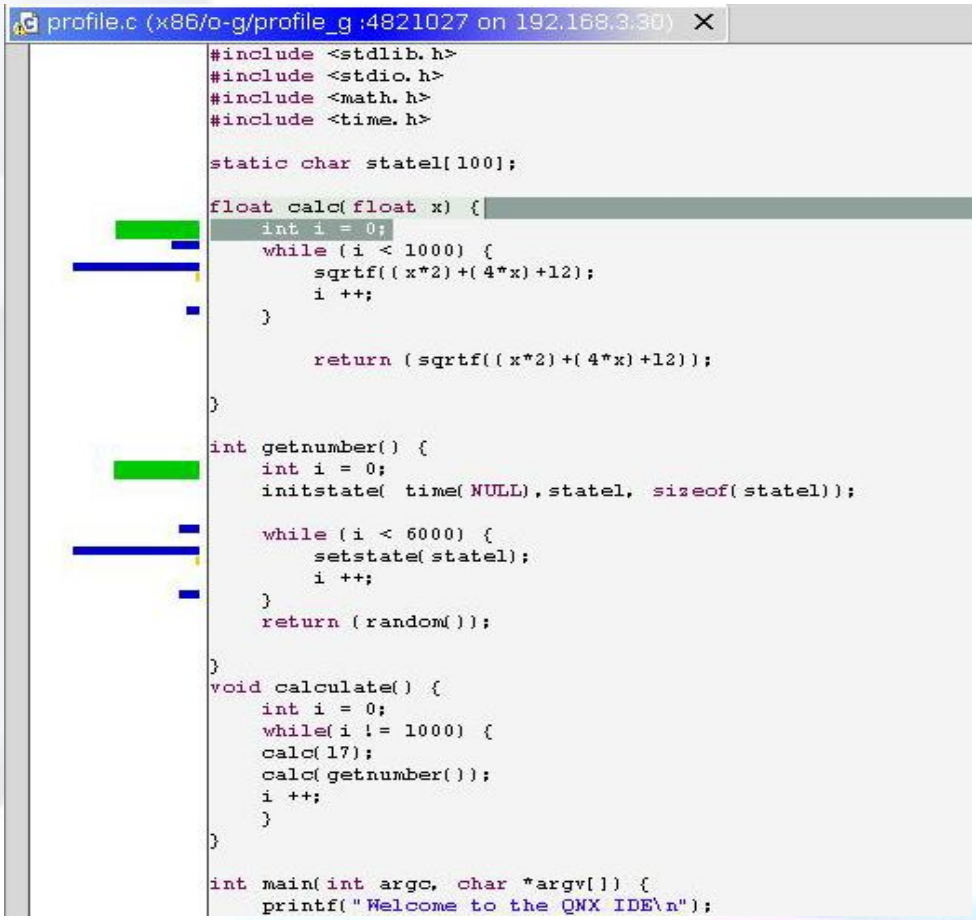
Profiling

Информация о вызовах



Profiling

Вывод информации о времени исполнения в исходном коде



Спасибо!

SWD Software Ltd.
Официальный дистрибьютор QNX

196135, Санкт-Петербург, пр. Юрия Гагарина 23
тел.: (812) 102-0833
тел.: (812) 373-0260
факс: (812) 373-0497
web: http://www.swd.ru/qnx/
e-mail: qnx@swd.ru