Advancements in UNICORE accounting

Piotr Bała, Krzysztof Benedyczak, Rafał Kluszczyński,
Grzegorz Marczak
ICM, University of Warsaw

Outline

- UNICORE Accounting system overview
- New core features
- Accounting portal
- Future work
- Short portal demo

Fundamental facts

- "RUS" used for legacy reasons.
- Accounting data retrieved from two sources:
  - XNJS processor (distributed with Unicore/X)
  - Standalone LRMS agent
- Records merged by a central RUS-service
  - can be exported to other services
- Communication via JMS messaging
- Record format based on the OGF UsageRecord.
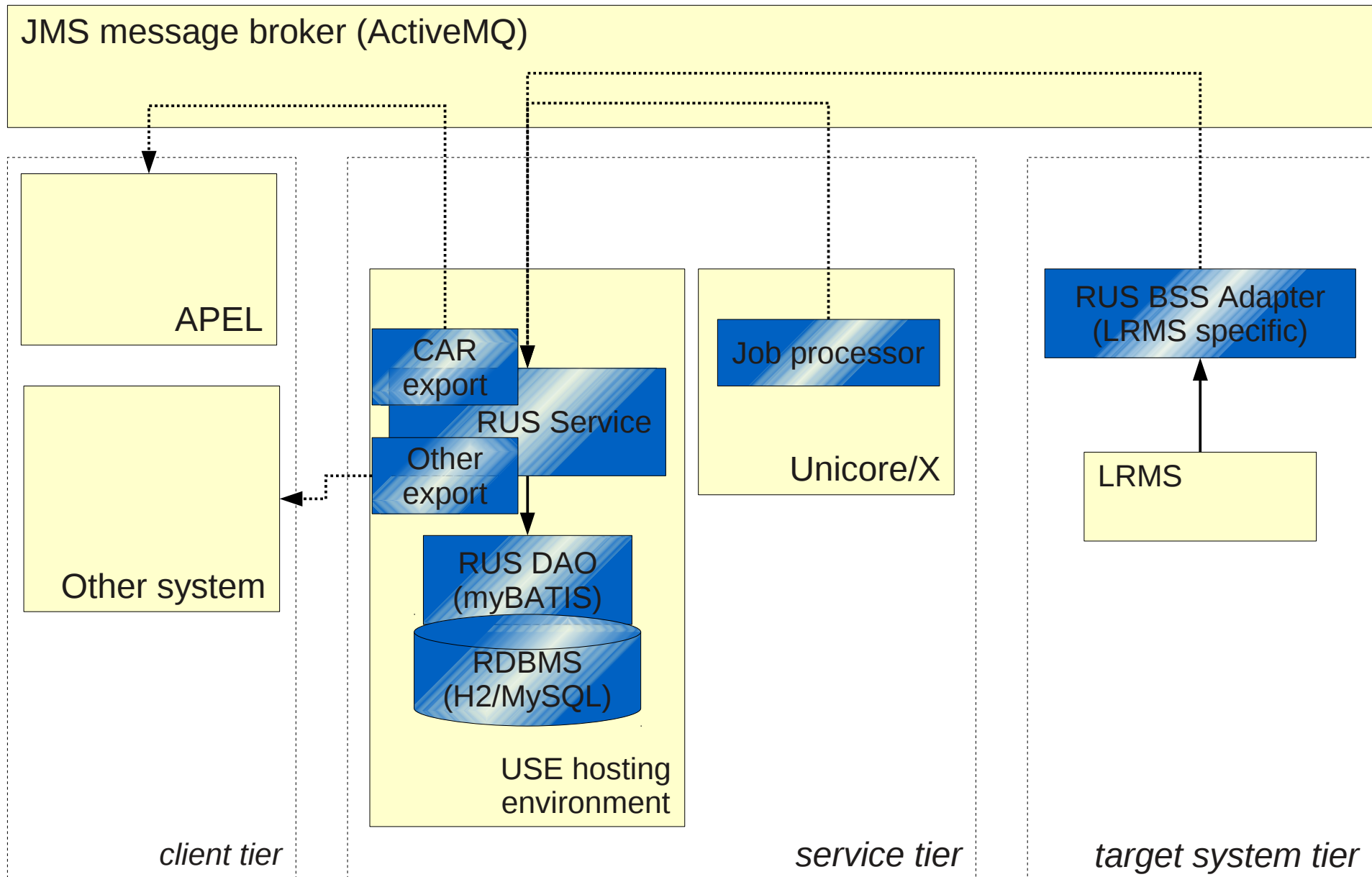- Java web application available as a visualization tool

