HPN / DC&IT – Winter Semester 2013/2014

Practical Assignment – Part 1

Due date: 09.12.2013

Write a simple calendar tool that allows the synchronization of appointments between different hosts. It is sufficient to implement a text-based console application that provides facilities to add, remove and modify appointments. Adding a graphical user-interface is optional.

The tool you are implementing needs to allow the interconnection of several machines to a “calendar network” where all hosts share the same set of appointments. Changes on one machine need to be propagated to all other machines. Note that you do not need to implement routing facilities. For this purpose each host needs to know the addresses of all other hosts. A discovery mechanism is also not necessary. New machines **join** the network by sending a register message to one of the machines already in the network. The address of the new host is thereupon propagated in the network. Hosts also need to be able to **sign off** from the network again.

An appointment in the calendar needs to have at least the following properties: A unique sequential number, date, time, duration, header and comment. Make sure, that appointments are persistently stored on each machine. When a host is restarted it needs to synchronize its local set of appointments with the appointments currently in the network.

Your tool has to provide functions to **add**, **remove** and **modify** appointments. In this first part of the practical assignment you do not need to care about inconsistencies caused for example by concurrent appointment creation or modification. Additionally, users need to be able to **list** all available appointments.

The assignment has to be solved in groups of four students. Each group needs to provide two different implementations of the tool based on two different programming languages. We propose to implement the solutions in **Java** and **C#**. However, if you feel more comfortable using other languages, feel free to do so. In this case you need to indicate the programming languages you are using in advance. It is important, that the two solutions are interoperable. For this purpose in this assignment you must use **XML-RPC** for the inter-machine communication.

You need to submit your solution by December 9th 2013. Details how to submit will be announced soon. Each group needs to present their solution personally. The presentation comprises a demo of your tool as well as an explanation of the source code. It is particularly important that the two different implementations are interoperable and that you show this in your demo. All group members need to be able to explain the source code and only those group members that are present for the presentation can pass the assignment.

You need to register your group till 4th of November by sending an email to markus.esch@fkie.fraunhofer.de. Please indicate all members of your group including email addresses, student IDs and respective study courses. In case you intend to use other programming languages than Java and C# indicate this in your registration email. The subject of your registration email needs to be: **HPN / DC&IT group registration**. In case you don’t find a group or do not get an email receipt of your registration contact us.

Summing up, you need to provide the following:

- A command line-based calendar tool
- Interconnection of multiple instances of the calendar tool on different machines

- Mechanisms to join and leave the “calendar network”
- Propagation of appointments and appointment changes to all machines in the “calendar network”
- Functions to add, remove, modify and list appointments
- An appointment has to have at least the following properties: A unique sequential number, date, time, duration, header and comment
- Persistent storage of appointments and synchronization after restart
- Two different implementations of the calendar tool with two different programming languages (We propose Java and C#)
- Inter-machine communication based on XML-RPC
- The two solutions need to interoperable

To get started, those links and hints might be useful:

- You can either use the Visual Studio 2013 from MSDN AA or the free Visual Studio Express to implement your C# solution. The most convenient way to realize XML-RPC in .Net is to implement a WCF-Service (http://tiny.cc/lnr5x) with XML-RPC endpoint behavior. An implementation of the XML-RPC protocol for WCF can be downloaded here: http://tiny.cc/gk6pp
- A Java XML-RPC library is provided by Apache and is available here: http://ws.apache.org/xmlrpc/