

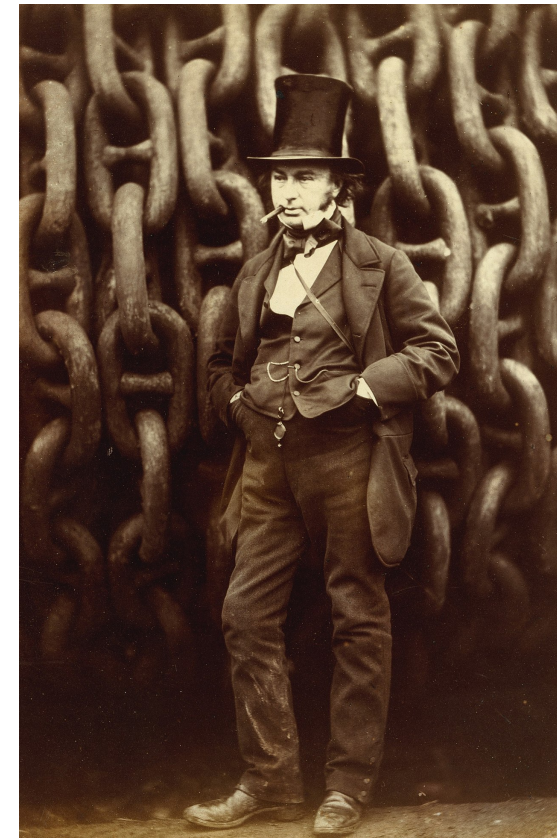
**James Price**

University of Bristol /  
GW4 Alliance



# Isambard Cray XC50 system and software environment

# 'Isambard' is a new UK Tier 2 HPC service from GW4



Isambard Kingdom Brunel  
1804-1859



# The tiered model of HPC provision

**Tier 0:** international



**Tier 1:** national



**Tier 2:** regional


TIER 2 HPC CENTRES

Edinburgh      Cambridge      UCL  
Loughborough      Bristol      Oxford

**Tier 3**



CRAY

 Isambard

**EPSRC**  
Pioneering research  
and skills

**GW4**

 UNIVERSITY OF  
**BATH**

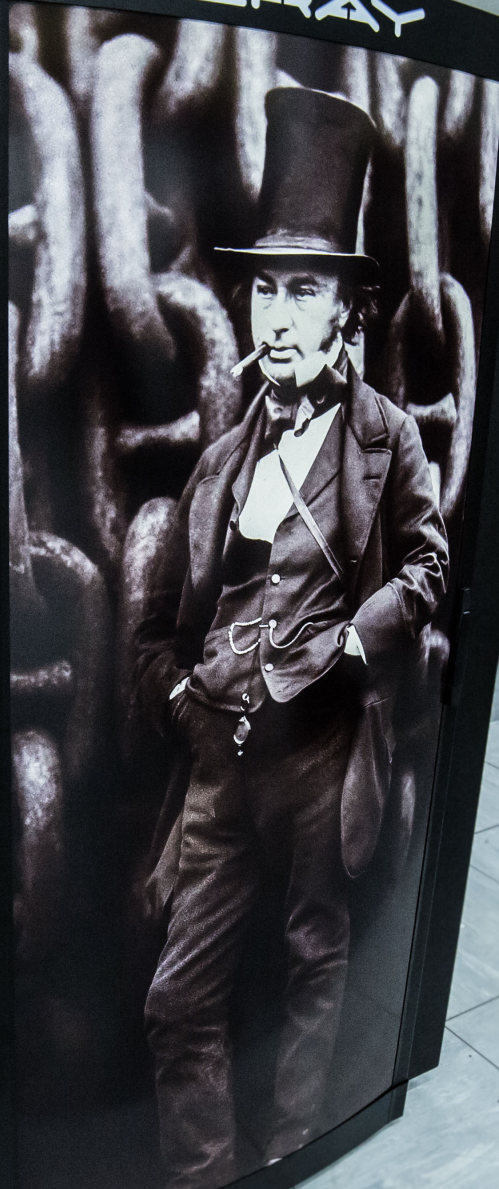
 University of  
**BRISTOL**

**CARDIFF**  
UNIVERSITY  
PRIFYSGOL  
CAERDYDD

UNIVERSITY OF  
**EXETER**

  
Met Office

arm



# Isambard system specification

- **10,752** Armv8 cores (168 x 2 x 32)
  - **Marvell ThunderX2 32 core @ 2.1GHz (2.5GHz turbo)**
  - **256 GB DDR4 memory per node**
  - **500 TB Lustre filesystem**
- Cray XC50 'Scout' form factor
- High-speed **Aries** interconnect
- Cray HPC optimised software stack
  - **CCE, Cray MPI, math libraries, CrayPAT, ...**
- Technology comparison (Phase 1):
  - **x86, Xeon Phi, POWER9, Pascal/Volta GPUs**
- Running as a production service (not a prototype)
- 25% of machine time allocated to EPSRC users via RAP



# Accessing Isambard

```
# Log in to the Isambard bastion node
# This node acts as a gateway to the rest of the system
# You will be assigned an account ID (01, 02, ...)
# Your password is vihpsAPR19
ssh br-trainXX@isambard.gw4.ac.uk
```

```
# Log in to Isambard Phase 2 from the bastion node
ssh phase2
```

# Using Isambard – modules

**# List available modules**

```
module avail
```

**# Load a module**

```
module load XXXX
```

**# Unload a module**

```
module unload XXXX
```

**# Change the programming environment/compiler**

```
module swap PrgEnv-cray PrgEnv-gnu
```

**# Compile MPI codes with the Cray wrappers:**

**# cc (for C)**

**# CC (for C++)**

**# ftn (for Fortran)**

# Using Isambard – running jobs

```
# Submit job
```

```
qsub job.sh
```

```
9376.xci00
```

```
# Check job status
```

```
qstat -u $USER
```

Job status:

**Q** = Queued

**R** = Running

**C** = Complete

(job will disappear shortly after completion)

Job ID	Username	Queue	Jobname	SessID	NDS	TSK	Req'd Memory	Req'd Time	Elap S	Time
9376.master.gw4	br-trai	arm	job.sh	154770	1	64	--	00:02	R	00:00

```
# Inside job script - use aprun to launch executable on compute nodes
```

```
aprun -n NUM_TASKS ./application args
```

```
# More information on aprun usage here:
```

<https://pubs.cray.com/content/S-2496/CLE%206.0.UP01/xctm-series-user-application-placement-guide-cle-60up01/run-applications-using-the-aprun-command>