The big picture
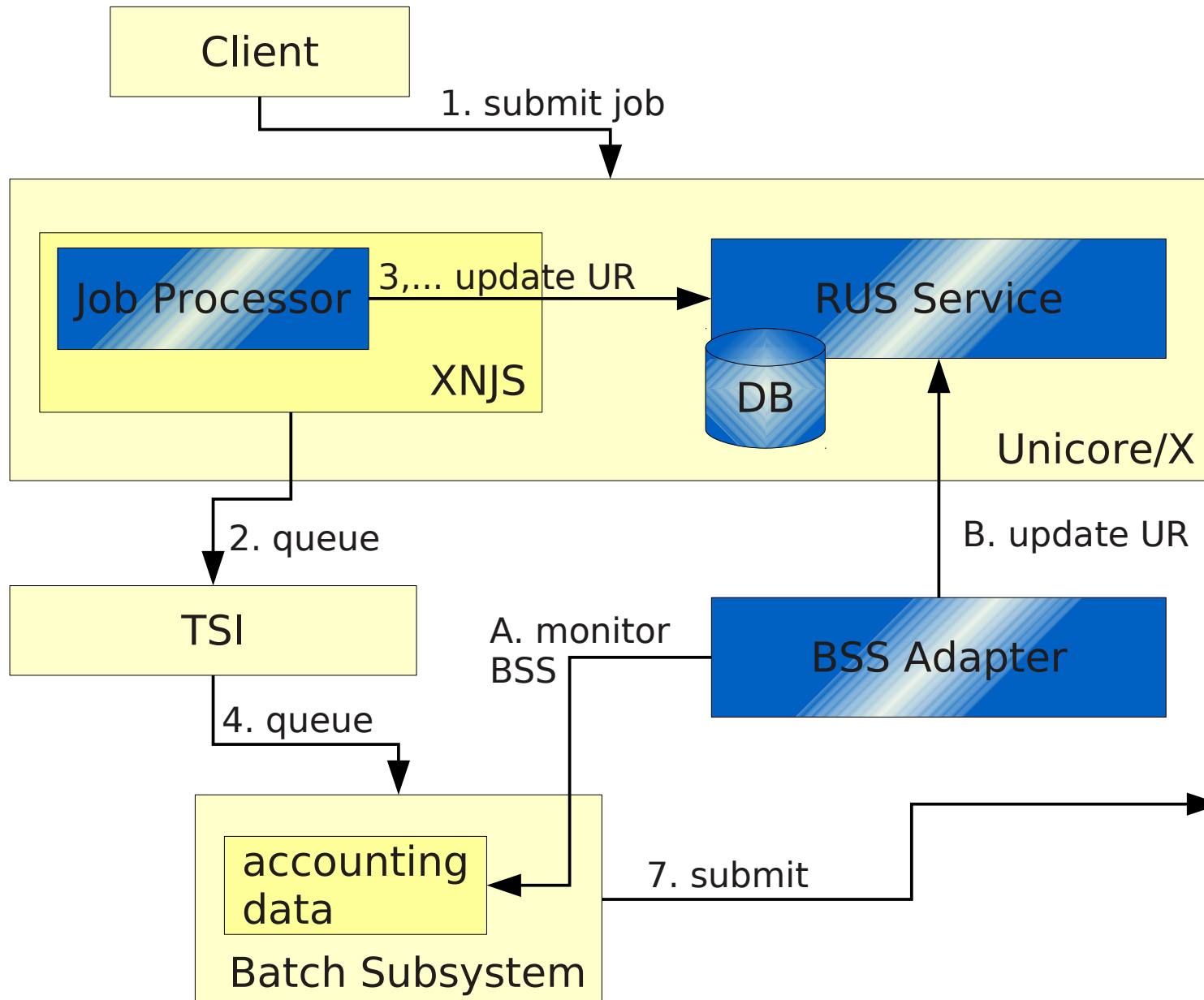
Recent new features

- XNJS job-processor module integrated with Unicore/X since 6.5
- Added support for SLURM in bss-adapter
  - along with Torque and SGE/OGE
- Enhanced the collected data
