

JuxMem: Juxtaposed Memory

Table of contents

1 Projects that use JuxMem.....	2
2 Middlewares that use JuxMem for storing data.....	2

1. Projects that use JuxMem

JuxMem is used in the following projects:

- The [GDS](#) (Grid Data Service) project ;
- The [ASSIST](#) project.

2. Middlewares that use JuxMem for storing data

JuxMem is used in the following middlewares:

- The [DIET platform](#), a GridRPC-based middleware. An example of how JuxMem is used in DIET is shown below. To be more precise, DIET is dynamically linked to the JuxMem-C library, as all the entities of DIET are using the client side of JuxMem. Please Note that JuxMem-J2SE is used for all other entities required in a JuxMem network of peers.

DIET without JuxMem
DIET with JuxMem

In this example, a client C successively uses two servers (S1 and S2) for two different computations. We assume that the second computation depends on the first one: the output data D2 produced on server S1 is used as input data for the computation scheduled on S2. We also assume that D2 is intermediate data that is not needed by the client. On the left side, we illustrate a typical scenario using the current GridRPC API, with no support for data management. The client C needs to explicitly transfer the output data D2 from server S1 to server S2 (steps 2 and 3). Then, the second computation on server S2 can take place and returns data D3 to client C (step 4). On the right side we show how these computations would be handled if the GridRPC infrastructure provided support for localization transparency and persistence. The server S1 stores the output data D2 in a data management infrastructure (step 2). Then, the client C only needs to transmit the ID of data D2 to S2 (step 3a). Consequently, the data transfer between S1 and S2 occurs in a transparent manner for the client (step 3b). We assume that the servers are connected to the storage service using high-performance links, whereas this may not be true for the links between clients and servers. Using a data management infrastructure clearly avoids unnecessary and costly data transfers between the client and servers. Arrows 2 and 3 (left side) represent these unnecessary data transfers that can be optimized out when using a data management infrastructure.

As a preliminary step, a data management service called Data Tree Manager (DTM) has been specifically developed for the DIET platform. This solution uses DIET's computing servers for persistent data storage and needs no external storage resources. However, a simpler and more flexible approach is to fully let data management at the charge of an

external data-sharing service. As explained in the previous section, the benefits of such a service consist in its mechanisms for transparent access, persistence, fault tolerance and consistency. This approach is at the core of the design of the JuxMem software platform.

The [Grid-TLSE](#) application uses DIET for solving matrices-based problems. This application has been successfully and transparently using JuxMem for storing persistent matrices.

- A prototype of CCM components that use JuxMem is working. This work is based on JuxMem-J2SE and [OpenCCM](#) as well as on JuxMem-C and [MicoCCM](#). A more mature solution will be available soon.
- A work is currently in progress so that [P3](#) (Personal Power Plant) can use JuxMem. This work is based on JuxMem-J2SE.