  - e.g. maximum job time known after submission
  - U/X time (start at U/X, submit to BSS, …)
- More reliable merging
  - revised algorithms handling mostly all, even very weird cases
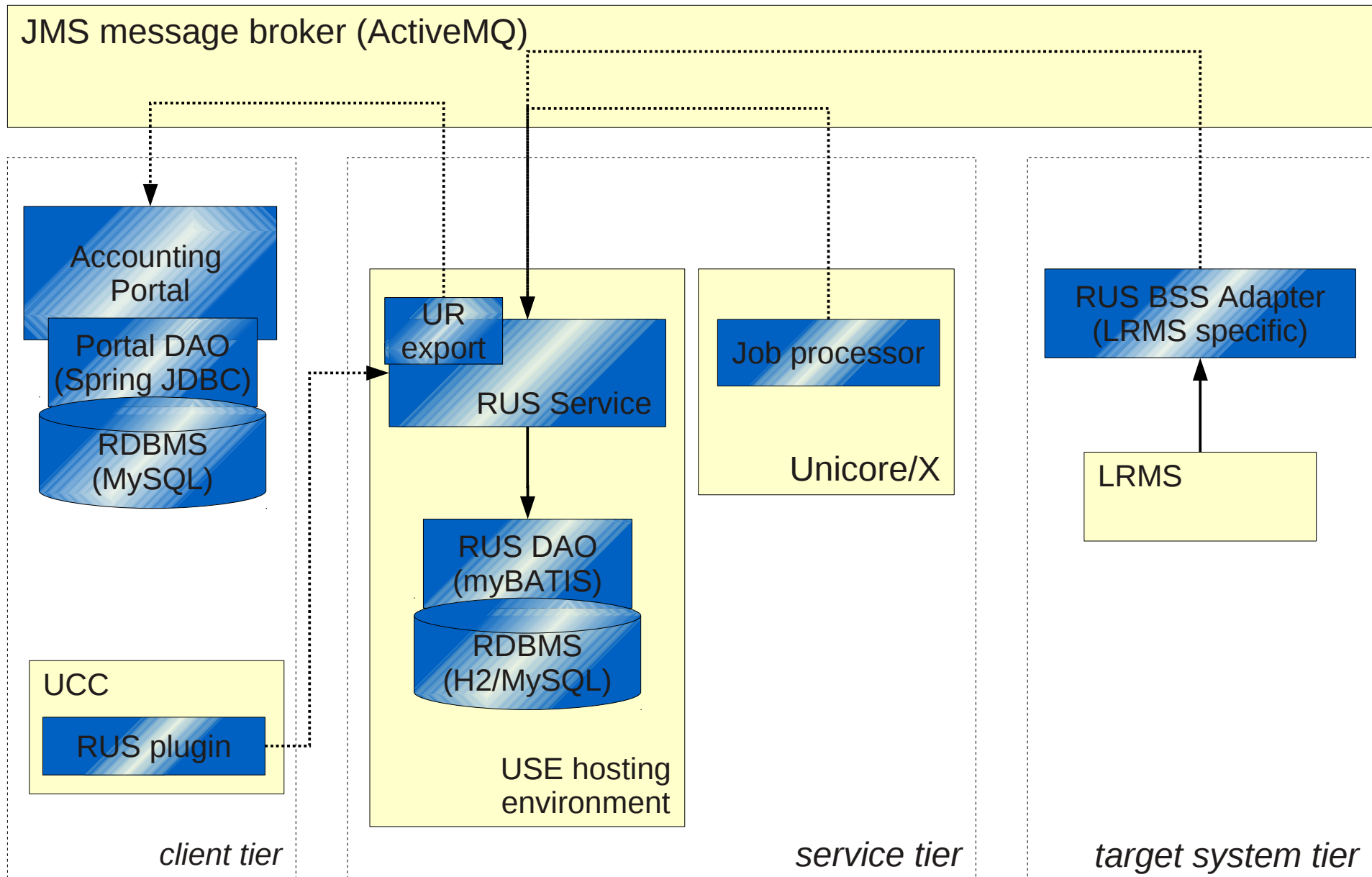- APEL export (EMI CAR & EMPA support)

Another big picture
Export to 3rd party systems

JMS message broker (ActiveMQ)

APEL

Other system

CAR export

RUS Service

Other export

RUS DAO (myBATIS)

RDBMS (H2/MySQL)

USE hosting environment

Job processor

Unicore/X

RUS BSS Adapter (LRMS specific)

LRMS

client tier

service tier

target system tier

The 1.x DB layer

- 1.x version featured a central DB, exclusively maintained by rus-service.
  - hybrid, XML + caching columns
- Around 1,000,000 job records, the DB started to become a bottleneck
  - record merging/insertion and queries
- The portal UI had its own copy of the DB with custom format.
 - same problems with performance, 'custom' bugs, ...

Old, 1.x DBs

JMS message broker (ActiveMQ)

Accounting Portal

Portal DAO (Spring JDBC)

RDBMS (MySQL)

UCC

RUS plugin

UR export

RUS Service

RUS DAO (myBATIS)

RDBMS (H2/MySQL)

USE hosting environment

Job processor

Unicore/X

RUS BSS Adapter (LRMS specific)

LRMS

client tier

service tier

target system tier

The new DB layer

- Database layer is now a separate software component, shared by the portal and rus-service.
- A table with daily job summaries was greatly enhanced and become a primary source for reporting queries.
  - It holds the data which is 90% of time sufficient for administrators/management.
- Both jobs and summaries tables have separate 'rolling' mechanism:
  - old records are automatically moved to 'history' tables (by default after 6m and 3 years).

The big picture
2.x version
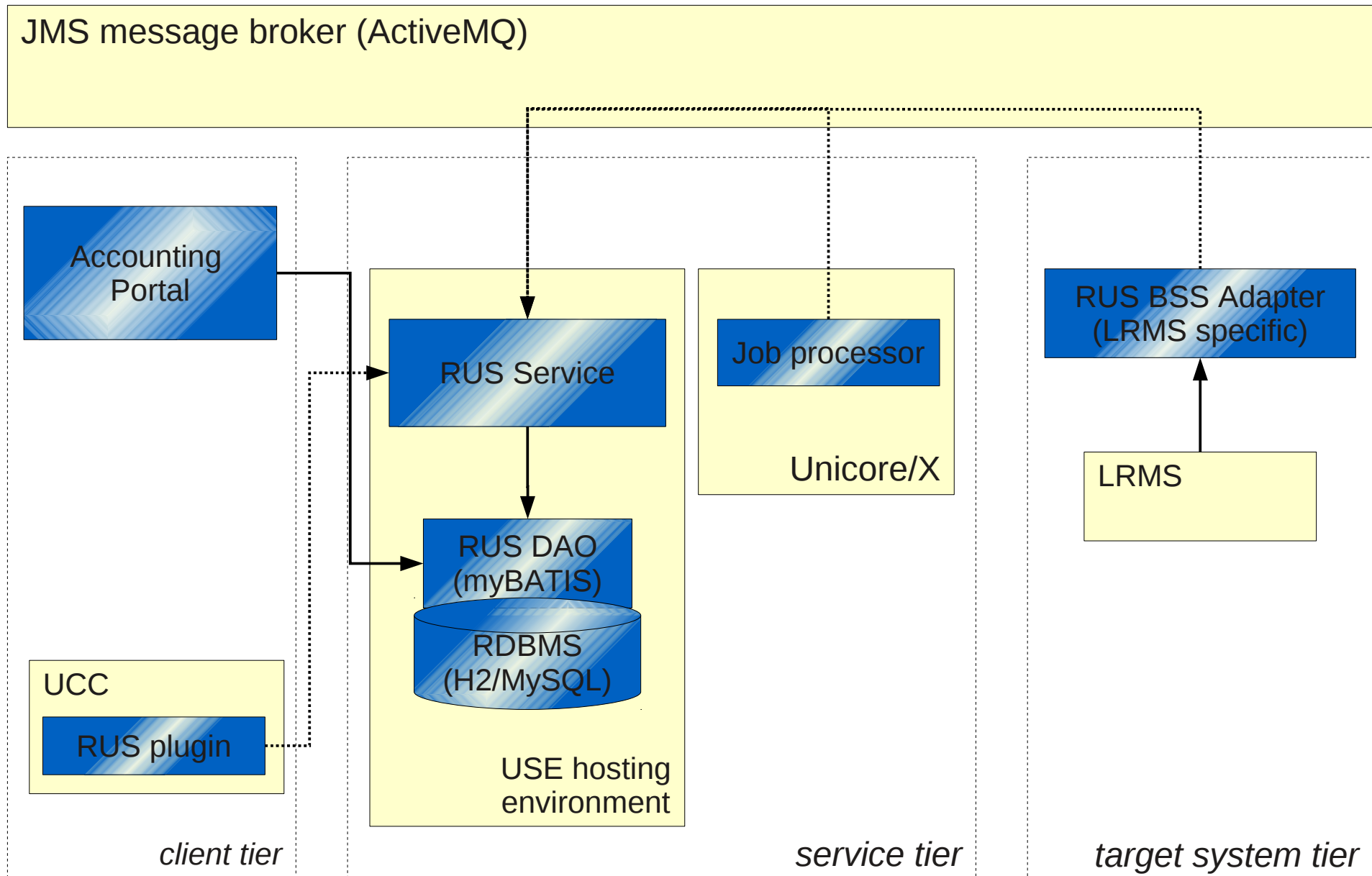
The new portal

- **A common database layer used.**
  - Code elimination, uniform performance characteristics
  - accounting data not processed by the portal anymore.
- **JMS and own database are now not used**
  - May be easily readded, if it is needed to haev a separate portal DB.
- **The fundamental interface parts are based on the daily-aggregated information**
  - greatly increased performance

The new portal (2)

- ◆ The search criteria panel was extended.
- ◆ It is shared between data presentation views.

The new portal (3)

◆ Search criteria are automatically filtered for users with limited permissions.

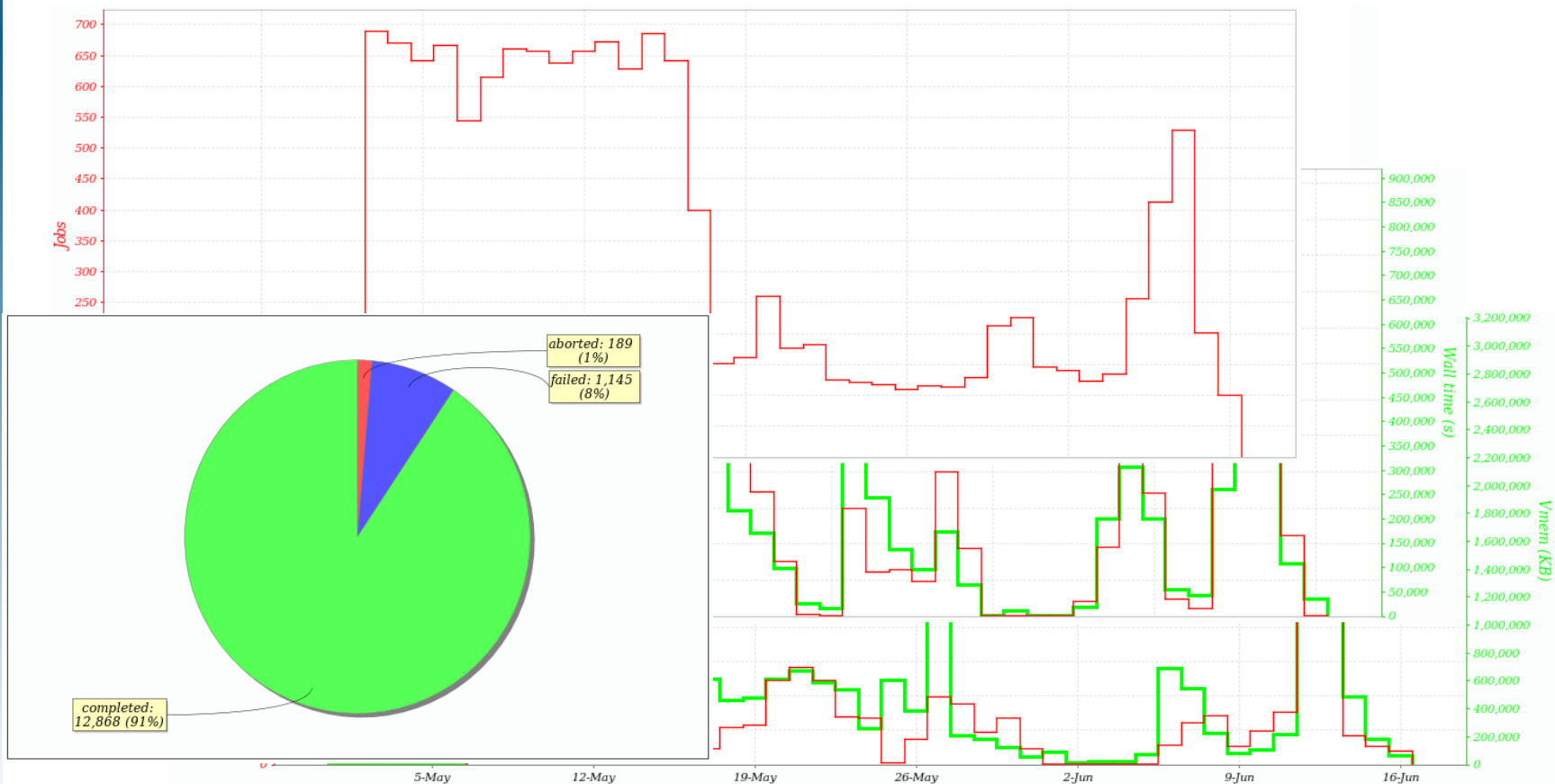▪ Only criteria available for the user-visible jobs are presented.

The new portal (4)

- Fully rewritten charts view.
  - Still a lot to improve here...

Current & future work

- Additional enhancements in web UI
- Accounting of reservations
- (?) LoadLeveler support
- (?) Support for separate DB in portal (still the same format as in case of rus-service).

Another big picture
separate portal DB

JMS message broker (ActiveMQ)

Accounting Portal

RUS DAO (myBATIS)

RDBMS (H2/MySQL)

UCC

RUS plugin

UR export

RUS Service

RUS DAO (myBATIS)

RDBMS (H2/MySQL)

USE hosting environment

Job processor

Unicore/X

RUS BSS Adapter (LRMS specific)

LRMS

client tier

service tier

target system tier